Integration of passenger satisfaction in railway timetable rescheduling for major disruptions

THÈSE N^{O} 8209 (2018)

PRÉSENTÉE LE 26 JANVIER
À LA FACULTÉ DE L'ENVIRONNEMENT NATUREL, ARCHITECTURAL ET CONSTRUIT
LABORATOIRE TRANSPORT ET MOBILITÉ
PROGRAMME DOCTORAL EN GÉNIE CIVIL ET ENVIRONNEMENT

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Stefan Yves Gabriel BINDER

acceptée sur proposition du jury:

Prof. D. Lignos, président du jury
Prof. M. Bierlaire, directeur de thèse
Prof. M. Gendreau, rapporteur
Dr M. Laumanns , rapporteur
Prof. F. Corman, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2018

"Prediction is very difficult, especially about the future."

— Niels Bohr

Acknowledgements

I always said I would never do a PhD. Looking back however, I realize that the experience of creating this piece of research work was one amazing journey, and that I do not regret anything about changing my mind. During this extended journey, several people helped, accompanied, supported and encouraged me along the way. I am deeply grateful to all of them, because this thesis would not have been possible without them.

First and foremost, I would like to thank my supervisor, Prof. Michel Bierlaire. I would not have accomplished this thesis without his guidance and expertise, which helped me to focus my ideas in the most critical moments of the process. He was always available when I needed advice, and his constructive feedback pushed myself towards excellence. He is not only an outstanding researcher, teacher and mentor to work with, but also an extremely human person. He manages to create a fantastic working environment, as all (former and current) members of the lab can attest. Indeed, it was a real pleasure to share non-academic discussions with him over beers, raclette, fondue bourguignonne, tartare Natasha, and in many more occasions.

Besides my supervisor, I would also like to thank the members of my thesis committee, Prof. Francesco Corman, Prof. Michel Gendreau and Dr. Marco Laumanns, and the committee president, Prof. Dimitrios Lignos. Their insightful comments helped me to improve the quality of the thesis from several perspectives. I would also like to thank them for making my private defense an enjoyable moment.

I am extremely thankful towards my colleagues and friends of the TRANSP-OR laboratory. In particular, I would like to thank Jianghang Chen and Yousef Maknoon for their regular supervision during these years. The discussions were sometimes agitated, but always useful. I also want to thank all members of the lab for creating the most amazing, cool, awesome, stimulating, fun and friendly work environment I could ever imagine. Without them, I would not have been able to complete this long journey. I thoroughly enjoyed the coffee breaks, the fruit breaks, the lunch breaks (except Epicure), the beers at Sat, the conference trips, the trips to nowhere (Schwarzwald), the skiing, the hiking, the parties, the laughs and all the crazy moments! Thanks to Bilge Atasoy, Mila Bender, Stefano Bortolomiol, Jianghang Chen, Anne Curchod, Antonin Danalet, Matthieu de Lapparent, Daniel Emery, Anna Fernández Antolín, Aurélie Glerum, Flurin Hänseler, Hainan Hu, Sofia Kalakou, Eva Kazagli, Xinjun Lai, Gael Lederrey, Claudio Leonardi, Virginie Lurkin, Yousef Maknoun, Iliya Markov, Nicolas Molyneaux, Stefano Moret, Marija Nikolic, Nikola Obrenovic, Yuki Oyama, Meritxell Pacheco Paneque, Tomáš Robenek, Marianne Ruegg, Riccardo Scarinci, Shadi Sharif

Azadeh, Nitish Umang, Zhengchao Wang, and to the members of LUTS laboratory for a great working environment and stimulating seminars.
Last but not least, I would like to thank my family in Switzerland and in Austria, and all my friends outside EPFL. I would not have made it without their unconditional support. Best of all, during this thesis I have also met the person I want to spend the rest of my life with: Anna. Thank you for always being here for me, especially during the long last months of the writing process. You constantly reminded me how much there is still to discover after a PhD, and I cannot wait to explore it with you!

Lausanne, December 1st, 2017 Stefan Binder

Abstract

Unexpected disruptions occur for many reasons in railway networks and cause delays, cancellations, and, eventually, passenger inconvenience. This thesis focuses on the railway timetable rescheduling problem from a macroscopic point of view in case of large disruptions, such as track unavailabilities due to, e.g., rolling stock malfunction or adverse weather conditions. Its originality is to consider three objectives when designing the so-called disposition timetable: the passenger satisfaction, the operational cost and the deviation from the undisrupted timetable. These goals are usually incompatible: for instance, the best possible service for the passengers may also be the most expensive option for the railway operator. This inadequacy is the key motivation for this thesis.

The problem is formally defined as a multi-objective Integer Linear Program and solved to optimality on realistic instances. In order to understand the trade-offs between the objectives, the three-dimensional Pareto frontier is approximated using ε-constraints. The results on a Dutch case study indicate that adopting a demand-oriented approach for the management of disruptions not only is possible, but may lead to significant improvement in passenger satisfaction, associated with a low operational cost.

For a more efficient investigation of the multiple dimensions of the problem, a heuristic solution algorithm based on adaptive large neighborhood search is also presented. The timetable is optimized using operators inspired directly from recovery strategies used in practice (such as canceling, delaying or rerouting trains, or scheduling additional trains and buses), and from optimization methods (e.g., feasibility restoration operators). Results on a Swiss case study indicate that the proposed solution approach performs well on large-scale problems, in terms of computational time and solution quality.

In addition, a flexible network loading framework, defining priorities among passengers for the capacitated passenger assignment problem, is introduced. Being efficient and producing stable aggregate passenger satisfaction indicators (such as average travel time), it is used in an iterative manner for the evaluation from the passenger perspective of the timetable provided by the rescheduling meta-heuristic.

The timetable rescheduling problem is a hard problem and this thesis makes significant methodological and practical contributions to the design of passenger-centric disposition timetables. It is the first attempt to explicitly integrate multiple objectives in a single framework for railway timetable rescheduling, as the state-of-the-art usually neglects passenger considerations, or considers them only implicitly. Further, the use of practice-inspired optimization methods allows railway operators to easily implement the results of the proposed framework.

Keywords: railway timetable rescheduling, passenger demand model, capacitated passenger assignment, Pareto frontier approximation, meta-heuristic, integer linear program.

Résumé : Cette thèse porte sur le problème de l'optimisation de l'horaire dans le cas d'une perturbation majeure affectant un réseau ferroviaire. En particulier, elle démontre la pertinence de l'intégration des flux de passagers dans le modèle pour une optimisation judicieuse de l'horaire. Le problème est analysé en détail pour deux cas d'étude, l'un aux Pays-Bas, l'autre en Suisse.

Zusammenfassung: Diese Arbeit befasst sich mit der Optimierung des Fahrplans in Eisenbahnnetzwerken, in denen bedeutende Betriebsstörungen aufgetreten sind. Insbesondere zeigt sie auf, dass die Berücksichtigung von Passagierflüssen massgebend für eine sinnvolle Optimierung des Fahrplans ist. Zwei Fallstudien (in den Niederlanden und in der Schweiz) untersuchen das Problem ausführlich.

Contents

1 Introduction

1.1 Motivation

Mobility has always been an essential requirement of human beings and a necessary condition for a functional economy. However, growing environmental concerns in recent decades have associated transportation with a number of negative externalities. Motorized travel is responsible for a large part of carbon dioxide emissions and therefore contributes significantly to climate change. A massive amount of energy, primarily obtained from fossil fuels, is required to move people and goods on a daily basis, thus depleting non-renewable resources globally. Pollution at the local level raises many health issues due to poor air quality and excessive noise levels. Congestion, accidents, urban sprawl and the deterioration of lanscape quality are also consequences of saturated transportation networks.

Through its unique combination of characteristics (such as high operational speed, reliability, energy efficiency and safety rate), railway as a mode of transportation for passengers and freight can be seen as a solution to some of the problems posed by motorized traffic. Indeed, governments and transport authorities have recognized the shift towards a higher modal split of railways as an efficient and environmentally friendly way to address growing mobility demand.

Nevertheless, the modal share of railway transportation in the European Union has remained at a low and stable level for the past fifteen years, at around 6–7% (see Fig. 1.1).

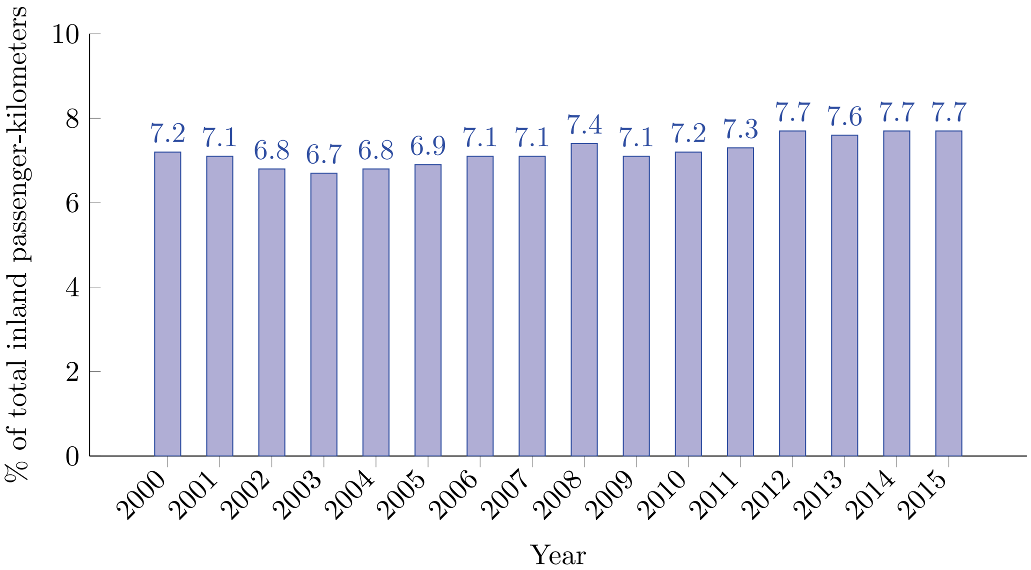


Figure 1.1: Modal split of trains for passenger transport in the European Union (28 countries). Source: Eurostat (2017).

The limited attractiveness of railway transport can be explained by several factors, including the consequences of delays and the (perceived) unreliability of travel times when compared to other modes, such as private transportation. A survey about the satisfaction of European travelers on railway services (European Commission, 2013) reported that only 55% of respondents were “very satisfied” or “fairly satisfied” with the punctuality and reliability of trains. This figure drops to 47% when asked about the availability of travel information, especially during delays. The only two concerns obtaining even lower satisfaction levels are the cleanliness and good maintenance of the rolling stock, and the bicycle access to trains.

The dissatisfaction of railway passengers is understandable when considering disruption statistics. Fig. 1.2 reports the number of railway disruptions, as well as their total annual duration, over the past six years in the Netherlands. A disruption is registered whenever a disruption message is published on the website of the Dutch national railways. Such a message is usually placed when there are no or less trains running on a railway line, or when trains are running with a considerable delay — no disruption message is published when there are only a few trains delayed. The number has steadily risen from about 5 daily disruptions in 2011 to more than 8 daily disruptions in 2016. The average duration of a disruption in this period was 2 hours and 48 minutes. These figures indicate that major disruptions, such as the unavailability of railway tracks due to unexpected events, occur frequently in railway networks, causing train delays and cancellations, and, eventually, deteriorated passenger service. Due to the disruption, passengers using the railway network have longer travel times and reach their destination later than expected. In addition, passengers can experience even larger delays if they miss a connection.

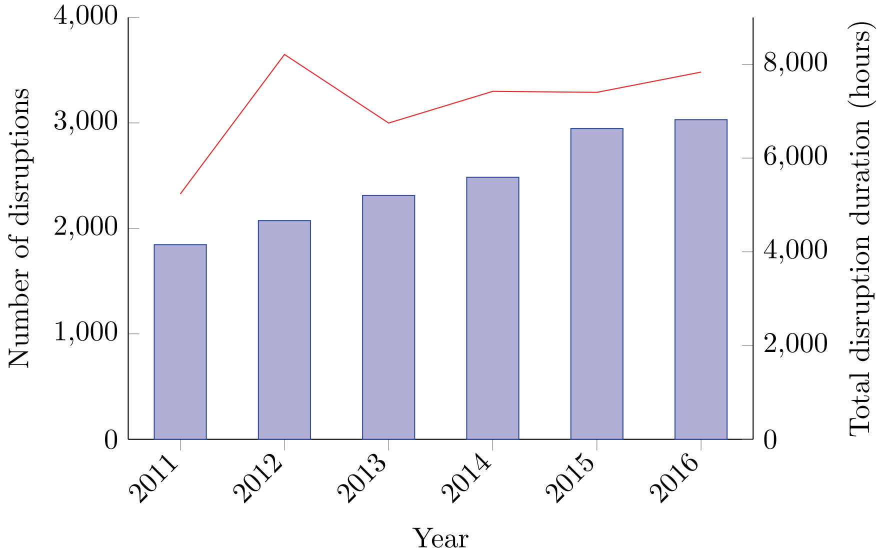


Figure 1.2: Number of disruptions (blue bars) and corresponding total disruption duration (red line) in the Dutch railway network. Source: Rijden de Treinen (2017).

Recently, a legal framework with the aim of providing reliable, frequent and integrated transportation services was created by the European Communities (Directive 2001/14/EC). The directive deregulates the European railway market and translates into a separation of the concerns of the Infrastructure Manager (IM) and the Train Operating Companies (TOCs). The former is a non-profit organization with the duty to provide exploitable railway infrastructures, and to grant non-discriminatory access to the available rail capacity. On the other hand, the latter, who offer transportation services for passengers and freight, operate in a deregulated market environment, and are thus pushed to focus on the level of service provided to customers. Now more than ever, it is crucial for TOCs to outperform the competition in terms of passenger satisfaction, measured with indicators such as frequency of service, punctuality and reliability. Providing an adequate response to disruptions is thus a necessary step in this direction. Also, avoiding the cost of increasingly common "compensation payments" for passengers suffering from large delays (introduced by Directive 2001/14/EC) is a strong incentive for TOCs to minimize the consequences of disruptions.

Corman (2010) notes that a substantial share of responsibility is carried by the IM in increasing the level of transport service offered, as the available railway capacity imposes the main limitations on potential improvement in performance and reliability. Statistics from the Netherlands corroborate that the most common cause of disruptions is "infrastructure": events such as damaged overhead wires, defective switches, power failure, signal failure, damaged railway bridges, etc. accounted for 38% of all disruptions in the past six years (Rijden de Treinen, 2017). However, infrastructure construction and maintenance projects are very expensive and time-consuming. Additionally, Rijden de

Treinen (2017) report that, after "infrastructure", the three most common causes for disruptions were, in order, "rolling stock problems" (22%), "accidents" (13%), and "external influences" (9%), all of which are beyond the control of the IM. More innovative solutions, like technological improvements, efficient planning and management making better use of the existing infrastructure, are therefore needed. Note however that, in many countries (like Switzerland), the IM is not only responsible for infrastructure provision, but also for managing infrastructure allocation and use, in other words, timetabling and traffic management. Despite the TOC's interest in passenger satisfaction, it might therefore be difficult to coordinate with the IM, who is eventually in charge of traffic management.

The reasons outlined above call for a framework that is able to assess quantitatively the effects of disruption management and to evaluate trade-offs between the different stakeholders. In particular, the compromise between passenger satisfaction and operational cost is of special interest to TOCs. Due to its complexity, the disruption management problem is usually broken up into three consecutive phases (Veelenturf et al., 2016): timetable rescheduling, rolling stock rescheduling and crew rescheduling. In the first phase, a so-called disposition timetable is designed. The latter is obtained by applying recovery decisions — such as reordering, retiming, rerouting or canceling — to the timetable operated under "normal" circumstances. The disposition timetable then becomes an input of the second phase, which determines an updated rolling stock allocation (if necessary). Finally, the objective of the last phase is to obtain a feasible crew assignment, given the output of the two previous phases. Solving the three phases individually may lead to globally sub-optimal solutions, in particular if feedback loops are necessary when, given the output of an earlier phase, no feasible solution is found for a later phase of the problem. However, solving the three phases of the recovery problem in an integrated way is a very hard problem.

In this context, this thesis focuses on timetable rescheduling from a macroscopic point of view during major disruptions. Its originality is to consider three objectives to design disposition timetables: passenger satisfaction, operational cost and deviation from the undisrupted timetable. The decision to concentrate on severe disruptions is motivated by the potentially higher level of service improvements and the scarcity of passenger-centric models in the scientific literature. The timetable provided by our framework can then be used as an input for the second and third phases of the disruption management problem: Dollevoet et al. (2017), for instance, introduce and evaluate such an iterative framework. The objectives and the scientific contributions of this thesis are presented in the two following sections.

1.2 Objectives

The overall aim of this thesis is to improve the design of railway disposition timetables in case of major disruptions, by including passenger satisfaction in the decision process. To that end, the problem is addressed from the supply side (railway operator), as well as from the demand side (passengers). Specifically, the objectives of this thesis are the following:

1. Passenger-oriented timetable rescheduling (supply-demand interactions)
    (a) To define a rich yet tractable mathematical model for the practically relevant problem of railway timetable rescheduling.
    (b) To model passenger satisfaction and thereby quantify the goals of the multiple stakeholders of the problem.
    (c) To integrate both sides of the problem by adapting passenger flows to the disposition timetable.
2. Passenger travel choice modeling under saturated conditions (demand)
    (a) To provide an efficient, flexible and behaviorally realistic framework in order to assign passengers on a capacitated public transportation network.
    (b) To identify the effects of several assignment strategies on the level of passenger satisfaction.
3. Practice-inspired timetable rescheduling (supply)
    (a) To introduce a heuristic framework for the rescheduling problem, where the search for timetable optimization is guided by practice-inspired measures.
    (b) To apply the proposed passenger assignment framework, so as to evaluate several competing timetables in terms of passenger satisfaction.

1.3 Scientific contributions

The growing importance of the field has recently attracted the attention of the scientific literature, which is summarized below. Within this literature, the contributions of this thesis are laid out. A detailed review of the relevant literature is provided in Sections 2.2 and 4.2, respectively.

Railway timetable rescheduling The review of Cacchiani et al. (2014) showed that, in the field of railway recovery models, the major part of the recent scientific literature addresses minor disturbances, instead of major disruptions. In that context, most frameworks represent the railway network at a microscopic level. Further, most contributions have an operations-centric approach to railway timetable rescheduling, whereas the literature on passenger-centric formulations is growing but remains sparse. Contributions considering passenger satisfaction in major disruptions, such as Cadarso et al. (2013); Kroon et al. (2015); Corman et al. (2016); Veelenturf et al. (2017), use a heuristic iterative framework to compute passenger flows during disruptions. Also, these papers do not provide a framework to compare the multiple objectives of the problem explicitly. In this context, the contributions of this thesis include:

- the presentation of a multi-objective framework for the railway timetable rescheduling problem, with a special emphasis on minimizing passenger inconvenience;
- the introduction of a meta-heuristic for the efficient investigation of the tridimensional Pareto frontier of the rescheduling problem;
- the usage of practice-inspired operators for timetable rescheduling;
- the application of the models to realistic case studies.

Passenger assignment Most contributions adopt an iterative procedure to obtain an equilibrium solution of the schedule-based capacitated transit assignment problem, complemented with simulation-type network loading. Fundamentally, three types of approaches are used to obtain the passenger's "best path": (i) Poon et al. (2004) apply a time-dependent optimal path algorithm to find the path of lowest generalized cost; (ii) Hamdouch and Lawphongpanich (2008) define the concept of optimal travel strategy as the set of paths with least expected travel cost at every decision-making instant of the trip and solve it by dynamic programming; (iii) Nuzzolo et al. (2012) introduce a route choice model to select from a set of paths with different assignment probabilities. To address the tight capacity constraints explicitly, these models use a first-in-first-out queue at the stations, coupled with the rule that on-board passengers have priority over those wishing to board. This priority rule is endogenous because it depends on the passenger assignment. The endogeneity forces the existing frameworks to embed the passenger priorities implicitly in the network loading process, making them difficult to modify, in the case where one desires to study the impacts of alternative orderings. In this context, the contributions of this thesis include:

- the development of a flexible and efficient network loading model for the capacitated passenger assignment problem;
- the definition of exogenous priority lists ordering the passengers before the assignment, thus extracting the complexity from the assignment;
- the application of the model to a real case study.

1.4 Thesis structure

The remainder of this thesis is structured as described in the following.

Chapter 2 describes the state of the art in railway disruption management, both from the practical and the scientific point of view. A real example of a disruption that occurred on the Swiss railway network is analyzed in detail. The scientific literature associated with timetable rescheduling is reviewed in classified. The chapter also introduces the case studies to which reference is made throughout this thesis.

Chpater 3 presents an Integer Linear Program formulation for the railway timetable rescheduling problem, considering major disruptions from a macroscopic point of view. It takes into account passenger inconvenience, operational cost and deviation from the undisrupted timetable as three objectives to minimize.

This chapter is based on the article:

> Binder, S., Maknoon, Y., and Bierlaire, M. (2017). The multi-objective railway timetable rescheduling problem, Transportation Research Part C: Emerging Technologies 78:78-94.

Earlier versions of this chapter have been presented at the following conferences:

> Binder, S., Maknoon, Y., and Bierlaire, M. (2016). Exploring Pareto solutions of the multi-objective railway disruption management problem. 16th Swiss Transportation Research Conference, May 19, 2016, Ascona, Switzerland.

> Binder, S., Maknoon, Y., and Bierlaire, M. (2016). On the trade-off between passenger convenience and operational costs in railway disruption management. Ninth Triennial Symposium on Transportation Analysis (TRISTAN IX), June 17, 2016, Oranjestad, Aruba.

Chapter 4 introduces a novel schedule-based passenger assignment algorithm that considers vehicle capacities explicitly. Network loading is based on exogenous priority lists defining the order between passengers. By separating the passenger ordering from the actual assignment, the model remains computationally cheap, and can thus be used to evaluate timetables from the passenger perspective in an iterative framework. This chapter improves the procedure used to compute passenger flows in Chapter 3, where simplifying assumptions on passenger behavior are necessary to maintain the model's tractability.

This chapter is based on the article:

> Binder, S., Maknoon, Y., and Bierlaire, M. (2017). Exogenous priority rules for the capacitated passenger assignment problem, Transportation Research Part B: Methodological 105:19-42.

An earlier version of this chapter has been presented at the following conference:

> Binder, S., Maknoon, Y., and Bierlaire, M. (2016), A passenger assignment model for the railway timetable rescheduling problem. 5th symposium of the European Association for Research in Transportation (hEART), September 16, 2016, Delft, The Netherlands.

Chapter 5 introduces an iterative meta-heuristic for railway timetable rescheduling, using the assignment model of Chapter 4 to compute the passenger inconvenience of the disposition timetables.

This chapter is based on the conference paper:

> Binder, S., Maknoon, Y., and Bierlaire, M. (2017). Efficient investigation of multiple dimensions of the railway timetable rescheduling problem. Proceedings of the 17th Swiss Transportation Research Conference May 17-19, 2017.

An earlier version of this chapter has been presented at the following conference:

> Binder, S., and Bierlaire, M., An efficient algorithm for the multi-objective railway timetable rescheduling problem. 6th symposium of the European Association for Research in Transportation (hEART), September 12, 2017, Haifa, Israel.

Chapter 6 gives concluding remarks, establishes the link between this thesis and practice, and determines directions for future research.

2

Context and case studies

This chapter provides insights about the state of the art in railway disruption management, of which two important aspects are discussed. On the one hand there is the state of the art in practice, and on the other, there are the approaches developed in academic research. Caimi et al. (2012) noted that there was still a wide gap between both fields, illustrated by the difficulties faced when putting results from operations research into practice in the context of highly complex systems. Since the publication of this article, the gap has narrowed (thanks to contributions such as Corman and Quaglietta, 2015; Samà et al., 2017), but it still exists.

Both angles are therefore presented in this chapter. Section 2.1 describes the precise course of events during a disruption that occurred on April 4th, 2014, between two major stations of the Swiss railway network. Presenting how a railway operator addressed the situation through a real example shows the high level of complexity of the rescheduling process and points towards the direction in which academic research needs to evolve in order to bridge the aforementioned gap. Section 2.2 reviews the scientific literature related to the train timetable rescheduling problem. A taxonomy of recent contributions is provided, in order to emphasize the contributions of this thesis. Chapter 4 provides a similar review of passenger assignment models. Finally, Section 2.3 concludes the chapter by describing the case studies that will be used throughout this thesis.

2.1 Railway disruption management in practice: a real example

Remark The data of the disruption presented in this section was provided by Rudolf Wampfler, Chief of Operations at the Western Management Centre of the Swiss National Railways (SBB). In the example, we refer to the stations by their abbreviation, as listed in Table 2.1. To help the reader locate the involved stations, the railway map of Switzerland is included in Appendix 2.A and a schematic representation of the network surrounding the accident location can be found in Fig. 2.1.

Table 2.1: Station names and their abbreviations.

Abbreviation	Station name	Abbreviation	Station name
BI	Bienne	LS	Lausanne
BN	Bern	LZ	Luzern
BNAS	Bern Ausserholligen	MS	Münsingen
BNBS	Bern Bümpliz Süd	NEU	Neuchâtel
BNBZ	Bern Bümpliz	NWA	Niederwangen
BUL	Bulle	OL	Olten
FLM	Flamatt	OWA	Oberwangen
FRI	Fribourg	ROM	Romont
GEAP	Genève-Aéroport	SG	St. Gallen
GUE	Gümligen	TH	Thun
KOE	Köniz	THO	Thörishaus
LN	Langnau	THOD	Thörishaus Dorf
LPN	Laupen	YV	Yverdon

At 12:43 on April 4th, 2014, the InterCity train IC720 running between SG and GEAP hit an obstacle next to the station of BNAS, close to Bern. It continued its course and stopped near the following station, BNBS. Due to the accident, the tracks became unavailable for railway traffic between BNAS and BNBS. At the time of the accident, the duration of the blockade was unknown.

In the first minutes after the accident, SBB took the following initial emergency measures:

- search for a replacement train for IC720, starting its journey in LS;
- decision to route passengers traveling from OL to LS (and vice versa) via NEU and YV, instead of via BN and FRI;
- precise location of IC720, and dispatching of a manager to the accident scene;
- search for direct replacement buses running between FRI and BN;

- decision to turn-around or cancel all trains scheduled to pass through the disrupted tracks (see Table 2.2 for details).

At 13:13, SBB had set up the "concept" depicted in Fig. 2.1, summarized in Table 2.2. All long-distance trains (7XX, 25XX and 141XX) are turned around in the closest major station (FRI or BN). Trains 141XX that are supposed to start their journey in BN are completely canceled. Regional trains (151XX and 152XX) turn around in THO if coming from the FRI side, or in BN, if from the other side. The possibility to schedule additional trains was evaluated by SBB, but rejected due to the low level of train occupancy and passenger demand at the time when the disruption occurred. Also, it was impossible to reroute long-distance trains through BI and YV because of rolling stock incompatibility.

Additionally, the following replacement buses were provided:

- For local travelers (and stranded long-distance travelers), a bus service was set up to bridge the gap between THO and BN (5 buses running after 13:35, 7 after 14:30).
- For long-distance travelers, 9 direct buses had left FRI for BN at 14:05, and additional buses were being prepared.
- One of the buses arriving in BN from FRI loaded the waiting passengers and drove back to FRI. The other buses ended their service in BN.

Table 2.2: Infrastructural measures taken by SBB to manage the disruption.

Train group	From, To	Measure	# affected trains
7XX	GEAP–FRI–SG	Turn-around in FRI	3
7XX	SG–FRI–GEAP	Turn-around in BN	3
25XX	GEAP–BN–LZ	Turn-around in FRI	3
25XX	LZ–BN–GEAP	Turn-around in BN	3
141XX	BUL–ROM–FRI–BN	Turn-around in FRI	3
141XX	BN–FRI–ROM–BUL	Cancel	3
151XX	FRI–BN–MS–TH	Turn-around in THO	7
151XX	TH-MS–BN–FRI	Turn-around in BN	7
152XX	LPN–BN–GUE–LN	Turn-around in THO	6
152XX	LN–GUE–BN–LPN	Turn-around in BN	6

At 13:24, the ending time of the disruption was forecast to be 16:00. The forecast was confirmed at 14:54. After the confirmation, SBB began the planning of post-disruption operations. In particular, it had to be decided which train would be the first to cross the disrupted track once reopened and when it would be possible to operate again the "normal" timetable. At 16:02, the tracks were cleared and normal operations could

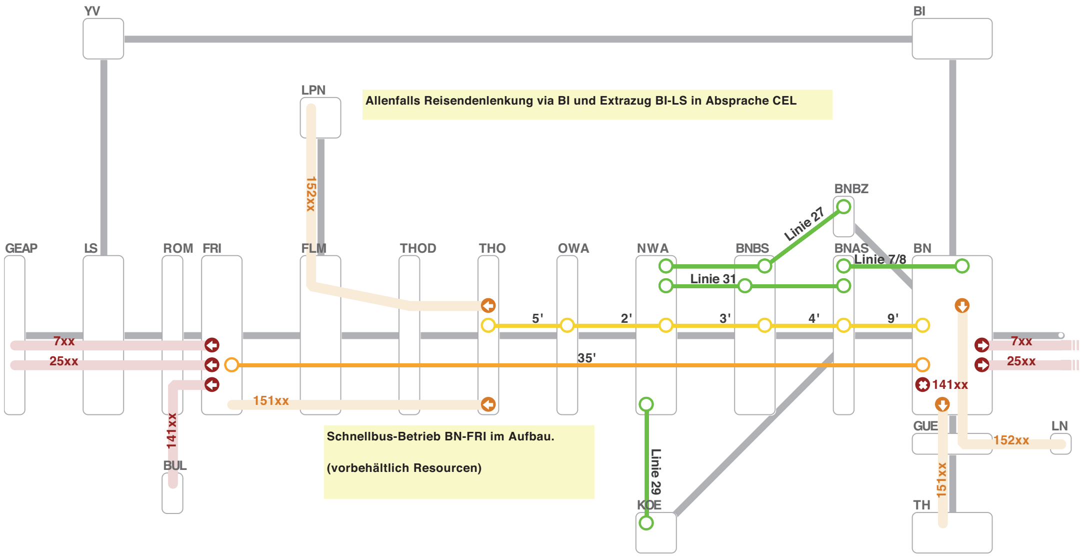


Figure 2.1: Rescheduling "concept" of SBB. The white rectangles with rounded corners represent train stations, the grey lines indicate the railway tracks between the stations, the red and orange arrows show the stations where trains are turned around, the red cross indicates canceled trains and the green lines depict bus lines operating normally. The given travel times are in minutes for buses that are used as train replacements.

resume. SBB expected residual delays of about 5 to 10 minutes for the first trains passing through the reopened area.

Alongside the infrastructural measures described above, SBB performed the following actions in order to manage the disruption. These actions were carried out concurrently with the infrastructural measures and needed proper coordination.

- The damaged train IC720 was evacuated, repaired, cleaned and sent back to BN. The stranded passengers were reallocated on emergency buses and trains.
- Several customer service representatives were dispatched to inform and guide passengers in the major stations (8 in BN, 1 in LS, 2 in FRI, 1 in BNBS, 1 in NWA, 1 in THO).
- Passenger guidance was displayed in the stations in order to allow travelers to adapt their journey as early as possible.
- Passengers with special needs as well as passengers that missed an international connection (or even a plane) had to be considered separately and individual solutions were found.
- Coordination with local bus companies and police was necessary for the smooth operation of the replacement buses in the congested city centers.

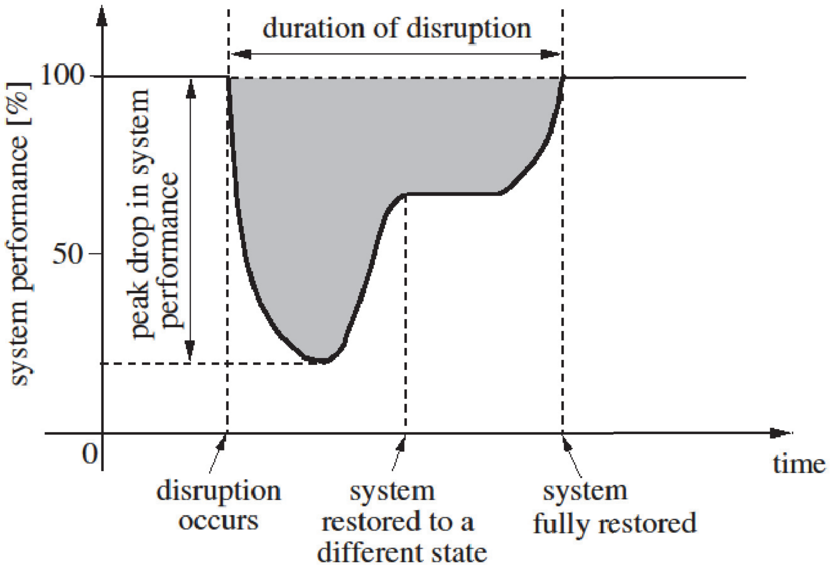


Figure 2.2: Illustration of the system performance during the disruption recovery process. Source: Dorbritz (2012).

Discussion Dorbritz (2012) identifies three stages of the disruption recovery process, illustrated in Fig. 2.2. The first stage corresponds to a peak drop in system performance, just after the occurrence of the disruption. After this initial plunge, the performance stabilizes at a lower level during a transition stage. The duration of this stage can be very long, depending on the time the disruption requires to be solved. Finally, the system returns to its original state after the clearance of the disruption.

The three stages described in Dorbritz (2012) appear clearly in the example. The initial stage, where system performance drops sharply, corresponds to the minutes just after the accident, when emergency measures needed to be taken by SBB. About half an hour later, the intermediate stage is reached: a “concept” was set up and operations were running smoothly, even though the ending time of the disruption was not yet known. The system performance level is lower than in undisrupted conditions, but stable (or gradually improving as the buses are being dispatched). As soon as the ending time of the disruption was confirmed, SBB started to plan the post-disruption operations in order to make the transition back to the initial timetable as short and smooth as possible. This final stage lasted about half an hour, that is, the time for residual delays to be absorbed by the buffer times of the normal timetable.

Despite the high complexity of the network and potential cascade effects (such as rolling stock imbalances or missed connections), SBB used a very straightforward “concept” (Table 2.2) during the intermediate stage: trains are either turned around at the last major station before the disruption, or canceled, and additional buses are scheduled to bridge the gap. Other options (such as train rerouting or the addition of emergency trains) were not implemented, due to rolling stock unavailability/incompatibility, and to the lack of sufficient passenger demand. It is for the intermediate stage that the models presented in Section 2.2 and in this thesis are most valuable. They provide a tool to quantify the effects of existing "concepts" used by the operators, and can help them to choose which one(s) to implement in practice. In addition, the models can suggest (and quantify) new solutions that the operators might not have thought of. By contrast, it is more difficult to directly apply the results of these models in the transition stages of the disruption management problem, when local knowledge and dispatcher experience are critical, especially in the initial time-constrained stage.

2.2 Literature review

The railway timetable rescheduling problem has attracted increasing attention from academic research in recent years. We refer the reader to Cacchiani et al. (2014) for an in-depth review of real-time rescheduling problems and solution approaches. The authors classify publications according to three main criteria, defined in Table 2.3.

Table 2.3: Criteria classifying the railway recovery literature, according to Cacchiani et al. (2014).

Criterium	Description
Disturbance	Primary delay (i.e., a process taking longer than initially scheduled) that can be handled by rescheduling the timetable only, without rescheduling the resource duties (such as crews and rolling stock)
Disruption	(Relatively) large external incident strongly influencing the timetable and requiring resource duties to be rescheduled as well
Microscopic	Very precise representation of the railway infrastructure (sometimes at the switch or track section level), in order to compute detailed running times and blocking times between trains (see, e.g., Hansen and Pachl, 2014)
Macroscopic	High-level representation of a railway infrastructure, considering only stations and tracks (details such as signals or track sections are ignored)
Operations-centric	Focus on minimizing negative effects related to railway companies, such as delays or the number of canceled trains
Passenger-centric	Focus on minimizing negative effects related to passengers, such as total travel time or number of connections

The review of railway recovery models presented in Cacchiani et al. (2014) shows that the major part of the recent scientific literature deals with disturbances rather than disruptions. Further, in most papers, the railway network is represented at the microscopic rather than at the macroscopic level. Most papers also have an operations-centric approach to railway timetable rescheduling, instead of a passenger-centric view.

As this thesis focuses on railway disruption management, we give only a brief overview of publications dealing with disturbances in the following literature review (Section 2.2.1).

Railway disruption management problems are then reviewed in detail, in Sections 2.2.2 and 2.2.3.

2.2.1 Railway disturbance management

Microscopic approach A major part of the recent scientific literature on railway timetable rescheduling has been dedicated to disturbances considered at the microscopic level.

One of the most widely used concepts, the Alternative Graph (AG) model, originally introduced by Mascis and Pacciarelli (2002), has been extensively used in this domain: D'Ariano et al. (2007) present a branch-and-bound algorithm based on an AG model for scheduling trains in real-time and D'Ariano et al. (2008) describe the implementation of the AG model in the Railway traffic Optimization by Means of Alternative graphs (ROMA) tool. In ROMA, the combined problem of train sequencing and routing is approached iteratively: for given train routes, an optimal sequencing is computed using the branch-and-bound algorithm introduced in D'Ariano et al. (2007), and then the solution is improved by local rerouting of trains. The Alternative Graph framework was extensively used thereafter: see, e.g., Corman et al. (2009, 2010a,b, 2012); Samà et al. (2017).

There is however a large amount of additional contributions to the field that propose other microscopic models for the railway timetable rescheduling problem (see, e.g., Mannino and Mascis, 2009; Caimi et al., 2012; Boccia et al., 2013; Lusby et al., 2013; Sato et al., 2013). Also, Toletti and Weidmann (2016) propose a microscopic model for small perturbations, and compare different passenger-centric objective functions; and Toletti et al. (2017) introduce a multi-criteria optimization framework, minimizing train delay, passenger delay and energy consumption. As these publications are only of indirect interest to this thesis, they will not be reviewed here; refer to Cacchiani et al. (2014), and references therein, for a complete overview.

Macroscopic approach The literature discussing railway timetable rescheduling for disturbances on a macroscopic level can be split in two categories: operations-centric and passenger-centric formulations.

Operations-centric contributions to the literature include Törnquist and Persson (2007); Acuna-Agost et al. (2011); Min et al. (2011) and Törnquist Krasemann (2012). Törnquist and Persson (2007) introduce a Mixed-Integer Linear Programming (MILP) model for the rescheduling problem on a N-tracked network. Continuous variables represent start, end and delay times of events, and binary variables express whether an event uses a track, and the order of the trains. The objective function minimizes the final arrival delay of all trains. As some instances could not get solved in reasonable computational time, a greedy heuristic is introduced in Törnquist Krasemann (2012). Acuna-Agost et al. (2011) extend the MILP formulation of Törnquist and Persson (2007) by adapting feasible travel times on tracks according to unplanned stops and by allowing more than one train per track section for trains running in the same direction. Min et al. (2011) propose a MILP formulation to solve the train-conflict resolution problem. Based on the observation that if the timetable is fixed at the segment level, the remaining problem is again a train-conflict resolution problem, a column generation-based method is used to solve the problem. The objective function minimizes the difference to the original timetable.

The literature on passenger-centric formulations is mainly based on the delay management problem (DMP) introduced by Schöbel (2001), which decides if connecting trains should wait for a delayed feeder train or if they should depart on time. The main decision variables of the proposed MIP model, based on an event-activity network, are binary variables deciding if a passenger connection between two trains is maintained or if it is dropped. The objective is to minimize the total passenger delay. The DMP is extended in Schöbel (2009) and Schachtebeck and Schöbel (2010) to include infrastructure capacity constraints, by introducing disjunctive constraints for conflicting train paths. Dollevoet et al. (2012) further extend the DMP to include rerouting of passengers. The routing decisions of the passengers are modeled with binary variables that describe if a connection is used by a passenger on its path from its origin to its destination. Finally, station capacities are considered in another extension of the DMP (Dollevoet, 2013). In Corman et al. (2016), the authors integrate the microscopic representation of railway operations and the passenger perspective of the delay management problem.

2.2.2 Operations-centric railway disruption management

Only very few publications deal with disruptions at the microscopic level. Hirai et al. (2009) formulate the train stop deployment problem (i.e., decisions about where disrupted trains should stop in order to let unobstructed trains pass) after a disruption as a MIP. The main idea is to penalize stops outside stations or deviating from the original schedule. Corman et al. (2011) consider a disruption on a double-track network, where some of the tracks become unavailable. The problem is split into separate dispatching areas, each of which is modeled individually through an Alternative Graph formulation. Finally, boundary constraints between the dispatching areas guarantee that the local solutions are globally feasible. This methodology is compared to a centralized approach, where general dispatching rules are imposed. For both approaches, the authors face increasing difficulty to obtain a feasible timetable for larger time horizons. Xu et al. (2017) present a job-shop scheduling model based on the Alternative Graph formulation. The model integrates traffic management (i.e., train orders and times) as well as train speed management in a single optimization framework. It is tested on a Chinese high-speed railway line with different degrees of speed limitations.

The following papers consider disruptions in railway networks at the macroscopic level.

Brucker et al. (2002) consider the problem of rescheduling trains in case of a track closure due to construction works on a double-track line. A local search heuristic that minimizes lateness is presented and tested on a real world instance.

Narayanaswami and Rangaraj (2013) develop a MILP model that resolves conflicts caused by a disruption that blocks part of a single bidirectional line. Trains can only meet and pass each other in the stations, and cannot be canceled. Train movements are rescheduled in both directions of the line for a small artificial instance, with the objective of minimizing the total delay of all trains.

Albrecht et al. (2013) consider the problem of disruptions due to track maintenance, arising when maintenance operations take longer than scheduled and thus force to cancel additional trains. A disposition timetable including track maintenance is constructed using a Problem Space Search meta-heuristic. The methodology is tested on a single track railway network in Australia.

Corman et al. (2014) compare centralized and distributed procedures for train rescheduling, and propose heuristic algorithms to coordinate dispatching areas. The authors test their algorithms on a Dutch railway network with various traffic disturbances, including delays and blocked tracks.

Louwerse and Huisman (2014) consider partial and complete blockades in case of a major disruption on a double track line. They develop a mixed-integer programming model to generate the disposition timetable. The authors consider two rescheduling possibilities for trains: cancellations and delays. The objective is to minimize both of them. Schedule regularity constraints (e.g., operating approximately the same number of trains in each direction during a partial blockade) are included in the formulation in order to take the rolling stock problem into account implicitly. In case of a complete blockade, both sides of the disruption are considered independently (i.e., trains will reverse before the disrupted area but no coordination with the other side is considered).

Meng and Zhou (2014) consider the N-track simultaneous train rerouting and rescheduling problem. They propose an integer programming model in order to retime, reorder, retrack or reroute trains. They show that the dualized subproblem can be solved rapidly for a medium-size network.

Zhan et al. (2015) consider railway rescheduling on a high-speed line in case of a complete blockade. Due to the nature of the seat reservations, trains that have started their journey have to end in their final destination and cannot be rerouted or canceled. The problem is formulated as a MILP with the same objective function as Louwerse and Huisman (2014). The model is tested on a real-world Chinese case study and is able to reduce the effects of the disruption on passenger service.

Veelenturf et al. (2016) extend the MILP model of Louwerse and Huisman (2014). The extended model is able to deal with a real-world railway network and includes the possibility to retime, reorder, cancel and reroute trains. The paper also considers the transition phases between the undisrupted timetable and the disposition timetable, as well as back to the original timetable when the disruption has ended. The model is tested on a part of the Dutch railway network, and in most cases the computational time is acceptable.

Zhan et al. (2016) reschedule high-speed trains on a double-track line with a partial blockade (unavailability of one track). The authors formulate the rescheduling problem as a MILP and use a rolling horizon approach to solve it. The model is tested on the Beijing-Shanghai line and decides the sequence of trains, the passing times at the stations, as well as canceled services.

2.2.3 Passenger-centric railway disruption management

Cadarso et al. (2013) develop an integrated optimization model for timetable and rolling stock rescheduling that accounts for dynamic passenger demand. The problem is solved in two iterative steps. First, the anticipated disrupted demand is computed using a logit model. As demand figures are estimated before the timetable is adjusted, they are based on line frequencies in an anticipated disposition timetable, rather than on actual arrival and departure times. In the second step, the timetabling and rolling stock rescheduling problem is formulated and solved as a MILP model, subject to the anticipated demand calculated in the first step. Recovery strategies include canceling existing train services or scheduling extra ones. However, the possibility of retiming existing trains is not considered. Computational experiments are performed on the regional rapid transit network of Madrid.

Kroon et al. (2015) present a mathematical model and an iterative heuristic to solve the real-time rolling stock rescheduling problem with dynamic passenger flows. The rescheduled timetable is used as an input in their formulation. The model minimizes a combination of system-related costs (such as penalties for the modification of rolling stock compositions) and service-related costs that express the effect of train capacities on the total passenger delay. Computational results are reported on problem instances constructed from the Netherlands Railways network.

Veelenturf et al. (2017) extend the formulation of Kroon et al. (2015) to integrate the rescheduling of rolling stock and timetable in disruption management. Timetable decisions are limited to additional stops of trains at stations where they normally would not halt. The fact that passengers will adapt their path to the new schedule is taken into account in a heuristic iterative framework: after each generation of a new schedule, passenger flows are simulated to evaluate the service from the passenger's point of view.

2.2.4 Taxonomy of macroscopic railway disruption management

Table 2.4: Characteristics of operations-centric contributions.

Publication	Model structure	Solution algorithm	Objective (min)	Recovery decisions	Largest reported instance size
Brucker et al. (2002)	IP	H (polynomial local search)	Maximal lateness of trains	O	30 hours, 200 trains, double track line
Narayanaswami and Rangaraj (2013)	MIP	C	Total delay of all trains	D	30 minutes, 6 trains, double track line
Albrecht et al. (2013)	S	H (Problem Space Search)	Total, maximal train and maintenance delay	D	50 trains per day, single track line
Corman et al. (2014)	AG	B&B, H (various)	Average and maximal train delays	D, O, R	43 stations, 205 trains, double track network
Louwerse and Huisman (2014)	IP	C	Cancellations, delays, and spatial and temporal imbalances	D, C, TA	12 stations, 2 hours, 16 trains, 2 double track lines
Meng and Zhou (2014)	IP	LR, H (priority rule-based)	Total completion time of all involved trains	D, O, K, R	Up to 40 trains, a network of 287.7 km in length, 85 stations and 97 segments
Zhan et al. (2015)	MIP	C	Cancellations and train delays	D, O, C, A	23 stations, 130 trains, double track line
Veelenturf et al. (2016)	IP	C	Cancellations and train delays	D, O, R, C	39 stations, 60 trains per hour, double track network
Zhan et al. (2016)	MIP	C, R	Cancellations and deviations	D, O, C, A	23 stations, 130 trains, double track line

We provide a taxonomy of the macroscopic models reviewed in Section 2.2.2 and 2.2.3, which allows to understand where this thesis fits in the general context of railway timetable rescheduling. The taxonomy is split into two parts, based on whether a publication considers passengers (Table 2.5), or not (Table 2.4). The following abbreviations are used:

- Model structure: Mixed Integer Program (MIP), Simulation-based model (S), Alternative Graph (AG), Integer Program (IP)
- Solution algorithm: Commercial solver (C), Lagrangian relaxation (LR), Heuristics (H), Branch-and-bound (B&B), Rolling horizon (R), Passenger simulation (PS), Epsilon-constraints (ε)

- Recovery decisions: Order (O), Delay (D), Rerouting (R), Cancellation (C), Emergency train (E), Additional stops (A), Rolling Stock (RS), Retrack (K), Turnaround (TA)
- Passengers: Logit (L), Rerouting (RR), Strategy (S), Shortest path (SP), Capacity (C), User equilibrium (UE), Sytem optimum (SO)

Table 2.5: Characteristics of passenger-centric contributions.

Publication	Model structure	Solution algorithm	Objective (min)	Recovery decisions	Passengers	Largest reported instance size
Cadarso et al. (2013)	MIP	C, PS	Operational cost, cancellations, denied passengers and deviation from original schedule	C, E, RS	L, RR, C	46 stations, 2 hours, 530,000 passengers
Kroon et al. (2015)	IP	C, PS	Penalties for rolling stock modifications and passenger delays	RS	S, SP, UE, RR, C	14 stations, 11,415 passenger groups, 3 hours
Schöbel (2001)	IP	-	Sum of passenger delays	D	SP, SO	-
Schachtebeck and Schöbel (2010)	IP	H (various)	Sum of delays and missed connections	D, O	SP, SO	598 stations, 92 trains, double track network
Schöbel (2009)	IP	H (various)	Sum of delays and missed connections	D	SP, SO	183 stations, 1962 trains
Dollevoet et al. (2012)	IP	H (various)	Total delay of all passengers	D, O	SP, SO, RR	775 passengers, 404 trains
Corman et al. (2016)	MIP	PS, H (various)	Total passenger time in system	D, O	SP, SO, RR	7 stations, 202 trains, double track network
Veelenturf et al. (2017)	IP	PS, H (various)	Rolling stock cost and passenger delays	A	S, SP, UE, RR, C	3 hours, 16 trains, 450,000 passengers
Chapter 3 of this thesis	IP	C, ε	Passenger inconvenience, operational cost and deviation from undisrupted timetable	D, R, C, E	SP, SO, RR, C	11 stations, 24 trains, 55 passenger groups, double track network
Chapter 5 of this thesis	IP	PS, H (ALNS and SA)	Passenger inconvenience, operational cost and deviation from undisrupted timetable	D, R, C, E, TA	SP, UE, RR, C	13 stations, 65 trains, 14,920 passengers, double track network

2.3 Case studies used throughout this thesis

The accident presented in Section 2.1 could not be used as a case study for this thesis due to the lack of available demand data and the difficulty to make assumptions about passenger demand at the national level. Instead, two other case studies with a similar network structure (i.e., double-track network, rerouting possibilities for trains and passengers, ...) were implemented. The first one is inspired from Dollevoet et al. (2012) and represents a heavily used part of the Dutch railway network. Because of its limited size (especially the relatively low number of passenger groups), it should be considered more as an "academic case study" than a realistic one. The second case study is borrowed from Robenek et al. (2016). It models the morning rush hour on the network of regional trains in the canton of Vaud, Switzerland. With about 15,000 individual passengers and 65 trains, it represents a real-world instance.

We recurrently use the concept of *generalized travel time* to model the time a passenger spends in the system. The travel time components (in-vehicle travel time, in-vehicle waiting time, transfers, penalties for early/late departure) are weighted using multiplying factors, as described in Sections 3.2.2, 4.3.1 and 5.2.2. As commonly done in the literature, the weights of the various elements of the generalized travel time are defined relative to the in-vehicle time of the path. We use the values reported in Table 2.6, obtained from the literature. When considering crowding in the assignment model, the multiplying factors introduced in Table 2.7 are used. Note that the passenger assignment depends on the values of these parameters, especially penalties for transfers and early/late departures. With appropriate data, a sensitivity analysis on these values would therefore be very interesting.

Table 2.6: Values of weighting factors in the passengers' generalized travel time.

Parameter	Value	Unit	Reference
β_1	2.5	[min/min]	Wardman (2004)
β_2	10	[min/transfer]	de Keizer et al. (2012)
β_3	0.5	[min/min]	Small (1982)
β_4	1	[min/min]	Small (1982)

Table 2.7: Multiplying factors for crowding function (Wardman and Whelan (2011)).

Load Factor (%)	Seated multipliers	Standing multipliers
50	0.86	-
75	0.95	-
100	1.05	1.62
125	1.16	1.79
150	1.27	1.99
175	1.40	2.20
200	1.55	2.44

2.3.1 Dutch case study

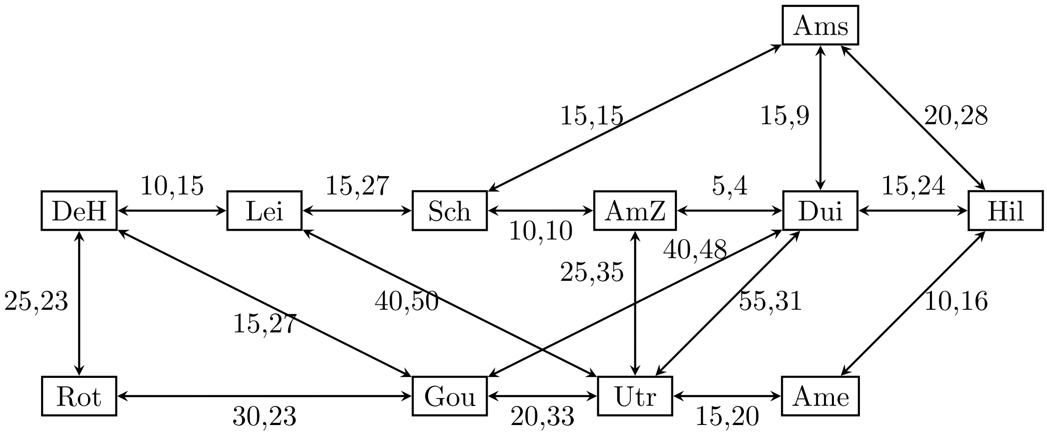


Figure 2.3: Case study network based on a heavily used part of the Dutch railway network. Stations are indicated by rectangles, and tracks between the stations by double-headed arrows. The first number associated with every track is the travel time (in minutes) between the stations; the second is the travel distance (in kilometers).

In the first case study, we consider the railway network indicated in Fig. 2.3. It represents a heavily used part of the Dutch railway network and consists of 11 stations and 18 tracks between the stations. Each station has two platforms and we consider all track sections to be double tracked. The travel times and distances between the stations are obtained from the Netherlands Railways website. They are reported in Fig. 2.3. We assume there are four stations with shunting yards for trains scheduled in the original timetable (Rot, Ams, Sch, AmZ). Five trains are located in each of these shunting yards. Furthermore, we assume there are two shunting yards for emergency trains, located in stations Rot and Ams. Two emergency trains are available in each shunting yard. This gives a total of 24 trains possibly operated in the disrupted case. Every train has a capacity of 400 passengers. The unit cost of operating a train in the Netherlands was not available to us. For the sake of this illustrative case study, we have obtained the value from the Swiss Federal Railways annual report (Swiss Federal Railways, 2015), where a regional service costs 30 CHF per kilometer.

We model passengers travelling home in the evening after work and consider a time horizon of two hours, which is discretized into intervals of five minutes. Within this time horizon, passenger groups are generated according to the procedure described in Appendix 2.B. A "Circos" diagram (Krzywinski et al., 2009) representing the aggregate origin-destination demand of the case study can also be found in this appendix (Fig. 2.B.1).

2.3.2 Swiss case study

For the second case study, we use the network of regional S-trains in canton Vaud, Switzerland, during the morning peak hour. This case study was initially used by Robenek et al. (2016), with the timetable for the year 2014. We recall here its main characteristics for the reader's convenience. The timetable as well as the assumptions regarding passenger demand were updated for the year 2016.

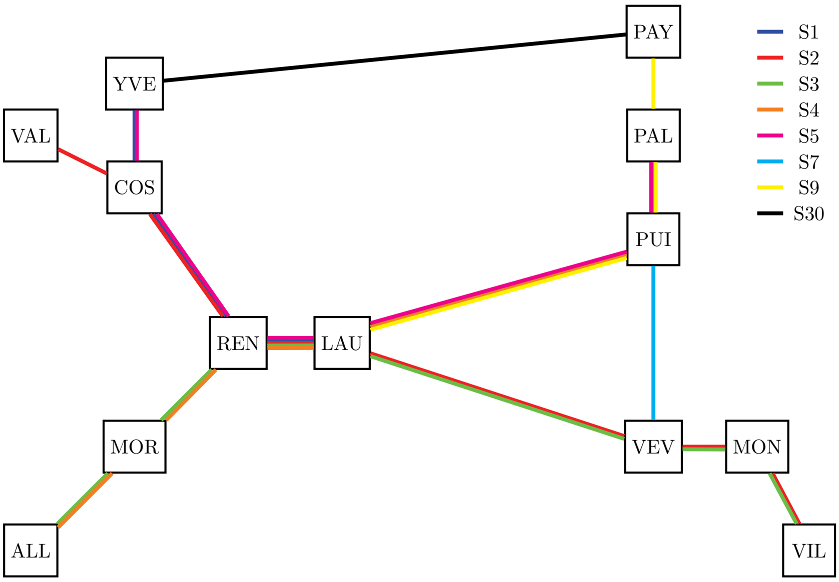


Figure 2.4: Network of S-trains in canton Vaud, Switzerland (2016).

Table 2.8: List of S-train lines in canton Vaud, Switzerland (2016).

Line	From	To	Departure times						
S1	YVE	LAU	05:28	06:28	07:28	08:28			
	LAU	YVE	05:54	06:54	07:54	08:54			
S2	VAL	VIL	05:10	06:10	07:10	08:10			
	VIL	VAL	05:23	06:23	07:23	08:23			
S3	ALL	VIL		06:07	07:07	08.07			
	VIL	ALL	05:49	06:51	07:51	08:51			
S4	ALL	PAL	05:37	06:37	07:37	08:37			
	PAL	ALL		06:34	07:34	08:34			
S5	YVE	PAL	05:57	06:57	07:57	08:57			
	PAL	YVE		06:06	07:07	08:07			
S7	VEV	PUI		06:09	07:09	08:09			
	PUI	VEV		06:36	07:36	08:36			
S9	LAU	PAY	05:25	06:24	07:24	08:24			
	PAY	LAU	05:40	06:40	07:40	08:40			
S30	PAY	YVE	05:30	06:02	06:30	07:02	07:30	08:02	08:30
	YVE	PAY	05:04	06:04	06:33	07:04	07:33	08:04	08:33

The timetable data used in this case study has been downloaded directly from the official website of the Swiss National Railways (SBB), `www.sbb.ch`, for the year 2016. The reduced network of S-trains is presented in Fig. 2.4. We consider the 13 main stations in this network, i.e., LAU, REN, MOR, ALL, COS, VAL, YVE, VEV, MON, VIL, PUI, PAL, PAY. The timetable of the morning peak hours, between 5:00am and 9:00am, is used for this case study. There are 8 bidirectional lines: S1, S2, S3, S4, S5, S7, S9 and S30. We include all trains with a departure time from the beginning of the line between 5:00am and 9:00am. Table 2.8 reports the first and last station of every train line, along with the departure time from the first station of the line. Overall, there are 65 S-trains considered in the morning peak hours. SBB is currently operating Stadler Flirt train units on lines S1, S2, S3, S4 and S5; and Domino train units on the other lines. Stadler Flirt train units have a capacity of 160 seats and 220 standing people, while Domino units can accommodate 188 sitting and 100 standing people.

We consider a deterministic passenger demand, derived from SBB's annual report of 2015 (Swiss Federal Railways, 2015). Not all data required is available, so we rely on realistic assumptions and approximations to generate synthetic passenger data. We consider a total of 14,920 passengers in our case study. The exact procedures and assumptions used to obtain this number can be found in Appendix 2.C. Again, a "Circos" diagram of passenger demand can be found in the corresponding appendix (Fig. 2.C.1).

Appendix

2.A Railway map of Switzerland

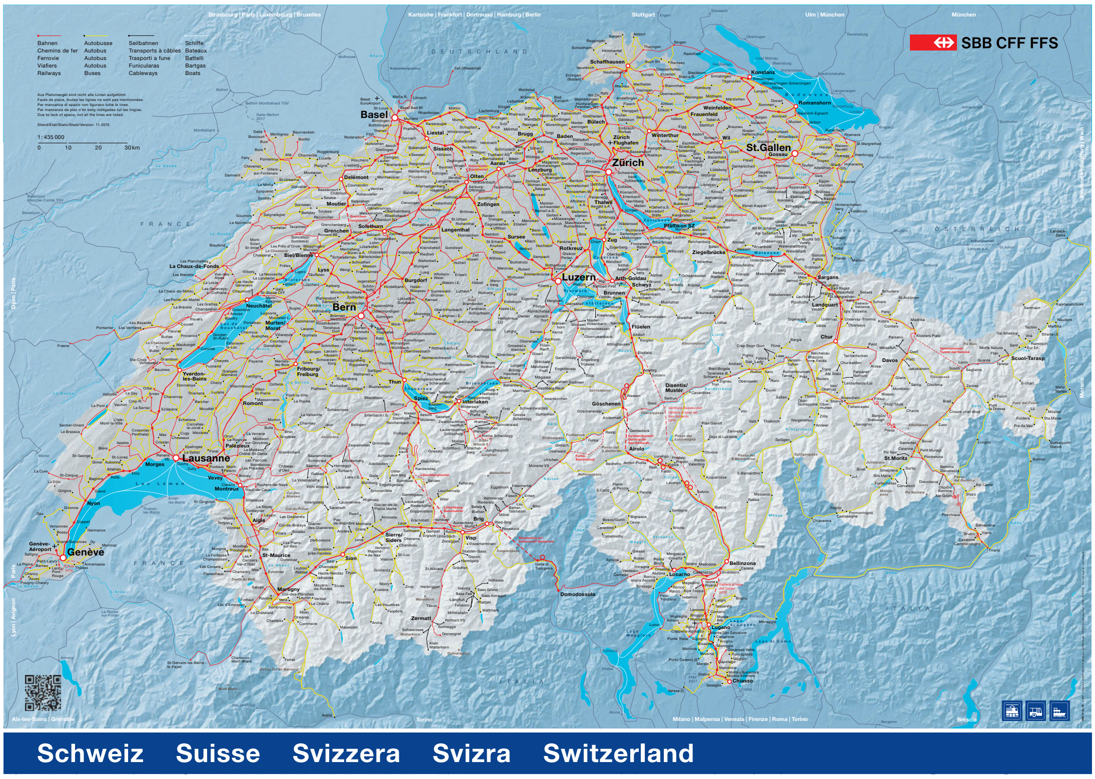


Figure 2.A.1: Railway map of Switzerland (downloaded from `https://www.sbb.ch/content/dam/infrastruktur/trafimage/karten/karte-netzkarte-schweiz.pdf`).

2.B Passenger demand assumptions for Dutch case study

Table 2.B.1: Probability of a station being chosen as a destination.

Station	Probability	Cumulative probability
Rotterdam	0.23	0.23
Gouda	0.03	0.26
Utrecht	0.12	0.38
Amersfoort	0.06	0.43
Den Haag	0.19	0.62
Leiden	0.05	0.67
Schiphol	0.07	0.74
Ams. Zuid	0.07	0.82
Duivendrecht	0.07	0.89
Hilversum	0.03	0.93
Amsterdam	0.07	1.00

To model the demand, we define passenger groups characterized by origin station, destination station and desired departure time from origin. As we model people travelling back home in the evening peak hour, we assume that the probability of a station being a destination of a passenger group is proportional to the number of inhabitants in that city. Table 2.B.1 shows the probabilities of a station being the destination of a passenger group. In the evening peak hour, the probability of choosing a station as a passenger origin should be proportional to the number of jobs located close to this station. This data is however very difficult to obtain or estimate. To simplify, we therefore assume that the probability of being an origin station is uniformly distributed, with the constraint that the origin station should be different from the destination station. The desired departure time is generated using a non-homogeneous Poisson process: we consider an arrival rate of 50 passenger groups per hour in the first hour and 10 passenger groups per hour in the second. We obtain a total of 55 passenger groups, and consider passenger groups with a size of 100 passengers.

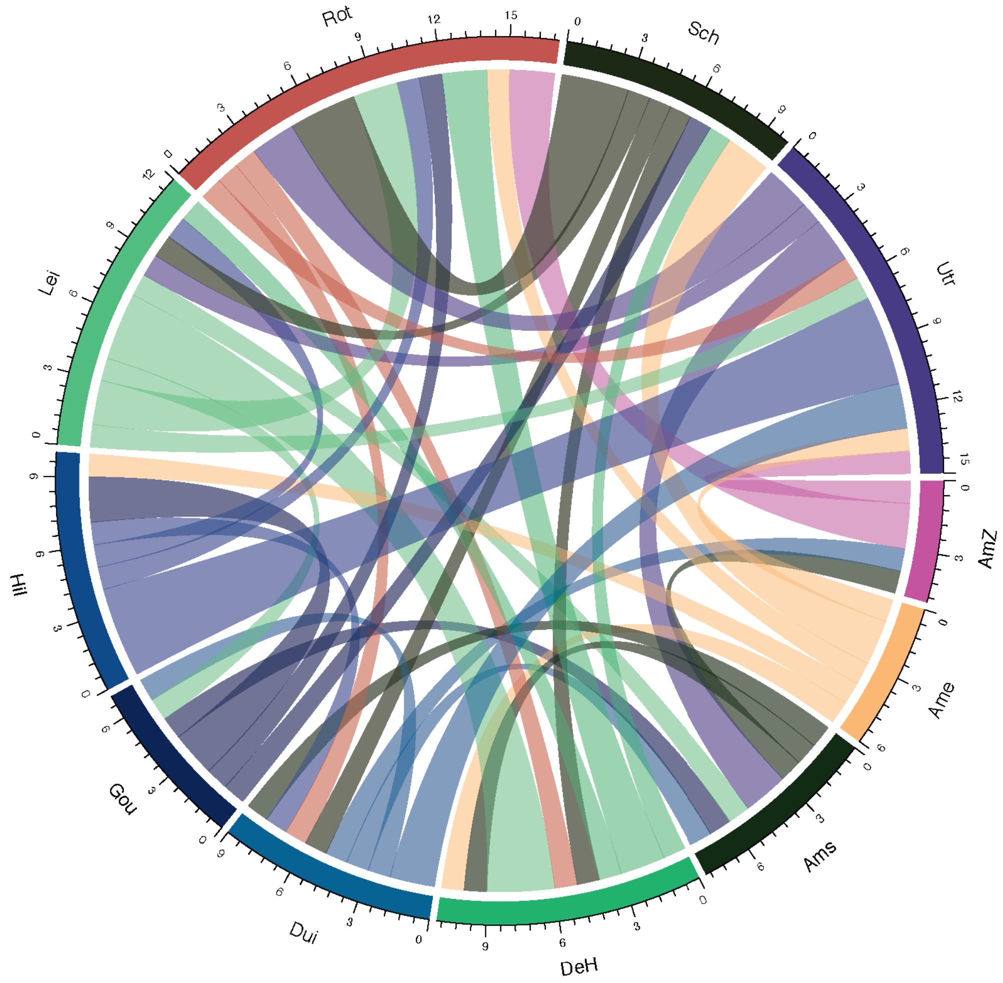


Figure 2.B.1: Passenger demand for the Dutch case study.

2.C Passenger demand assumptions for Swiss case study

The number of passengers in the network has been estimated in the following manner. In 2015, Switzerland had 8,237,666 inhabitants, from which 761,446 lived in Canton Vaud[1], leading to a ratio of roughly 1:10. SBB's annual report (Swiss Federal Railways, 2015) indicates $1.21 \cdot 10^6$ passenger journeys in 2015 for whole Switzerland. By assuming that the growth rate remained the same than the previous years, this gives $1.25 \cdot 10^6$ passenger journeys in 2016. Applying the 1:10 ratio, we arrive to a demand volume of 125,000 daily passenger journeys in Canton Vaud. However, not all these journeys are realized using S-trains. Since almost all of the trains in Canton Vaud have to pass through its capital

[1] https://www.bfs.admin.ch/bfs/fr/home/statistiques/population/effectif-evolution/population.assetdetail.194607.html

city Lausanne, we can derive the ratio between S-trains and other trains passing through Lausanne, of 40-60. This leaves us with 50,000 daily passenger journeys using S-trains in Canton Vaud. SBB's annual report also provides hourly passenger distributions for regional services from Monday to Friday. According to this report, 30% of these journeys are realized between 5:00am and 9:00am, which gives roughly 15,000 daily passenger journeys in the morning peak hours for the S-train network in Canton Vaud.

Table 2.C.1: Hourly distribution of passenger demand, taken from Swiss Federal Railways (2015).

Hours	Percentage	Demand
5:00am - 6:00am	3%	1,500
6:00am - 7:00am	7%	3,500
7:00am - 8:00am	13%	6,500
8:00am - 9:00am	7%	3,500
Morning peak hours	30%	15,000

Table 2.C.1 reports the hourly rates of passenger demand given by the SBB report, as well as the demand obtained for our case study. Finally, this demand has been smoothed into minutes by using a non-homogeneous Poisson process to compute the desired departure time from origin for every passenger. Due to the randomness of the process, the total number of passengers in the network is 14,920.

In order to generate realistic origin-destination flows, we make the following assumptions. Lausanne (LAU) is the biggest city in Canton Vaud and all the lines except S7 and S30 pass through its station (see Fig. 2.4). It is also used by many users to transfer to long-distance trains. We therefore assume it has the largest probability of being a destination, with half of the demand going to this station ($p(D = LAU) = 0.5$). Renens (REN) is assumed to be the city with the second highest probability of being a destination, because it is the closest station to two big Swiss universities, and also based on the network structure (all lines except S7, S9 and S30 stop there). We assume that 20% of the demand travels to Renens ($p(D = REN) = 0.2$) in the morning peak hours. We assume that the rest of the stations have equal probabilities of being a destination: $p(D = i) = 0.3/11, \forall i \in S \backslash \{LAU, REN\}$. This probability is rather small, in accordance with the assumption that most of the morning peak demand is traveling towards work and/or school places, situated in bigger cities such as Lausanne and Renens. Finally, the probability of being an origin station is uniformly distributed among all the stations (except the already selected destination station): $p(O = i) = 1/12, \forall i \in S$.

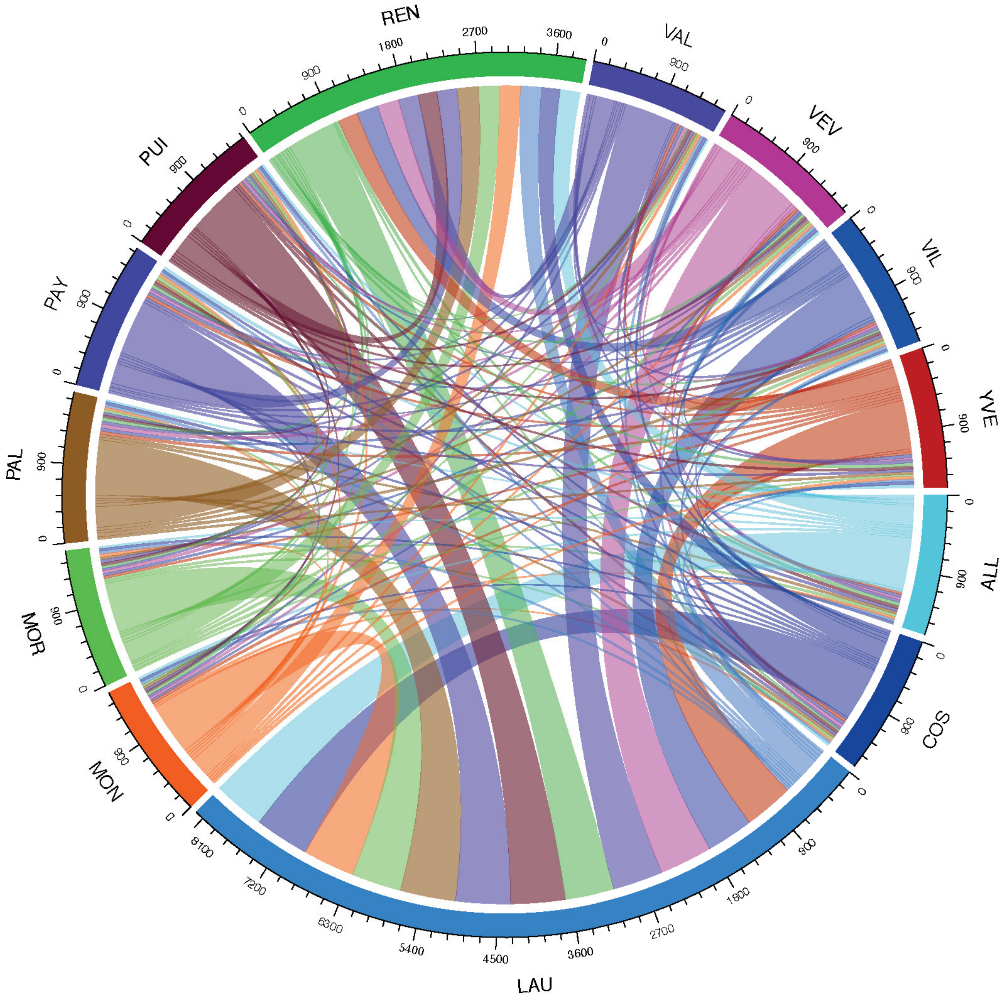


Figure 2.C.1: Passenger demand for the Swiss case study.

3

The multi-objective railway timetable rescheduling problem

This chapter is based on the article:

`Binder, S., Maknoon, Y., and Bierlaire, M. (2017). The multi-objective railway timetable rescheduling problem, Transportation Research Part C: Emerging Technologies 78:78-94.`

The work has been performed by the candidate under the supervision of Dr. Yousef Maknoon and Prof. Michel Bierlaire.

3.1 Introduction

The objective of this chapter is to introduce a new framework for railway timetable rescheduling in case of major disruptions. The framework generates a so-called disposition timetable, which is conflict-free in terms of operational constraints (e.g., no two trains can be scheduled on the same resource at the same time) and as convenient as possible for the passengers.

When constructing a disposition timetable, the objective of the railway operator is to minimize the operational cost, while the aim of the passengers is to receive the best possible level of service. The two goals are usually incompatible: the best possible service for the passengers may also be the most expensive option for the operator. This inadequacy is the key motivation for our work: constructing disposition timetables that take into account passenger satisfaction, while keeping operational costs low. In case of disruptions in railway operations, we also need to take into account the deviation from the undisrupted timetable. Considering this "cost" is necessary in order to avoid solutions where the schedule of the entire network is overhauled because of a local disruption. This is beneficial both for the passengers and the operator. Also, once the disruption is resolved, it is easier and quicker to come back to the undisrupted timetable if the disposition timetable is not too different.

As the literature review in Section 2.2 showed, there is no shortage of operations-centric formulations to solve the railway timetable rescheduling problem. However, passenger-centric formulations remain scarce in the scientific literature. The taxonomy of Section 2.2.4 shows that the framework introduced in this chapter is very comprehensive, especially in terms of possible recovery decisions: most rescheduling measures implemented in other publications appear in our framework. The trade-off for this large number of different rescheduling possibilities is the (relatively) low size of instances that can be solved to optimality. This justifies the use of a rescheduling heuristic, introduced in Chapter 5. In terms of passenger modeling, Cadarso et al. (2013), Kroon et al. (2015), Corman et al. (2016) and Veelenturf et al. (2017) use a heuristic iterative framework to consider passenger flows during disruptions. By contrast, in this chapter, the passenger travel choices are integrated with the timetable rescheduling problem in a single model, which can be solved to optimality. In this sense, our framework can be seen as an integrated version of those contributions. Furthermore, we consider explicitly interactions between demand and supplied capacity, which Cadarso et al. (2013) disregards. The delay management problem, introduced by Schöbel (2001), is not considered explicitly in our work; however, by minimizing the generalized passenger travel time in the objective function we also minimize connection times.

The main contribution of this chapter is to propose an Integer Linear Programming formulation for the multi-objective railway timetable rescheduling problem. It takes into account passenger satisfaction and imposes upper bounds on operational cost and deviation from the undisrupted timetable. To the best of our knowledge, this is the first attempt to integrate three objectives in a single framework for railway timetable rescheduling. The timetables constructed by this approach are therefore results of the trade-off between the conflicting objectives. The exploration of the three-dimensional Pareto frontier allows to analyze this trade-off and to quantify the quality of the timetables according to the three objectives. The contributions of this chapter can be summarized as follows:

- We address the timetable rescheduling problem as a tri-objective problem, with a special emphasis on minimizing passenger inconvenience.

- We propose an Integer Linear Program that integrates both the timetable rescheduling problem and passenger routing.
- We allow a high flexibility for the new timetable and define a measure that quantifies the deviation from the undisrupted timetable.
- We carry out computational experiments on a realistic case study and are able to solve the model to optimality for several instances.

In this chapter, we consider major disruptions causing the unavailability of one (or more) track(s) for a known time period. We therefore look at the rescheduling problem from a macroscopic point of view, disregarding details such as track assignments in stations or signalling. Note that the macroscopic representation of the problem imposes to omit certain details, such as speed profiles and exact train orderings in the vicinity of stations. The real-life feasibility of the solution is therefore not guaranteed. We assume that the elements left out of our representation are solved at a later stage of the recovery process.

The remainder of this chapter is structured as follows. The problem is formally described in Section 3.2 and presented as an Integer Linear Program in Sections 3.3 and 3.4. Section 3.5 reports the results of the computational experiments on the Dutch case study introduced in Section 2.3.1. Finally, Section 3.6 gives some concluding remarks.

3.2 Problem description

In the passenger railway service, a timetable is defined as the set of arrival and departure times of every train at each of the stations where it stops. In this timetable, some trains may run along a predetermined “train line” (i.e., sequence of visited stations). Here, because we consider a disruption scenario, we do not assume any predetermined train line, hence the sequence of visited stations may be different for each train. Given the objectives described in the previous section, our approach determines which trains should be delayed, canceled or rerouted through another part of the network. We also include the possibility of scheduling “emergency trains”, situated in shunting yards near given stations. By solving the model, one obtains a disposition timetable as well as the modified routings of the passengers through the network.

3.2.1 Infrastructural model

Time is discretized into $n + 1$ time intervals of length τ and we introduce the set of time steps $H = \{0, \tau, 2\tau, \ldots, n\tau\}$, where $n\tau$ is the considered planning horizon. We model the railway network at a macroscopic level. The infrastructure is represented by a set of stations $s \in S$ and a set of tracks $Q \subseteq S \times S$ connecting the stations. A track $(s, s') \in Q$ is an uninterrupted railway track linking s to s' directly, without passing in any other station. Each station s is characterized by its available platforms $p \in P_s$ and the presence or absence of a shunting yard. We denote by $S_R \subseteq S$ the subset of stations with a shunting yard, and by R the set of shunting yards. Every shunting yard $r_s \in R$ is associated with exactly one station $s \in S_R$.

We define two stations $s, s' \in S$ to be neighboring if $(s, s') \in Q$ and $(s', s) \in Q$. There are therefore two tracks connecting any two neighboring stations in the undisrupted case. Tracks can be used by trains in both directions: each track can either be used by trains running in the same direction consecutively, or it can be used by trains scheduled in opposite directions alternatively. In any case, a certain headway has to be respected if two trains are scheduled on the same track. Between two neighboring stations, the running time $t(s, s')$, in minutes, and the distance $d(s, s')$, in kilometers, are known, symmetric (i.e., $t(s, s') = t(s', s)$, $d(s, s') = d(s', s)$) and equal for all trains. Trains cannot switch tracks between stations and overtakings occur only within stations (i.e., a platform in a station can be reached from any incoming/outgoing track).

Two different groups of trains are considered: *original trains* and *emergency trains*. The set of original trains K_1 contains the trains that are operated in the undisrupted timetable. Their schedule is an input to the rescheduling model. The set of emergency trains K_2 represents trains that are located in shunting yards, ready to be scheduled if needed. All trains begin and end their trip at a shunting yard and the number of emergency trains available in each shunting yard is given by n_r. We assume that all trains are homogeneous, with the same capacity q, defined as the maximal number of onboard passengers.

3.2.2 Passenger travel choice model

We assume that passengers form groups that share the same origin-destination pair and desired arrival time at destination. As the travel time is deterministic in our modeling framework, the groups can equivalently be characterized by the desired departure time. We adopt the latter representation in the following. A passenger group $g \in G$ is denoted by a triplet (o_g, d_g, t_g), where $o_g \in S$ is the origin station, $d_g \in S$ the destination station, and $t_g \in H$ the desired departure time from the origin. The number of passengers in group g is n_g. The model does not allow splitting of the groups (groups of size one can be considered in order to model passengers individually).

For every passenger group, we consider the set $\Omega(o_g, d_g)$ of all paths linking the origin station o_g to the destination station d_g. A path is a sequence of access, in-vehicle, waiting, transfer and egress movements (refer to Section 3.3 for a definition in terms of arcs in a space-time graph). We associate a utility function with every alternative (i.e., path) and assume that each passenger group chooses the one with the highest utility. The utility function of every alternative $\omega \in \Omega(o_g, d_g)$ for passenger group g depends on the following attributes (similarly to Robenek et al., 2016):

- *In-Vehicle Time* (VT^g_ω): time, in minutes, spent by the passenger group in one (or more) train(s) along the path,
- *Waiting Time* (WT^g_ω): time, in minutes, spent by the passenger group waiting between two consecutive trains at a station along the path (does not consider the waiting time for the first train),
- *Number of Transfers* (NT^g_ω): number of times the passenger group needs to change trains along the path,
- *Early Departure* ($ED^g_\omega = \max(0, t_g - t)$): time difference (in minutes) between the desired (t_g) and the actual (t) departure time from origin for the passenger group, if early,
- *Late Departure* ($LD^g_\omega = \max(0, t - t_g)$): time difference (in minutes) between the actual (t) and the desired (t_g) departure time from origin for the passenger group, if late.

We assume that the price of the trip is equal among all the paths for a given origin-destination pair, so that it does not need to enter the utility function. Based on the aforementioned description for a given passenger group g, the utility of the alternative j is defined as follows:

$$V^g_\omega = -(VT^g_\omega + \beta_1 \cdot WT^g_\omega + \beta_2 \cdot NT^g_\omega + \beta_3 \cdot ED^g_\omega + \beta_4 \cdot LD^g_\omega), \tag{3.1}$$

where $\beta_1, \ldots, \beta_4$ are the relative weights of the attributes described above. This quantity is in minutes and expresses the generalized travel time of passenger group g along path $\omega \in \Omega(o_g, d_g)$, with a negative sign. As commonly done in the literature, the weights of the various elements of the generalized travel time are defined relative to the in-vehicle time of the path. We use the values reported in Table 2.6, obtained from the literature. We assume that passengers have full knowledge of the system and that they choose the path with the highest utility (i.e., the lowest generalized travel time) to travel from origin to destination. This implies that passengers can adapt to a new timetable, for instance by choosing a path that leaves their origin earlier or later, or that passes through another set of stations. Also, our model does not consider demand elasticity, that is, the number of travelers does not change as a consequence of the disruption.

Due to train capacity issues, it is possible that, for some passenger groups, no feasible alternative exists between origin and destination within the time horizon. We therefore include an artificial "opt-out path" for those disrupted passenger groups. This path models the worst possible option to travel from origin to destination. We therefore associate it with the lowest possible utility: the duration of the time horizon, with a negative sign.

3.2.3 Recovery decisions

We consider a disruption in the railway network where a number of tracks become unavailable. Multiple track blockades can occur at the same time, and at different locations in the network. We assume that the network is disrupted for the whole time horizon of the rescheduling problem. Hence, we do not consider what happens after the end of the disruption (i.e., once all tracks can be used again). The fact that one of our objectives is to minimize the deviation from the undisrupted timetable is supposed to facilitate the recovery of the usual timetable.

In order to recover from the disruption, we consider the four following decisions (the three first ones concern original trains):

Cancellation A train may be fully or partially canceled. A partially canceled train is only operated on a subset of the stations of its original route and canceled afterwards. Observe that a full cancellation is a special case of a partial cancellation.

Delay The arrival or departure of a train at a station may be delayed up to a maximal amount of time. A train may also be delayed only for a part of its route. Note that a train with a delay of zero is equivalent to a train in the undisrupted timetable. We do not allow trains to run earlier than in the undisrupted timetable, as this is usually avoided in practice because passengers might miss their planned train.

Rerouting A train may be rerouted through another path than the originally planned one.

Emergency train At every station with a shunting yard, a limited number of emergency trains is available. These may be scheduled as needed.

3.3 Space-time graph

To represent the problem mathematically, we introduce a directed space-time graph $G(V, A)$, inspired from Nguyen et al. (2001). We first describe the sets of nodes and arcs that characterize *all* possible movements in the network in the following section. Based on these sets, we describe in Section 3.3.2 which arcs are available for rescheduling in case of a disruption.

3.3.1 Complete graph

The set of nodes $V = N \cup N_R \cup N_O \cup N_D$ consists of four different types of nodes. A time-expanded node $(s, p, t) \in N$ represents platform $p \in P_s$ of station $s \in S$, at time $t \in H$. N_R is the set of time-invariant shunting yard nodes, associated with shunting yards $r \in R$. Finally, N_O and N_D are the sets of time-invariant origin and destination nodes of the passenger groups. We denote by $s(r), s(o)$ and $s(d)$ the station associated with node $r \in N_R, o \in N_O$ and $d \in N_D$, respectively.

Eight types of arcs are defined, representing all feasible movements of trains and passengers:

- *Starting arcs* model a train leaving the shunting yard at the start of its trip. They are given by the set $A_{Sta} = \{(r, (s, p, t))|r \in N_R, (s, p, t) \in N, s(r) = s, \forall p \in P_s, \forall t \in H\}$.
- *Ending arcs* model a train arriving at the shunting yard at the end of its trip. They are given by the set $A_{End} = \{((s, p, t), r)|(s, p, t) \in N, r \in N_R, s(r) = s, \forall p \in P_s, \forall t \in H\}$.
- *Driving arcs* model the movements of passengers and trains between two neighboring stations. Train driving arcs are given by the set $A_{Dri} = \{((s, p, t), (s', p', t'))| (s, p, t), (s', p', t') \in N, \forall s, s' \in S : (s, s') \in Q, \forall p \in P_s, \forall p' \in P_{s'}, \forall t, t' \in H : t' - t = t(s, s')\}$. For every passenger group g, the set A^g_{Dri} is a duplicate of the arc set A_{Dri} and represents the driving movements of the passenger group. Arcs in A_{Dri} and in A^g_{Dri} are weighted differently, as described in Table 3.1.
- *Waiting arcs* model passengers and trains waiting in a station. Train waiting arcs are given by the set $A_{Wai} = \{((s, p, t), (s, p, t'))|(s, p, t), (s, p, t') \in N, \forall s \in S, \forall p \in P_s, \forall t, t' \in H : t' - t = \tau\}$. Similarly to driving arcs, the set A^g_{Wai} is a duplicate of A_{Wai}, for every passenger group g, and its arcs are weighted accordingly (see Table 3.1). Note that a train (or a passenger) can use consecutive waiting arcs in a station in order to wait longer than τ.
- *Transfer arcs* model passengers transferring from one train to another in a station, with a minimal transfer time m and a maximal transfer time M. They are given by the set $A^g_{Tra} = \{((s, p, t), (s, p', t'))|(s, p, t), (s, p', t') \in N, \forall s \in S, \forall p \in P_s, \forall p' \in P_s \setminus \{p\}, \forall t, t' \in H : m \leq t' - t \leq M\}$.
- *Access arcs* model passenger group g arriving at the origin. They are given by the set $A^g_{Acc} = \{(o, (s, p, t))|o \in N_O, (s, p, t) \in N, s(o) = s = o_g, \forall p \in P_s, \forall t \in H\}$.
- *Egress arcs* model passenger group g leaving the system at destination. They are given by the set $A^g_{Egr} = \{((s, p, t), d)|(s, p, t) \in N, d \in N_D, s(d) = s = d_g, \forall p \in P_s, \forall t \in H\}$.

Table 3.1: Arc weights.

Name	Start node	End node	c_a	t_a^g
Starting	r	(s,p,t)	0	-
Ending	(s,p,t)	r	0	-
Driving	(s,p,t)	(s',p',t')	$c \cdot d(s,s')$	$t(s,s')$
Waiting	(s,p,t)	(s,p,t')	0	$\beta_1 \cdot (t'-t)$
Transfer	(s,p,t)	(s,p',t')	-	$\beta_2 + (t'-t)$
Access	o	(s,p,t)	-	$\beta_3 \cdot \max(0,(t_g - t)) +$ $\beta_4 \cdot \max(0,(t - t_g))$
Egress	(s,p,t)	d	-	0
Opt-out	o	d	-	$n\tau$

- *Opt-out arcs* model passenger group g not succeeding to take the train from origin to destination. They are given by the set $A^g_{Opt} = \{(o,d) | o \in N_O, d \in N_D, s(o) = o_g, s(d) = d_g\}$.

The set of train arcs is given by $A_T = A_{Sta} \cup A_{End} \cup A_{Dri} \cup A_{Wai}$, while the set of passenger arcs associated with passenger group g is $A^g = A^g_{Dri} \cup A^g_{Wai} \cup A^g_{Tra} \cup A^g_{Acc} \cup A^g_{Egr} \cup A^g_{Opt}$. Note that driving and waiting arcs describe both train and passenger movements. The nodes associated with passenger group g are denoted by N^g.

The cost of using an arc $a \in A_T$ for a train (c_a), or an arc $a \in A^g$ for a passenger group g (t_a^g), are listed in Table 3.1. We assume that the operational cost is proportional to the distance travelled by the trains. Therefore, only driving arcs have an operational cost c_a different from zero (c is the cost of running a train, per kilometer). Passenger arcs are weighted according to the utility function introduced in Section 3.2.2. The cost of a path in the graph for a passenger group is obtained by summing up the weights t_a^g of the arcs in the path. Note that a driving or waiting arc is weighted differently if is used by a train or a passenger.

3.3.2 Rescheduling graph

The arcs introduced in the previous section represent *all* possible train and passenger movements in the network. In case of a disruption, some of these movements are forbidden. Also, we need to distinguish train arcs available to original trains and to emergency trains. The general features of the procedure used to generate the rescheduling graph $G(V, A^*)$ are explained here. The interested reader can refer to Appendix 3.A for the detailed algorithm.

When a track becomes unavailable in the network, it cannot be used by any train. The driving arcs in A_T corresponding to this track are therefore removed from the graph.

We denote by $A^* \subset A_T$ the subset of available train arcs in the disrupted graph.

The timetable of the original trains is an input to the rescheduling problem. Based on this timetable, we construct, for every original train $k \in K_1$, the set of *available arcs*, A_k, as a subset of the disrupted train arc set, A^*. An original train can be canceled, delayed or rerouted. Cancellation is modeled by introducing additional decision variables to the problem (see Section 3.4). As detailed in Appendix 3.A, the sets A_k^D and A_k^{RR} are introduced to model delays and reroutings. For every original train $k \in K_1$, only arcs in $A_k = A_k^D \cup A_k^{RR}$ can be used during the disruption. Similarly, we define the set of *available nodes*, N_k, as the set of head nodes of all arcs in A_k.

By contrast, emergency trains do not have a previous timetable and can therefore be scheduled at any time. The set of *available arcs* is simply the disrupted train arc set A^* in that case. The only constraints for emergency trains is to start and end their trip at a shunting yard and not to use tracks conflicting with other trains. To prevent conflicts, we consider the two following situations:

- In the case where trains run consecutively in the same direction on a track, we assume that a separation of τ minutes between them is sufficient. Hence, there is no need to include additional constraints in this case.
- In the case where a track (s, s') is used alternatively in opposite directions by two trains, it needs to be ensured that the track is vacated by the first train before the second one can use it. For every train driving arc $a = ((s, p, t), (s', p', t')) \in A_{Dri}$, we define the set of conflicting arcs $\Theta(a) = \{((s_1, p_1, t_1), (s_1', p_1', t_1')) \in A_{Dri} | s = s_1', s' = s_1, p = p_1, p' = p_1', t_1 \geq t, t_1 < t' + \tau\} \cup \{a\}$. This set formalizes the fact that if a train goes from station s to s', from time t to t', there can be no other train in the opposite direction until time $t' + \tau$.

3.4 ILP Formulation

In this section, we present an Integer Linear Programming formulation for the multi-objective railway timetable rescheduling problem. We consider a tri-objective optimization problem: the quantities to minimize are (i) passenger inconvenience, (ii) operational cost, and (iii) deviation from the undisrupted timetable. The multi-objective aspect of the problem is addressed by using ε-constraints.

Based on the graph defined in the previous section, we introduce the following binary decision variables:

- $w_a^g = \begin{cases} 1 & \text{if passenger group } g \in G \text{ uses arc } a \in A^g, \\ 0 & \text{otherwise} \end{cases}$

- $x_a^k = \begin{cases} 1 & \text{if original train } k \in K_1 \text{ uses arc } a \in A_k, \\ 0 & \text{otherwise} \end{cases}$
- $y_a = \begin{cases} 1 & \text{if an emergency train uses arc } a \in A^*, \\ 0 & \text{otherwise} \end{cases}$
- $z_i^k = \begin{cases} 1 & \text{if original train } k \in K_1 \text{ is canceled after node } i \in N_k, \\ 0 & \text{otherwise} \end{cases}$

Table 3.2 summarizes all the notations used in the model for the reader's convenience. The last column indicates where the notion is further explained, if necessary.

Table 3.2: Notations used in the ILP model.

Name	Description	Reference
G	Set of passenger groups	3.2.2
A^g	Set of arcs associated with passenger group $g \in G$	3.3.1
N^g	Set of nodes associated with passenger group $g \in G$	3.3.1
N	Set of time-expanded nodes	3.3.1
N_R	Set of shunting yard nodes	3.3.1
S_R	Set of stations with a shunting yard	3.2.1
$A_i^{g+} \subset A^g$	Set of passenger arcs leaving node $i \in N \cup N_O$	
$A_i^{g-} \subset A^g$	Set of passenger arcs entering node $i \in N \cup N_D$	
A_T	Set of undisrupted train arcs	3.3.1
$A^* \subset A_T$	Set of disrupted train arcs	3.3.2
$A_i^+ \subset A^*$	Set of train arcs leaving node $i \in N \cup N_R$	
$A_i^- \subset A^*$	Set of train arcs entering node $i \in N \cup N_R$	
$\Theta(a) \subset A^*$	Set of train arcs conflicting with arc $a \in A^*$	3.3.2
K_1	Set of original trains	3.2.1
$A_k \subset A^*$	Set of available arcs of original train $k \in K_1$	3.3.2
N_k	Set of available nodes of original train $k \in K_1$	3.3.2
$A_k^D \subset A_k$	Set of available delay arcs of original train $k \in K_1$	3.3.2, 3.A
$A_k^{RR} \subset A_k$	Set of available rerouting arcs of original train $k \in K_1$	3.3.2, 3.A
$A_{k,i}^+ \subset A_k$	Set of available arcs of train $k \in K_1$ leaving node $i \in N \cup N_R$	
$A_{k,i}^- \subset A_k$	Set of available arcs of train $k \in K_1$ entering node $i \in N \cup N_R$	
t_a^g	Weight of arc $a \in A^g$ used by passenger group $g \in G$	3.3.1
n_g	Size of passenger group $g \in G$	3.2.2
c_a	Cost of running a train on arc $a \in A_T$	3.3.1
r_k	Shunting yard node where train $k \in K_1$ begins its trip	
q	Passenger capacity of a train	3.2.1
n_r	Number of emergency trains available in depot $r \in R$	3.2.1
t_i^k	Time difference between node $i \in N_k$ and the original arrival time of train k at its last station	
t_a	Time duration of arc $a \in A_k^{RR}$	
d_a^k	Delay of arc $a \in A_k^D$, compared to the original timetable of train k	
c_e	Cost of starting an emergency train	

3.4.1 Objective functions

The three objective functions are defined as follows (Eqs. (3.2)–(3.4)). As detailed in Section 3.2.2, *passenger inconvenience* (z_p) is given by the generalized travel time of the passengers. The *operational cost* of the timetable (z_o) is the running cost of original trains as well as emergency trains. The *deviation cost* (z_d) represents the deviation from the undisrupted timetable and is a weighted sum of the different rescheduling possibilities: cancellations, reroutings, delays and the cost of adding an emergency train (the respective weighting factors are $\delta_c, \delta_r, \delta_d, \delta_e$).

$$z_p = \sum_{g \in G} \sum_{a \in A^g} t_a^g \cdot n_g \cdot w_a^g \tag{3.2}$$

$$z_o = \sum_{a \in A^*} c_a \cdot y_a + \sum_{k \in K_1} \sum_{a \in A_k} c_a \cdot x_a^k \tag{3.3}$$

$$\begin{aligned} z_d = {} & \delta_c \sum_{k \in K_1} \sum_{i \in N_k} t_i^k \cdot z_i^k + \delta_r \sum_{k \in K_1} \sum_{a \in A_k^{RR}} t_a \cdot x_a^k \\ & + \delta_d \sum_{k \in K_1} \sum_{a \in A_k^D} d_a^k \cdot x_a^k + \delta_e \sum_{r \in R} \sum_{a \in A_r^+} y_a \end{aligned} \tag{3.4}$$

3.4.2 Constraints

The model we propose has three types of constraints: operational constraints, passenger routing constraints and ε-constraints.

Operational constraints Constraints of the first type ensure that all train movements are operationally feasible.

$$\sum_{a \in A_{k,r_k}^+} x_a^k = 1, \qquad \forall k \in K_1, \tag{3.5}$$

$$\sum_{a \in A_{k,i}^-} x_a^k = \sum_{a \in A_{k,i}^+} x_a^k + z_i^k, \qquad \forall i \in N_k, \forall k \in K_1, \tag{3.6}$$

$$z_i^k = 0, \qquad \forall i = (s,p,t) \in N | s \notin S_R, \tag{3.7}$$

$$\sum_{a \in A_r^+} y_a \leq n_r, \qquad \forall r \in N_R, \tag{3.8}$$

$$\sum_{a \in A_i^-} y_a = \sum_{a \in A_i^+} y_a, \qquad \forall i \in N, \tag{3.9}$$

$$\sum_{a \in A_i^-} (y_a + \sum_{k \in K_1} x_a^k) \leq 1, \qquad \forall i \in N, \tag{3.10}$$

$$\sum_{a'\in\Theta(a)} (y_{a'} + \sum_{k\in K_1} x^k_{a'}) \leq 1, \qquad \forall a \in A^*. \tag{3.11}$$

Constraints (3.5) ensure that the original trains leave their shunting yard at the beginning of their trip. Flow conservation constraints (3.6) make sure that a train either continues its trip after node $i \in N_k$, or it is canceled. Constraints (3.7) forbid cancellations in stations where there is no shunting yard available. The movements of emergency trains are governed by constraints (3.8) and (3.9): there cannot be more trains leaving a shunting yard than the number of trains available in this shunting yard and flow is conserved at every node i. Operational conflicts are avoided with constraints (3.10) and (3.11). The former ensure that for every node i, there is at most one incoming train (emergency or original). $\Theta(a)$ is the set of conflicting arcs of arc $a \in A_T$, i.e., if a train is scheduled on arc a, there can be no train scheduled on any arc of $\Theta(a)$.

Passenger routing constraints Constraints of the second type deal with the routing of the passengers and are presented below.

$$\sum_{a\in A^{g+}_{o_g}} w^g_a = 1, \qquad \forall g \in G, \tag{3.12}$$

$$\sum_{a\in A^{g-}_{d_g}} w^g_a = 1, \qquad \forall g \in G, \tag{3.13}$$

$$\sum_{a\in A^{g-}_i} w^g_a = \sum_{a\in A^{g+}_i} w^g_a, \qquad \forall g \in G, \forall i \in N^g, \tag{3.14}$$

$$w^g_a \leq y_a + \sum_{k\in K_1} x^k_a, \qquad \forall g \in G, \forall a \in A^* \cap A^g, \tag{3.15}$$

$$\sum_{g\in G} n_g{\cdot}w^g_a \leq q(y_a + \sum_{k\in K_1} x^k_a), \qquad \forall a \in A^* \cap A^g. \tag{3.16}$$

Constraints (3.12)–(3.14) are flow conservation constraints for every passenger group: passengers have to leave their origin, arrive at destination and use an uninterrupted path in their respective network. Constraints (3.15) and (3.16) link the passenger paths to the train paths and ensure that passengers only use arcs where a train is available, and that train capacities are not exceeded. Note that constraints (3.15) are redundant, but might be useful as valid inequalities when solving the LP relaxation.

ε-constraints To address the multi-objective aspect of the problem, we introduce ε-constraints (see, e.g., Ngatchou et al. (2005)). Any of the three objectives can be included as a constraint: $z_i \leq \varepsilon_i, i \in \{p, o, d\}$. The order in which objectives are minimized as well as the selection process for the upper bounds ε_i is detailed in the next section.

Another common approach in multi-objective optimization is to combine the objectives into a single objective function using weights. The drawback of this approach is that setting the weights is arbitrary and often complicated (especially if the objectives are expressed in different units). The use of explicit constraints provides interpretations that the weighted sum of objectives does not.

3.4.3 Pareto frontier

The goal of the methodology presented in this section is to explore the three-dimensional Pareto frontier of the problem in an easily interpretable way. By doing so, trade-offs between the three objectives can be quantified in a meaningful way. The generation of the whole exact three-dimensional Pareto frontier (see, e.g., Mavrotas (2009); Charkhgard (2016)) is however out of the scope of this work. Note in particular that the following procedure does not guarantee to find all non-dominated solutions.

Table 3.3: Main characteristics of the five-step methodology to explore the Pareto frontier.

Step	Objective	Constraints	Optimal value	Network
(0)	$\min z_p$	(3.8)–(3.16) $K_1 = \varnothing$	z_p^*	$G(V, A)$
(0b)	$\min z_o$	(3.8)–(3.16) $K_1 = \varnothing$ $z_p = z_p^*$	z_o^*	$G(V, A)$
(1)	$\min z_p$	(3.5)–(3.16)	$z_p^{**} \geq z_p^*$	$G(V, A^*)$
(2)	$\min z_d$	(3.5)–(3.16) $z_p \leq \varepsilon_p, \quad \varepsilon_p = \varepsilon \cdot z_p^{**},$ $\varepsilon \in \{1.0, 1.1, \ldots, 2.0\}$	z_d^{**}	$G(V, A^*)$
(3)	$\min z_o$	(3.5)–(3.16) $z_p \leq \varepsilon_p, \quad \varepsilon_p = \varepsilon \cdot z_p^{**},$ $z_d \leq \varepsilon_d, \quad \varepsilon_d = \varepsilon \cdot z_d^{**},$ $\varepsilon \in \{1.0, 1.1, \ldots, 2.0\}$		$G(V, A^*)$

In order to construct the Pareto frontier, we minimize the objectives in the following order: z_p, then z_d (with an upper bound on z_p), and finally z_o (with upper bounds on z_p and z_d). Choosing passenger inconvenience as the first objective to minimize seems natural for our passenger-centric formulation. Minimizing the deviation from the undisrupted timetable second is motivated by several computational experiments that showed that the trade-off can be best evaluated if the deviation cost is minimized before the operational cost. Also, in an extreme situation such as a disruption, the sensitivity to operational cost of the railway operator should be lower than during regular operations. Indeed, the public would be outraged if cost minimization appeared to be the main priority of the operator during the disruption. Further, emphasizing to minimize the deviation from the undisrupted timetable helps to recover the timetable at a later stage of the rescheduling process. These reasons justify the order we chose for the optimization framework. Clearly, the proposed framework is general, and can accommodate different orderings, that can be justified from the specific context. The following five-step methodology, summarized in Table 3.3, is used to explore the Pareto frontier.

The rescheduling problem takes an undisrupted timetable as an input. This timetable needs to be optimal with respect to the objectives we define, so as to have a benchmark — otherwise, the comparison would be unfair. Before solving the problem on the disrupted network $G(V, A^*)$, we therefore solve the problem on the undisrupted network $G(V, A)$, without any original trains, with passenger inconvenience as the objective to minimize. The passenger inconvenience obtained by the first step, z_p^*, might however be achieved with a lower operational cost, as there is no constraint on the latter. Thus, the next step minimizes the operational cost, while enforcing passenger inconvenience to be equal to the optimal value of the first step. These two steps (denoted by (0) and (0b) in Table 3.3) thus generate a timetable that is optimal in terms of passenger inconvenience for the undisrupted network, and is associated with the minimal operational cost for that level of inconvenience.

The next step is the first one to be applied on the disrupted network (step (1) of Table 3.3). As described above, we begin by minimizing the passenger inconvenience, without constraints on operational and deviation cost. This gives an optimal value for the disrupted case, in terms of passenger inconvenience, z_p^{**} (that is obviously higher than z_p^*). The objective value of step (1) is then used as an upper bound in the constraints of step (2). The problem is solved with the objective of minimizing z_d, under the constraint $z_p \leq \varepsilon_p, \varepsilon_p = \varepsilon \cdot z_p^{**}$. For every value of $\varepsilon \in \{1.0, 1.1, \ldots, 2.0\}$, the problem is solved and the value of the optimal deviation cost z_d^{**} is obtained. This allows to explore the trade-off when the passenger inconvenience varies in equally spaced intervals, from the best possible solution (z_p^{**}) to a solution with twice the passenger inconvenience. Finally, in the step (3), operational cost is minimized with upper bounds on passenger inconvenience and deviation cost, obtained from steps (1) and (2): $z_p \leq \varepsilon_p, \varepsilon_p = \varepsilon \cdot z_p^{**}$ and $z_d \leq \varepsilon_d, \varepsilon_d = \varepsilon \cdot z_d^{**}$. The problem is solved for every feasible combination of ε_p and ε_d.

The mathematical model presented in this work is an adaptation from the minimum cost flow problem. Note therefore that imposing integrality of the flows is not necessary. Even when solving the relaxation, the underlying minimum cost flow problem results in an integral flow (although the passenger groups might be split). The problem's complexity comes from the routing of the passengers through the network. As the weights of the passenger arcs depend on the passenger group, there needs to be one decision variable per arc and per passenger group. Thus, every additional passenger significantly increases the number of decision variables in the model.

3.5 Numerical experiments

3.5.1 Case study description

We illustrate the methodology on the network of the Dutch case study described in Section 2.3.1. The mathematical model is solved by CPLEX 12.5 on a Unix server with 8 cores of 3.33 GHz and 62 GiB RAM. Most instances are solved to optimality (with a gap of 0.01%) in the time limit of one hour.

The generalized travel time of the passengers is computed by using the weights given in Table 2.6 for the passenger arc costs t^g_a. The cost of the opt-out arc is the time horizon (two hours). We impose a minimal transfer time m of 5 minutes and a maximal transfer time M of 30 minutes. Given these inputs, Table 3.4 indicates the cardinalities of the sets of the space-time graph obtained by following the methodology described in Section 3.3.

Table 3.4: Cardinalities of the space-time graph.

Set	Cardinality	Set	Cardinality	Set	Cardinality
N	550	A_{Sta}	200	$\cup_{g\in G}A^g_{Dri}$	49,652
N_R	4	A_{End}	200	$\cup_{g\in G}A^g_{Wai}$	12,076
N_O	11	A_{Dri}	2,824	$\cup_{g\in G}A^g_{Tra}$	22,622
N_D	11	A_{Wai}	528	$\cup_{g\in G}A^g_{Acc}$	1,948
				$\cup_{g\in G}A^g_{Egr}$	1,948
				$\cup_{g\in G}A^g_{Opt}$	55

Two disruption scenarios are created. For the first disruption scenario, two random tracks are fully blocked. We choose to block the tracks between station Gouda (Gou) and Utrecht (Utr) in both directions. For the second one, we consider the worst-case scenario. To do so, we assume that the most heavily used tracks of the undisrupted timetable are blocked. These turn out to be the tracks between Den Haag (DeH) and Leiden (Lei), and between Leiden (Lei) and Schiphol (Sch). We create the second disruption instance by considering a full blockade of the tracks between theses stations. For both disruption scenarios, we assume the tracks are blocked for the whole time horizon of two hours.

To compute the deviation cost from the undisrupted timetable, we need four different weights: penalties for canceling, delaying and rerouting trains, and the cost of operating emergency trains. We use the same values as in Veelenturf et al. (2016), where the aim is to operate as many original trains as possible. The canceled time of a train, i.e., the time difference between the time when a train is supposed to reach its last station in the undisrupted timetable, and the time after which it is canceled in the disrupted timetable, is weighted heavily by a factor of $\delta_c = 50$. Every delayed minute, for each train departure, is weighted only by a factor of $\delta_d = 1$. The time a rerouted train spends on a different geographical path than the original one (i.e., on arcs in A_k^{RR}) is weighted by $\delta_r = 10$. These values ensure that rerouting a train is preferred over canceling it, while delaying is the least "costly" option. Finally, the penalty to operate an emergency train δ_e is 1,000 plus the operated time of the emergency train. The maximal allowed delay per train M_D is set to 30 minutes. It is therefore assumed that a train is canceled if it needs to be delayed by more than 30 minutes. The maximal delay value can be adjusted according to the operator's practice.

3.5.2 Results

Before applying the rescheduling framework, we run the model on the undisrupted network (i.e., steps (0) and (0b) of Table 3.3). We obtain a total passenger dissatisfaction of $z_p^* = 223,400$ minutes and an operational cost of $z_o^* = 87,750$ CHF. In this undisrupted timetable, 20 trains are operated and none of the passenger groups needs to take an opt-out arc, as expected.

Tables 3.5 and 3.6 show the detailed numerical results for the two disruption scenarios. The two first columns indicate the upper bounds that were imposed on passenger inconvenience and deviation from the undisrupted timetable. The third column gives the optimal value of the minimization of the operational cost (i.e., step (3) of Table 3.3), and the associated optimality gap is reported in the fourth column. Column five indicates the computational time. Columns six to ten report performance measures of the timetable from the operational point of view: number of rerouted trains, number of totally or partially canceled trains, number of emergency trains and total delay minutes. Passenger-related performance indicators are reported in columns eleven to fourteen: the average and maximal additional generalized travel time (for non-disrupted passenger groups), and the number of rerouted and disrupted passengers (disrupted passengers use the opt-out arc). Also, each block of rows separates instances with different upper bounds on the passenger inconvenience.

For the disruption scenario where tracks between stations Gouda and Utrecht are unavailable, a passenger dissatisfaction of $z_p^{**} = 241,600$ minutes is obtained, when minimizing the latter without any constraints on operational cost and deviation from the undisrupted timetable (step (1) in Table 3.3). In a second step, the deviation from the undisrupted timetable is minimized, and the optimal value of this problem depends on the bound on the passenger inconvenience (step (2) in Table 3.3). Table 3.5 compares a number of solutions of the model, for different values of ε_p and ε_d. Note that we only include non-dominated and feasible instances in Table 3.5.

Upper bounds		z_o	Gap [%]	Time [s]	Trains					Passenger groups			
ε_p	ε_d				rerouted	totally	partially	emergency	delay	additional travel time		rerouted	disrupted
						canceled	canceled		minutes	average	maximal		
241600	14055	96660	0	73	4	2	0	4	30	2.49	31	7	1
241600	15616	94170	0	111	4	2	0	4	55	2.49	31	7	1
241600	17178	91980	0	136	5	2	1	4	50	2.49	31	7	1
268444	10582	80250	0	726	5	2	0	0	80	3.00	40	5	5
268444	11078	78060	0	2741	10	2	0	0	400	4.24	57	7	4
268444	11574	76470	3.40	3600	7	2	0	1	135	1.51	30	4	6
268444	14055	70500	2.53	3600	9	2	2	2	255	3.80	40	7	4
268444	15616	67860	2.37	3600	10	2	2	3	300	3.80	40	9	4
268444	17178	67350	5.55	3600	9	2	1	2	205	3.87	40	11	4
295288	10582	71610	0	942	8	2	2	0	125	5.69	55	5	7
295288	11078	68790	1.30	3600	9	2	2	0	315	4.65	50	6	8
295288	11574	67020	3.02	3600	11	2	1	0	260	5.85	40	10	7
295288	14055	60660	3.51	3600	12	2	2	0	445	7.25	40	10	6
295288	15616	57420	1.99	3600	11	2	2	0	470	7.27	40	11	6
295288	17178	56340	3.15	3600	13	2	4	0	535	7.07	65	11	6
322133	10582	69450	1.59	3600	8	2	3	0	190	5.13	50	5	10
322133	11078	66240	1.51	3600	9	2	2	0	240	3.69	45	6	11
322133	11574	63210	0	1233	11	2	2	0	360	5.27	40	9	11
322133	14055	57060	2.53	3600	12	2	3	0	440	7.35	40	12	9
322133	15616	54180	1.90	3600	12	2	3	0	375	6.18	65	9	10
322133	17178	52380	3.05	3600	13	2	4	0	390	8.82	65	12	8
456335	9590	77280	0	15	7	2	0	0	195	3.84	50	1	29
456335	10086	70710	0	30	11	2	1	0	205	9.39	61	3	26
456335	10582	67770	0	40	11	2	3	0	280	3.76	77.5	5	28
456335	11078	63570	0	18	13	2	2	0	300	6.82	65	3	27
456335	14055	53160	0	20	14	2	4	0	315	8.03	61	7	26
483200	10582	67770	0	52	11	2	3	0	280	7.51	87	4	30
483200	11078	63570	0	24	12	2	2	0	310	8.75	91	3	30
483200	14055	53160	0	21	14	2	4	0	315	9.65	65	6	29

Table 3.5: Computational results for disruption Gouda-Utrecht (Gou-Utr).

Upper bounds		z_o	Gap [%]	Time [s]	Trains					Passenger groups			
ε_p	ε_d				rerouted	totally	partially	emergency	delay	additional travel time		rerouted	disrupted
						canceled	canceled		minutes	average	maximal		
356700	31575	71310	0	1832	9	4	2	4	245	5.82	89	12	12
356700	35521	65520	0	1986	8	5	2	4	215	5.82	89	13	12
356700	39468	65220	0	2971	7	6	1	4	175	5.82	89	13	12
356700	43415	65220	0	1269	9	6	1	4	220	5.82	89	13	12
356700	47362	65220	0	1123	11	6	1	4	250	5.82	89	13	12
390400	19025	61140	0	1752	10	3	2	1	390	4.98	86	9	17
390400	22162	53400	2.07	3600	9	3	4	2	305	4.93	61	11	17
390400	31575	45300	1.72	3600	8	5	3	2	245	5.73	89	12	17
390400	35521	43440	1.61	3600	8	6	5	2	360	6.71	89	11	16
390400	39468	41670	1.94	3600	8	7	5	3	345	6.62	89	11	16
390400	43415	40080	0.68	3600	6	8	3	3	335	6.67	89	12	16
390400	47362	39450	3.79	3600	8	9	3	3	435	7.93	89	12	15
424100	17425	55080	0	6	8	3	2	0	125	4.88	86	8	21
424100	20962	46170	0	988	13	3	5	0	440	9.67	86	14	18
424100	31575	37800	1.25	3600	12	5	5	1	505	7.24	61	13	20
424100	35521	36210	2.58	3600	9	6	4	1	295	3.89	35	8	22
424100	39468	34080	1.12	3600	11	7	5	2	535	7.33	61	13	20
424100	43415	32850	2.45	3600	10	8	4	2	485	7.05	61	11	20
424100	47362	31680	3.28	3600	7	9	5	1	235	7.80	89	10	20
457800	17425	54390	0	4	8	3	2	0	125	0.95	30	3	29
457800	20962	41760	0	26	13	3	5	0	535	10.42	86	13	21
457800	31575	32670	1.66	3600	13	5	5	0	540	6.49	65	11	24
457800	35521	30750	1.83	3600	13	5	7	0	465	7.42	60	12	24
457800	39468	29160	2.67	3600	11	6	6	1	465	4.55	31	9	27
457800	43415	27840	3.38	3600	11	8	5	0	565	10.36	60	13	22
457800	47362	26160	0	6654	8	9	4	1	455	6.69	31	7	25

Table 3.6: Computational results for disruption Den Haag-Leiden-Schiphol (DeH-Lei-Sch).

It can be observed that the number of disrupted passenger groups and the maximal additional travel time strongly depend on the value of the upper bound on the passenger inconvenience, as expected. For an upper bound with $\varepsilon = 1$, i.e. $\varepsilon_p = z_p^{**} = 241,600$, there is only one disrupted passenger group and the passenger group whose additional travel time increases most has to travel for 31 minutes more than in the undisrupted timetable. This value of the bound on z_p represents the best passenger experience that can be achieved under the circumstances. The price to pay for this level of passenger satisfaction is the high operational cost, 96,660 CHF (1.1 times the operational cost of the undisrupted case). However, when $\varepsilon = 2$ (i.e., $\varepsilon_p = 2 \cdot z_p^{**}$), more than half of the passenger groups are disrupted and the additional travel time can be over one and a half hour longer (91 minutes) for the passenger groups who are worst off. This solution is obviously not satisfying from the passenger perspective, but it has a very low operational cost: between 53,160 CHF and 67,700 CHF, depending on the bound on the deviation from the undisrupted timetable. Note that this operational cost is lower than the operational cost of the undisrupted timetable; this is explained by the fact that only trains operated in a timetable account for its operational cost. Hence, if a train is canceled, the operational cost of the timetable decreases, but it is balanced by an increase of the deviation from the undisrupted timetable. This reasoning also explains why the number of partially canceled trains increases (from 0 to 4) as the operational cost decreases. As canceling a train is costly (in terms of deviation from the undisrupted timetable), the model will always prefer to cancel a train only partially. Hence, the two trains that are totally canceled are trains that spent most of their time on the disrupted tracks and therefore need to be canceled totally. Finally, one can also observe that the number of emergency trains is maximal (4) for the lowest upper bound ε_p (thus offering a high level of service to the passengers). This number decreases very quickly when the bound on the passenger inconvenience becomes less tight, as it is very costly (both in terms of operational cost and in terms of deviation from the undisrupted timetable) to schedule an emergency train.

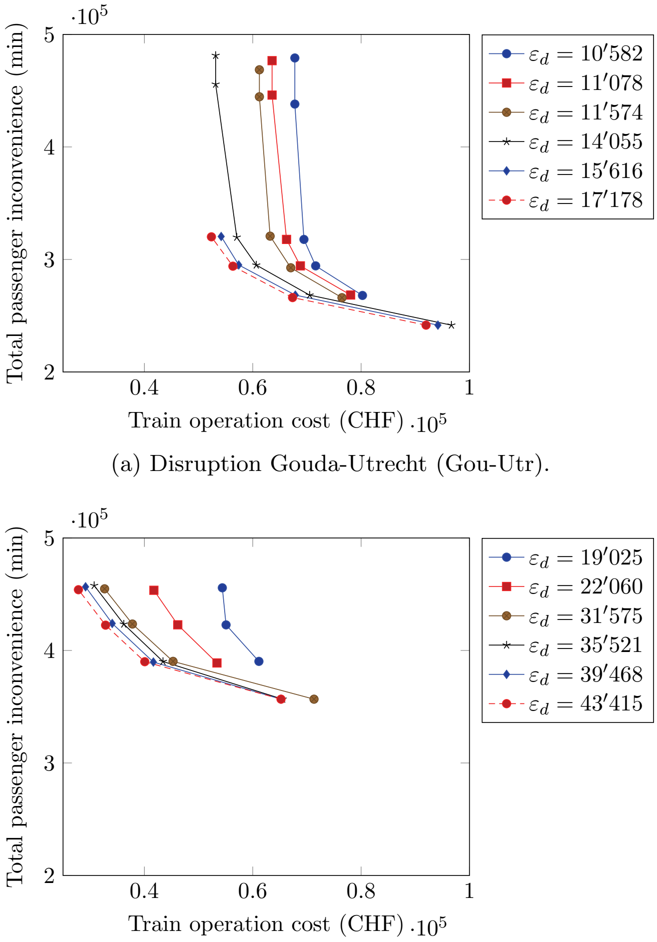
Regarding the disruption scenario where the busiest tracks become unavailable between Den Haag, Leiden and Schiphol (Table 3.6), a passenger dissatisfaction of $z_p^{**} = 356,700$ minutes is obtained without ε-constraints. The observations that were made for the less severe disruption can be reiterated in this case. The increased severity of the disruption shows in several ways:

- First, the unconstrained value of the passenger dissatisfaction (z_p^{**}) is almost 60% higher than in the undisrupted timetable ($z_p^{**} = 356,700$, whereas $z_p^{*} = 223,400$). In other words, in the best case, the passengers will be travelling 1.6 times longer (on average) than in the undisrupted case. By contrast, the unconstrained value of the passenger dissatisfaction increases only by 8% ($z_p^{**} = 241,600$, $z_p^{*} = 223,400$) in the less severe disruption.
- The upper bounds on the deviation from the undisrupted timetable need to be

much higher than in the less severe disruption. Otherwise, the problem becomes infeasible as a too tight bound does not allow enough flexibility in the rescheduling. For instance, the lowest possible bound ε_d is 17,425 for disruption DeH-Lei-Sch and 9,590 for disruption Gou-Utr.

- Even in the best possible configuration in terms of passenger dissatisfaction ($\varepsilon_p = z_p^{**}$), there is a very high number of disrupted passenger groups, and the maximal additional travel time is about one hour and a half.
- The emergency trains are used in many instances, even though they are considered "costly".

Pareto frontier



(a) Disruption Gouda-Utrecht (Gou-Utr).

(b) Disruption Den Haag-Leiden-Schiphol (DeH-Lei-Sch).

Figure 3.1: Pareto frontiers for several values of deviation from the undisrupted timetable.

In order to quantify the trade-off between operational cost and passenger satisfaction, we plot the Pareto frontier for different values of deviation from the undisrupted timetable in Figures 3.1a and 3.1b. The almost vertical line in Figure 3.1a indicates that a significant increase in passenger satisfaction can be achieved with only a little increase in operational cost. We can observe that the higher the deviation from the undisrupted timetable is (i.e., higher ε_d), the better the timetable will perform in terms of passenger satisfaction and operational cost. The issue with high values of ε_d is that the timetable might be very different from the undisrupted one. Hence, the train operating company might want to accept a higher operational cost to achieve the same passenger dissatisfaction, but with a lower deviation from the undisrupted timetable.

Figure 3.1b shows the Pareto frontier for the more severe disruption scenario. Again, the basic observations are consistent between the two plots. The fact that the lines are “less vertical” in the second figure underlines the severity of the disruption: even by increasing the operational cost, only so much passenger inconvenience decrease can be achieved.

3.6 Concluding remarks

In this chapter, an Integer Linear Program for the multi-objective railway timetable rescheduling problem was introduced. Passenger inconvenience, operational cost and deviation from the undisrupted timetable are considered as three different objectives to minimize. The infrastructure is modeled from a macroscopic point of view, by considering stations and track sections in between. When a track becomes unavailable, the model choses between delaying, canceling or rerouting the trains in the undisrupted timetable. The model also includes the possibility to schedule additional emergency trains from depots located near given stations. In addition, passenger flows are adapted dynamically to the new timetable.

The multi-objective nature of the problem is addressed using ε-constraints: one objective is minimized while constraints impose an upper bound on the two other ones. This method has the advantage of using meaningful bounds (e.g., the passenger inconvenience should not increase by more than 50%) in order to construct the Pareto frontier of the problem. This allows to quantify the trade-off between the three conflicting objectives when designing a disposition timetable.

Computational experiments were performed on a case study based on a heavily used part of the Dutch railway network. Results show that significant improvements can be achieved in terms of passenger satisfaction with only a minor increase in the operational cost of the timetable. Also, the higher the deviation from the undisrupted timetable is allowed, the better the timetable will perform in terms of passenger satisfaction and operational cost.

Using a commercial ILP solver, the model is solved to optimality on most instances, showing that it is possible to account for passenger satisfaction in disposition timetables. Furthermore, it is possible to keep the associated operational costs low and to control for the deviation from the undisrupted timetable. However, the computational time makes the current implementation impractical to use for real-time timetable rescheduling (some instances are solved in about 3 minutes, while others have an optimality gap of 3% after one hour). It can nonetheless be used by train operating companies that wish to generate offline recovery scenarios for highly disrupted scenarios and quantify their effect regarding the three aforementioned objectives.

Appendix

3.A Construction algorithm of the rescheduling graph

Algorithm 1: Construction of the rescheduling graph.

```
Input  : Complete graph, $G(V, A)$
         Set of available tracks during disruption, $Q^*$
         Timetable of every original train $k \in K_1$, $T_k \subset A$
         Ordered set of stations visited by original train $k \in K_1$, $S_k \subset S$
         Maximal delay $M_D$
Output: Rescheduling graph, $G(V, A^*)$
         Available arcs for original train $k \in K_1$, $A_k$

/* Initialization */
$A^* \leftarrow A$
/* Remove disrupted arcs */
for $a = ((s, p, t), (s', p', t')) \in A_{Dri}$ do
    if $(s, s') \notin Q^*$ then
        $A^* \leftarrow A^* \setminus \{a\}$
/* Construct available arcs for train $k \in K_1$ */
for $k \in K_1$ do
    $A_k \leftarrow T_k \cap A^*, A_k^D \leftarrow \varnothing, A_k^{RR} \leftarrow \varnothing$
    /* Construct set of delay arcs $A_k^D$ for original train $k \in K_1$*/
    for $a_1 \in T_k \cap A^*$ do
        for $a_2 \in A^*$ do
            if $a_1 = (r_1, (s_1, p_1, t_1)), a_2 = (r_2, (s_2, p_2, t_2))$ are starting arcs and
              $r_1 = r_2, s_1 = s_2, p_1 = p_2, t_2 - t_1 \leq M_D$ or $a_1 = ((s_1, p_1, t_1), r_1), a_2 = ((s_2, p_2, t_2), r_2)$ are ending
              arcs and $r_1 = r_2, s_1 = s_2, p_1 = p_2, t_2 - t_1 \leq M_D$ or
              $a_1 = ((s_1, p_1, t_1), (s'_1, p'_1, t'_1)), a_2 = ((s_2, p_2, t_2), (s'_2, p'_2, t'_2))$ are driving arcs and
              $s_1 = s_2, p_1 = p_2, s'_1 = s'_2, p_1 = p'_2, t'_2 - t'_1 \leq M_D$ or
              $a_1 = ((s_1, p_1, t_1), (s'_1, p'_1, t'_1)), a_2 = ((s_2, p_2, t_2), (s'_2, p'_2, t'_2))$ are waiting arcs and
              $s_1 = s_2, p_1 = p_2, t'_2 - t'_1 \leq M_D$ then
                $A_k^D \leftarrow A_k^D \cup a_2$
    /* Construct set of rerouting arcs $A_k^{RR}$ for original train $k \in K_1$ */
    for $s, s' \in S_k$ do
        Construct $P_{s,s'}$, the set of all paths in $G(V, A^*)$ from $(s, p, t) \in N$ to $(s', p', t') \in N$, with
          $p \in P_s, p' \in P_{s'}, t, t' \in H$ and $t' > t$.
        $l(p) \leftarrow$ duration of path $p \in P_{s,s'}$, i.e. time difference between last and first node in $p$.
        $l \leftarrow \min_p l(p)$
        for $p \in P_{s,s'}$ do
            if $l(p) \leq l + M_D$ then
                Add all arcs of path $p$ to $A_k^{RR}$.
    $A_k = A_k^D \cup A_k^{RR}$
```

4

An exogenous capacitated passenger assignment methodology

This chapter is based on the article:

`Binder, S., Maknoon, Y., and Bierlaire, M. (2017). Exogenous priority rules for the capacitated passenger assignment problem, Transportation Research Part B: Methodological 105:19-42.`

The work has been performed by the candidate under the supervision of Dr. Yousef Maknoon and Prof. Michel Bierlaire.

4.1 Introduction

In the previous chapter, the underlying demand-related assumptions result in a system-optimal assignment of the passengers on the timetable. Indeed, minimizing the overall passenger inconvenience, while enforcing that all trips are covered, leads to a situation where everyone is best off, *on average*. However, a system-optimal passenger assignment underestimates the passenger inconvenience, due to the selfish behavior of public transportation users, who are usually not willing to accept a longer personal travel time for a theoretical "greater good". The aim of this chapter is therefore to introduce a behaviorally more realistic algorithm for the passenger assignment problem with tight capacities.

One of the main issues of passenger assignment models is to account properly for passenger behavior in case of vehicle saturation. When passengers compete for the limited available capacity of trains, it is of critical importance to decide which passengers can board the train and which cannot. Indeed, consider a situation where a passenger has the choice to board a train or to wait for the next one on the same line. If she boards the train, she will use some of its available capacity and, at a later stop, her presence in the train might prevent another passenger to board. On the contrary, if she decides not to board the train, the available capacity might be sufficient to let the subsequent passenger board.

This simple example highlights why priorities between passengers lie at the core of a behaviorally meaningful passenger assignment model. Some early studies (including Nguyen and Pallottino, 1988; Spiess and Florian, 1989; de Cea and Fernández, 1993; Nuzzolo et al., 2001; Gao et al., 2004) disregard the tight capacity constraints and hence do not need to define priorities. Subsequent contributions address the tight capacity constraints either implicitly (by increasing link travel times to penalize the generalized cost when flow exceeds capacity — see, e.g., Tong and Wong (1999); Nguyen et al. (2001)) or explicitly (through an inequality forcing the the assigned flow to be lower or equal than the capacity — see, e.g., Poon et al. (2004); Hamdouch et al. (2004); Hamdouch and Lawphongpanich (2008); Nuzzolo et al. (2012)). As can be seen in the literature review presented in Section 4.2, most frameworks addressing the tight capacity constraints explicitly use a first-in-first-out queue at the stations, coupled with the rule that on-board passengers have priority over those wishing to board. This priority rule is endogenous because it depends on the passenger assignment. The endogeneity forces the existing frameworks to embed the passenger priorities implicitly in the network loading process, making them difficult to modify, in the case where one desires to study the impacts of alternative orderings.

By contrast, in this chapter, we propose a novel framework that considers an exogenous ordering of the passengers. By defining beforehand the rules that govern passenger priorities, complexity is extracted from the assignment problem itself. The main contribution of this chapter is to completely separate the problem of defining priorities among passengers from the actual assignment problem. By doing so, it becomes much simpler for the analyst to alter one problem without affecting the other. For instance, we use the shortest-path assignment in the illustrative case study presented in Section 4.6, but a more complex route choice model may be implemented easily. The fact that the priority lists are completely independent from the assignment process also guarantees their flexibility. Any behavioral or control rule can be used to construct the priority list, including rules that account for the results of a previous assignment, in an iterative context. In this case, an additional fixed-point problem needs to be solved.

The actual assignment algorithm takes three elements as input: (i) a time-dependent origin-destination matrix representing passenger demand, (ii) a space-time graph constructed from the railway timetable, and (iii) the passenger priority list. Passenger flows in the network are obtained by assigning the demand in the order given by the priority list (i.e., a passenger with higher priority will be assigned before a passenger with lower priority), thus settling the issues that arise in case of insufficient train capacity.

The remainder of this chapter is structured as follows. Section 4.2 reviews recent contributions to the field of passenger assignment models. The problem is then formally introduced in Section 4.3. The generation process for the exogenous passenger priority lists is explained in Section 4.4, whereas the assignment algorithm itself is described in Section 4.5. This section also details how our formulation can be embedded in a fixed-point specification in order to address endogenous passenger demand and priority specifications. Section 4.6 reports the results of the computational experiments on the case study. Finally, Section 4.7 provides concluding remarks.

4.2 Literature review

The recent literature on passenger assignment models for transit systems is either *frequency-based* or *schedule-based*. In the former approach, transit services are represented by lines with travel frequencies and single vehicles are not explicitly considered. Frequency-based static assignment models are generally suited for urban transportation systems (metro, bus) where the service is so frequent that it can be assumed that a passenger boards the first “attractive” vehicle when waiting at a stop. Seminal works in this field include Gendreau (1984), who first considered the combination of the effects of congestion with the common-lines problem arising in public transportation networks; Spiess and Florian (1989), who introduced the concept of optimal strategies; and Nguyen and Pallottino (1988), who formalized the concept in terms of shortest hyperpaths. Many extensions have been proposed in the following years (e.g., de Cea and Fernández (1993); Cominetti and Correa (2001)). The interested reader can refer to Fu et al. (2012) for an in-depth review of frequency-based passenger assignment models.

Single vehicle loads can only be approximated in frequency-based models. This approximation is especially unsuitable in case of irregular service (which is common in inter-urban systems such as trains or long-distance coaches), as it cannot account for peaks of passengers waiting at the station. In order to model the choice of passengers for a specific run of a specific transportation line, a schedule-based approach is needed. The loads and the performance of each single run can be obtained in such a framework. In schedule-based models, each vehicle is considered individually with its capacity, either implicitly or explicitly. The implicit approach is similar to road network modeling, where link costs are related to link flows through non-decreasing functions. Papers such as Tong and Wong (1999); Nguyen et al. (2001); Nuzzolo et al. (2001) and Nielsen (2004) use this approach. By contrast, the explicit schedule-based approach introduces vehicle capacity constraints, thus letting waiting passengers board the arriving train according to its residual capacity. The following papers use the explicit schedule-based approach to assign passengers on transit networks.

Poon et al. (2004) propose a model that explicitly describes the available capacity of every vehicle at each station, as well as the queuing time for every passenger. The paper focuses on the route choice problem, ignoring other choice dimensions, such as departure time or departure station. In their formulation, route choice for every passenger is modeled by selecting a path that minimizes a generalized cost function consisting of in-vehicle time, waiting time, walking time and line change penalties. The network is loaded (i.e., user equilibrium is achieved) by using a Method of Successive Averages (MSA) algorithm. The authors assume a First-In-First-Out (FIFO) queue discipline at the stations. Depending on the spare capacity of a transit vehicle at a station, the queue is split into passengers that can board the vehicle and passengers that remain at the station and wait for the next vehicle to arrive.

Hamdouch and Lawphongpanich (2008) also propose a user-equilibrium transit assignment model that explicitly considers individual vehicle capacities. For every origin-destination pair, passengers are divided into groups according to their desired arrival time intervals. It is assumed that every passenger group has a travel strategy resulting, at each station and each point in time, in a list of subsequent travel options that are ordered according to the passenger groups' preferences to continue their trip. Passenger preferences are described by the minimization of expected travel costs, made of in-vehicle time, fare and costs associated with early departures from home and/or arrivals outside the desired arrival time interval. When loading a vehicle at a station, on-board passengers continuing to the next station remain in the vehicle and waiting passengers are loaded according to the available remaining vehicle capacity. The paper considers two rules to sequence the boarding procedure of the waiting passengers: first-come-first-serve order and random order. If the vehicle is full, passengers unable to board need to wait for the next vehicle. Demand-supply interactions are defined by a user equilibrium approach and a solution method based on a MSA is proposed.

In Papola et al. (2009), a Dynamic Traffic Assignment model is extended to the case of scheduled services. It allows for explicit vehicle capacity constraints and FIFO queue representation at stations. The authors formulate the deterministic user equilibrium as a fixed-point problem in terms of flow temporal profiles. A MSA algorithm is proposed to solve the problem.

Sumalee et al. (2009) propose a dynamic transit assignment model that differentiates discomfort levels experienced by sitting and standing passengers. The probability of getting a seat is captured by a stochastic seat allocation model. The passengers choose departure time and travel route by minimizing the perceived expected disutility, made of walking time, waiting time, early/late arrival penalty, expected perceived in-vehicle time (including congestion effect), transit fare and number of transfers. It is assumed that passengers boarding a vehicle obey the FIFO discipline. Further, standing passengers already on-board have priority over boarding passengers to gain access to available seats. The authors formulate the departure time and route choice problem as a probit stochastic user equilibrium problem and develop a heuristic solution algorithm to find an equilibrium solution.

Nuzzolo et al. (2012) propose a schedule-based dynamic assignment problem for congested transit networks, explicitly considering vehicle capacities. Its novelty resides in the fact that more complex behavioral choice models are used for passengers, including the choice of departure time, departure station and departure train run. A day-to-day learning process for the passengers is also included in the model. The network loading procedure assigns users on each transit run according to their choice and to the residual capacity of the vehicles arriving at the stop. Again, FIFO queueing discipline is assumed.

Hamdouch et al. (2014) extend the framework of Hamdouch and Lawphongpanich (2008) by considering supply uncertainties. An analytical formulation is provided to ensure that on-board passengers have priority over boarding passengers and waiting passengers are loaded according to FIFO precedence. A user equilibrium model is proposed for the problem, which is solved by a MSA-type algorithm.

Kroon et al. (2015) present a deterministic algorithm to simulate passenger flows in a capacitated network. Passengers are grouped according to origin-destination pairs and arrival time in the system. Passenger flows emerge from the competition between the passengers for the limited capacity of the trains. When more passengers attempt to board a train than the capacity allows, passenger groups are split: some passengers can board the train while the remaining will have to look for an alternative travel route. The authors also assume that passengers who are already on a train when a train calls at a station, and who intend to continue on the same train, have priority over newly boarding passengers.

Liu and Zhou (2016) propose a new framework for the transit service network design problem with tight capacity constraints. The underlying transit assignment problem assumes boundedly rational travel behavior of the agents. The authors do not consider FIFO queuing discipline for waiting or transferring passengers. Instead, the boundedly rational travel decision rule is used to decide who can board the transit vehicle in oversaturated conditions.

4.2.1 Summary and contributions of this chapter

In all the reviewed works (except Kroon et al. (2015) — see below), priority is given to passengers already on-board with respect to passengers wishing to board a vehicle, and the priority among boarding passengers is decided using a FIFO or random rule. These priorities are endogenous because the passenger ordering depends on the assignment, and vice versa. The issue with endogeneity is the fact that the logic behind the priorities needs to be "hard-coded" into the assignment algorithm itself. By contrast, we propose exogenous priority rules, which remove the complexity from the network loading procedure and allow the modeler to investigate potential alternative rules very easily. Note that our framework can easily be included in a fixed-point formulation to calculate the user equilibrium, in the case where passenger decisions depend on the assignment (if crowding is considered, for instance). It can therefore be seen as an extension of recent contributions to the field of schedule-based passenger assignment models.

A notable exception is the work by Kroon et al. (2015), where passengers are moved through the network according to the available capacity of each vehicle. In case of vehicle saturation, passenger groups are split and the number of passengers that can board the train is proportional to the group size. Our framework is more flexible than this one as it allows to define any priority rule to order the passengers before the assignment.

This chapter provides a flexible and efficient methodology for the analyst to model the capacitated passenger assignment problem. The framework can also be used by railway operators in order to assess different boarding strategies at stations quantitatively. For instance, the operator might want to prioritize short-distance passengers for a given train line, because this strategy improves one of the performance indicators significantly. It would then still be the operator's responsibility to decide how to enforce the compliance of the passengers with the imposed priorities.

The contributions of this chapter are summarized as follows:

- We propose a flexible network loading framework for the capacitated passenger assignment problem.
- We define exogenous priority lists to order the passengers before the assignment, thus extracting the complexity from the assignment problem.
- We allow total flexibility for the definition of the priority rules.
- We carry out computational experiments on a realistic case study and our framework is able to assign the demand in little time.

4.3 Problem description

This section is dedicated to the formal description of the problem addressed in this chapter. We begin by explaining the representation of the demand and by describing the assumptions on passenger behavior that lead to our passenger travel choice model. The representation of the public transportation timetable as a space-time graph is portrayed thereafter. We conclude this section by discussing the interactions between the demand side (passengers) and the supply side (train capacities) of the problem, which lie at the core of a passenger assignment model.

In the remainder of this chapter, we reuse the following notation, introduced in Chapter 3. Time is discretized into $n+1$ time intervals of length τ (typically, one minute) and we introduce the set of time steps $H = \{0, \tau, 2\tau, \cdots, n\tau\}$, where $n\tau$ is the considered planning horizon. We model the railway network at a macroscopic level. The set of stations is denoted by S, and the travel time between two consecutive stations $s, s' \in S$ by $t(s, s') \in H$. This travel time is deterministic and independent of the train. The set of all trains in the timetable is denoted by K. Further notations are introduced when needed.

4.3.1 Passenger travel choice model

The assumptions on passenger behavior are very similar to Chapter 3 at the individual passenger level: the basic idea is to associate a generalized travel cost with every path from origin to destination and to assume that the passenger wishes to travel along the path of lowest generalized cost. The only difference is that passengers are not assumed to form groups as in Chapter 3. It is at the collective level that the assumptions differ between the two chapters, as described in Section 4.3.3.

Passenger demand is assumed to be known, in the form of an origin-destination (OD) matrix. The latter describes the number of passengers entering the system at a given origin station, at a certain time, and who wish to travel to a given destination station. The availability of such data becomes more and more frequent with the gradual introduction of smart cards in public transportation networks.

Based on the OD matrix, a passenger p is denoted by a triplet (o_p, d_p, t_p), where $o_p \in S$ is the origin station, $d_p \in S$ the destination station, and $t_p \in H$ the desired departure time from the origin. Note that, as we assume deterministic train travel times in our approach, a passenger can equivalently be characterized by the desired arrival time at the destination. We adopt the former representation in the following. The set of all passengers is denoted by P.

We assume that the passengers know the timetable (i.e., all the train departure and arrival times at all stations) and that they plan their path in the network accordingly. In particular, we assume that the passengers know the modified timetable in case of a disruption, for instance by using a smartphone application. A passenger path may consist of different trains if there is no direct connection between origin and destination of the passenger. For every passenger, we consider the set $\Omega(o_p, d_p)$ of all paths linking o_p to d_p. A path is defined as a sequence of access, in-vehicle, waiting, transfer and egress movements (refer to Section 4.3.2 for a definition in terms of arcs in the space-time graph). In order to distinguish different paths, we associate a generalized cost with every alternative (i.e., path) and assume that each passenger chooses the one with the lowest generalized cost. In general, the specification of generalized cost C^p_ω for alternative $\omega \in \Omega(o_p, d_p)$ and passenger p is a function of several attributes:

$$C^p_\omega = C(\mathbf{z}^p_\omega, \mathbf{y}_p; \beta),$$

where $\mathbf{z}^p_\omega$ is a vector of attribute values (such as travel time and cost) for alternative ω as viewed by passenger p, and β is a vector of parameters representing the weight of the different attributes. As preferences may vary across the passengers, this general specification also includes a vector $\mathbf{y}_p$ of socioeconomic characteristics (e.g., income, age, education) that models the heterogeneity of the population. Our formulation is thus general enough to accommodate heterogenous demand, but in this research we focus on homogenous demand, similarly to Chapter 3. We also assume that the price of the trip is equal among all paths for a given passenger. We therefore consider, for a given passenger p and path ω, the following attributes in the specification of the generalized cost:

- *In-Vehicle Time* (VT^p_ω): time, in minutes, spent by the passenger in one (or more) train(s) along the path,
- *Waiting Time* (WT^p_ω): time, in minutes, spent by the passenger waiting between two consecutive trains at a station along the path (does not consider the waiting time for the first train),
- *Number of Transfers* (NT^p_ω): number of times the passenger needs to change trains along the path,
- *Early Departure* ($ED^p_\omega = \max(0, t_p - t)$): time difference (in minutes) between the desired (t_p) and the actual (t) departure time from origin, if early,
- *Late Departure* ($LD^p_\omega = \max(0, t - t_p)$): time difference (in minutes) between the actual (t) and the desired (t_p) departure time from origin, if late.

Based on the aforementioned description for a given passenger p, the generalized cost of alternative ω is defined as follows:

$$C^p_\omega = VT^p_\omega + \beta_1 \cdot WT^p_\omega + \beta_2 \cdot NT^p_\omega + \beta_3 \cdot ED^p_\omega + \beta_4 \cdot LD^p_\omega, \tag{4.1}$$

where $\beta_1, \ldots, \beta_4$ are the relative weights of the attributes described above. C_ω^p is in minutes and expresses the generalized travel time of passenger p along path $\omega \in \Omega(o_p, d_p)$. As commonly done in the literature, the weights of the various elements of the generalized travel time are defined relative to the in-vehicle time of the path. We use the values reported in Table 2.6, obtained from the literature.

We assume that passengers have full knowledge of the system and that they choose the path with lowest generalized travel time to travel from origin to destination. Due to train capacity issues, it is possible that, for some passengers, no feasible alternative exists between origin and destination within the time horizon. We therefore include an artificial "opt-out path" for those passengers. This path models the worst possible option to travel from origin to destination. We therefore associate it with the highest possible travel time: the duration of the time horizon.

In case of high demand, capacity issues may also arise because of passenger congestion. In other words, a passenger may fail to board a train on his preferred path, characterized by Eq. (4.1), because this train has reached its capacity due to other passengers having boarded the train earlier. The consideration of such interactions between passengers is discussed in detail in Section 4.3.3 and lies at the core of the passenger assignment algorithm designed in Section 4.5.

4.3.2 Timetable representation as a space-time graph

The passenger paths are mathematically defined in a graph representation of the timetable. To that end, we introduce a directed space-time graph $G(V, A)$. The latter differs from the graph introduced in Chapter 3 in the following aspects:

- As the timetable is an input of the passenger assignment problem (i.e., the timetable is set), it is not necessary to represent all possible train movements in the graph, but only those that actually happen in the timetable. There is therefore no need for a distinction between train and passenger nodes or arcs, nor for a different weighting scheme of the arcs. Modeling railway shunting yards also becomes irrelevant. Hence, the sets of shunting yard nodes (N_R) and of starting and ending arcs (A_{Sta} and A_{End}) are not defined in this chapter. Additionally, it is not necessary to construct the rescheduling graph $G(V, A^*)$ from $G(V, A)$.
- For tractability reasons of the ILP model, the space-time nodes defined in Chapter 3 did not include a train index k, but only a platform index p. In this chapter, we disregard platforms in stations, and consider individual trains instead.
- Arc capacities, corresponding to the train capacities of the arcs they refer to, are introduced in this chapter. In Chapter 3, they were addressed directly in the ILP formulation.

The set of nodes $V = N \cup N_O \cup N_D$ consists of three different types of nodes. Starting from an empty graph, we add a space-time node $(s, t, k) \in N$ for each arrival/departure event of train $k \in K$ at/from station $s \in S$ at time $t \in H$. For instance, if train $k' \in K$ leaves station $s_1 \in S$ at time $t_1 \in H$, stops at station $s_2 \in S$ from time $t_2 \in H$ to $t_3 \in H$ and finishes its trip at station $s_3 \in S$ at time $t_4 \in H$, four space-time nodes are added: (s_1, t_1, k'), (s_2, t_2, k'), (s_2, t_3, k') and (s_3, t_4, k'). In addition, N_O and N_D are the sets of time-invariant origin and destination nodes of the passengers. We denote by $s(o)$ and $s(d)$ the station associated with node $o \in N_O$ and $d \in N_D$, respectively.

There are six types of arcs in the graph:

- *Driving arcs* model the movements of trains between stations. From the timetable, we define, for every train $k \in K$, the set of driving arcs A^k_{Dri}. A driving arc connects a departure event at one station $(s, t, k) \in N$ to an arrival event at the following station $(s', t', k) \in N$, with $t' = t + t(s, s')$. By repeating this procedure for every train in the timetable, we construct the set of driving arcs $A_{Dri} = \cup_{k \in K} A^k_{Dri}$.

- *Waiting arcs* model trains waiting at a station for passengers to board or alight. We define from the timetable, for every train $k \in K$, the set of waiting arcs A^k_{Wai}. A waiting arc connects an arrival event at a station $(s, t, k) \in N$ to a departure event from the same station $(s, t', k) \in N$, with $t' = t + w(s)$, where $w(s)$ is the waiting time at station s. By repeating this procedure, we construct the set of waiting arcs $A_{Wai} = \cup_{k \in K} A^k_{Wai}$.

- *Access arcs* model passenger p arriving at the origin. They are given by the set $A^p_{Acc} = \{(o, (s, t, k)) \in N_O \times N | s = s(o) = o_p\}$. Note that, by definition, passenger p can therefore take any train that departs from his origin station.

- *Egress arcs* model passenger p leaving the system at destination. They are given by the set $A^p_{Egr} = \{((s, t, k), d) \in N \times N_D | s = s(d) = d_p\}$.

- *Transfer arcs* model passengers transferring from one train to another in a station, with a minimal transfer time m and a maximal transfer time M. The set of transfer arcs is constructed in the following way: $A_{Tra} = \{((s, t, k), (s, t', k')) \in N \times N | \forall s \in S, \forall k \in K, \forall k' \in K \setminus \{k\}, \forall t, t' \in H : m \leq t' - t \leq M\}$.

- *Opt-out arcs* model passengers not succeeding to find a path from origin to destination. They are given by the set $A_{Opt} = \{(o, d) \in N_O \times N_D\}$.

The set of arcs is given by $A = A_{Dri} \cup A_{Wai} \cup (\cup_{p \in P} A^p_{Acc}) \cup (\cup_{p \in P} A^p_{Egr}) \cup A_{Tra} \cup A_{Opt}$. With each arc a, a capacity q_a and a weight c^p_a are associated (see Table 4.1). The capacity of driving and waiting arcs is given by the capacity of the associated train (i.e., the maximum number of passengers that can be in the train at the same time), q_k. Access

Table 4.1: Arc weights and capacities.

Name	Start node	End node	Weight (c_a^p)	Capacity (q_a)
Driving	(s,t,k)	(s',t',k)	$t'-t$	q_k
Waiting	(s,t,k)	(s,t',k)	$\beta_1 \cdot (t'-t)$	q_k
Access	o_p	(s,t,k)	$\beta_3 \cdot \max(0,(t_p-t))+$ $\beta_4 \cdot \max(0,(t-t_p))$	1
Egress	(s,t,k)	d_p	0	1
Transfer	(s,t,k)	(s,t',k')	$\beta_2+(t'-t)$	∞
Opt-out	o_p	d_p	$n\tau$	∞

and egress arcs have a capacity of one, as they are passenger-specific. Transfer and opt-out arcs have infinite capacity. The arcs are weighted according to the generalized cost function introduced in Section 4.3.1.

The cost of a path in the graph for a passenger is obtained by summing the weights of the arcs along the path. Note that the choice of a path determines the exact train(s) a passenger takes in the network, as well as his departure time from origin (penalizing early or late departure with respect to the desired departure time). Therefore, our framework is able to model accurately passenger behavior in congested public transportation networks by reproducing effects such as waiting for the next train when failing to board the current train (because of congestion) or adjusting the departure time in order to anticipate congestion.

4.3.3 Supply-demand interactions in the assignment

The goal of our framework is to obtain accurate passenger flows in a public transportation network where passengers compete for the limited capacity of the trains. When the number of passengers attempting to board a train exceeds its available capacity, it has to be decided eventually which passengers can board the train and which cannot. The remaining ones will have to look for another alternative (e.g., wait for the next train of the line). There are two main paradigms to take this decision:

System optimum Passengers are assumed to collaborate in order to minimize the overall inconvenience (i.e., generalized travel time) of all passengers.

User equilibrium Passengers are assumed to be selfish actors that attempt to minimize their personal inconvenience.

Although a system optimal passenger assignment yields a better experience for everyone *on average*, public transportation users are usually not willing to accept a longer personal travel time for a theoretical “greater good”. In this chapter, by contrast with the previous one, we therefore assume passengers to be selfish and independent and to minimize their personal travel time, given by Eq. (4.1). In order to decide which passengers can board a train in the situation described above, we introduce in the following section exogenous passenger priorities that order the passengers according to different criteria. Appendix 4.A reports the Price of Anarchy of using these orderings instead of a system-optimal assignment.

4.4 Exogenous passenger priority lists

We introduce in this section the concept of passenger priority lists that model the relative importance of the passengers. A priority list is characterized by an injective mapping $\Gamma : P \to \mathbb{R}$ that associates each traveler $p \in P$ with a “level of importance” $\Gamma(p)$. It is then assumed that passenger p has priority on passenger q if and only if $\Gamma(p) > \Gamma(q)$. The injection assumption is designed to avoid ties. The challenge for the modeler is to translate actual priority rules into an appropriate mapping Γ. Note that the framework must be general enough to model priority rules that are actually prevailing in practice, or rules that need to be evaluated by an operator, or artificial rules that are used to benchmark the system.

In order to allow for a great deal of modeling possibilities, we propose a probabilistic approach to build the priority list. The importance of passenger p is modeled using a random variable U_p. Therefore, the *probability* that passenger p has priority over passenger q is given by $Pr(U_p \geq U_q)$. Actual mappings can then be obtained by simulation. For each realization $r = 1, \dots, R$, the mapping $\Gamma^r(p)$ is defined as $\Gamma^r(p) = U_p^r$, where U_p^r is a realization of the random variable U_p. Note that if U_p is a continuous random variable, the probability that two passengers share the same level of importance is zero, so that each mapping is an injection with probability 1. However, due to finite arithmetic, ties may appear occasionally. If so, they may be broken in an arbitrary way. We present below concrete examples of the specification of the random variable U_p. The importance function U_p is composed of two components: a deterministic part (V_p) and a stochastic part (ε_p),

$$U_p = V_p + \varepsilon_p, \forall p \in P. \tag{4.2}$$

Deterministic part V_p In order to provide modeling flexibility, the deterministic part of the importance function is a function of various attributes of the passenger. In this research, we illustrate the concept using five different specifications, summarized in Table 4.2.

Table 4.2: Definitions of the passenger priority specifications.

Specification	Prioritized passengers	V_p
D	Passengers with early desired departure time from origin	$-t_p$
M	Passengers whose generalized travel time increases the most if they miss their first choice	$C^p_{\omega_2^p} - C^p_{\omega_1^p}$
S	Passengers with short origin-destination pairs	$-C^p_{\omega_1^p}$
L	Passengers with long origin-destination pairs	$C^p_{\omega_1^p}$
R	Random	0

- **Specification D** The first specification assumes that passengers with earlier desired departure time from origin are assigned before passengers with later desired departure time. This assumption is consistent with the observation that passengers who arrive earlier at a station board before the others. Also, trains will be filled up from the beginning to the end of their journey. In this case, $V_p = -t_p$, where t_p is the desired departure time of passenger p.

- **Specification M** The second specification assumes that passengers whose generalized travel time increases most if they miss their first travel choice are prioritized. The behavioral motivation behind this assumption is to prioritize passengers who have the most to "regret" if they cannot board their train. More specifically, we construct, for every passenger p, the set of all paths in $G(V, A)$ between o_p and d_p, $\Omega(o_p, d_p)$. Then we denote by ω_1^p (ω_2^p) the shortest (respectively, second-shortest) path of passenger p, i.e., $\omega_1^p = \{\omega \in \Omega(o_p, d_p) | C^p_\omega = \min_{\omega' \in \Omega(o_p,d_p)} C^p_{\omega'}\}$ and $\omega_2^p = \{\omega \in \Omega(o_p, d_p) | C^p_\omega = \min_{\omega' \in \Omega(o_p,d_p) \setminus \{\omega_1^p\}} C^p_{\omega'}\}$. The deterministic part of the importance function is then given by the difference in the generalized cost of the two paths: $V_p = C^p_{\omega_2^p} - C^p_{\omega_1^p}$.

- **Specifications S and L** The third and fourth specifications assume that the train operator wishes to give priority to passengers that have either short or long origin-destination pairs. In this case, the systematic part of the importance function is given by the generalized cost of the shortest path of passenger p, with a different sign. To prioritize short OD pairs, $V_p = -C^p_{\omega_1^p}$, whereas to prioritize long OD pairs, $V_p = C^p_{\omega_1^p}$.

- **Specification R** Finally, a priority specification where the passengers are assigned in a completely random order can be modeled using a constant systematic part of the importance function, so that only the random term matters. For instance, $V_p = 0$.

Specifications D and M are behaviorally driven and therefore quite realistic. Passengers arriving early at the station will probably board before passengers arriving later (specification D). Passengers prioritized in specification M (those who have the most to “lose” if they miss their train) can be assumed to be very aggressive in order to gain priority over other passengers, thus leading to a higher rank in the priority list. It may actually even happen that their second choice is not available either. This specification ignores that fact. Specifications S and L can be considered as control strategies that may be implemented by the train operator. For instance, the operator might want to give priority to “VIP” passengers or passengers on a particular OD for marketing reasons. Compliance of the passengers for specifications S and L can be enforced using specific lanes as in the airline industry or special carriages reserved for the prioritized passengers. Finally, specification R can be seen as a benchmark to compare to the other specifications.

Note that the variables influencing the order of the passengers in the five specifications are independent of the assignment. In reality however, the priority lists are not always exogenous. The flexibility of our framework also allows to accommodate endogenous priority lists: Section 4.5.3 details how it can be embedded in a fixed-point formulation.

Stochastic part ε_p The stochastic part is designed to capture various elements, not explicitly modeled by the deterministic part, that may influence the relative “importance” of passengers and, therefore, their priority to board the train. Examples include the exact time of arrival on the platform, or the distance from the platform to the train door. In practice, a probability distribution must be introduced. It is convenient to assume that the ε_p are independent and identically distributed across p. The mean of this distribution can be assumed to be any arbitrary value (typically zero), as any deviation from the assumption can be captured by the deterministic part of the importance function. The variance, on the other hand, matters a lot, as it controls the relative importance between the deterministic and the stochastic parts of the importance function. For instance, following the assumptions applied in the context of discrete choice models, we assume that ε_p are independent and identically extreme value distributed with location parameter $\eta = 0$ and scale parameter $\mu > 0$, noted $\varepsilon_p \sim EV(0, \mu)$. Choosing the extreme value distribution to model the stochastic part of the importance function allows to obtain the logit closed-form formula (see, e.g., Ben-Akiva and Lerman (1985)) for the probability that passenger p has priority over passenger q:

$$Pr(U_p \geq U_q) = \frac{\exp(\mu V_p)}{\exp(\mu V_p) + \exp(\mu V_q)}.$$

This assumption is motivated by the analogy with choice models, where decision makers maximize their utility. Note that any other assumption about the error term can also be used. Here, the importance of passengers is also maximized, to identify passengers with higher priorities. The cumulative distribution function of a random variable $x \sim EV(0, \mu)$ is given by $F(x) = \exp(-\exp(-\mu x))$. Draws v from this distribution are obtained from draws u from the uniform distribution as follows:

$$v = -\frac{1}{\mu}\ln(-\ln u), \qquad u \sim U(0,1). \tag{4.3}$$

Note that the variance of the random variable is inversely proportional to the square of the scale parameter: $\frac{\pi^2}{6\mu^2}$. Hence, the scale factor μ captures the variability of the disturbances. Appendix 4.B describes how its value is determined for every specification.

4.5 Passenger assignment algorithm

Given a public transportation network, we present a stochastic schedule-based passenger assignment algorithm that considers vehicle capacities explicitly. Fig. 4.1 describes the general framework. The passenger assignment model takes the following information as an input: a train timetable and a passenger OD matrix (both described in Section 4.3), as well as a priority list Γ for the passengers, as discussed in Section 4.4. The latter defines the order in which the passenger demand is assigned on the network: a passenger with a higher priority will be assigned before another passenger with lower priority. Based on these inputs, the assignment model computes the passenger flows on every arc of the network. Performance indicators can then be deduced from the passenger flows.

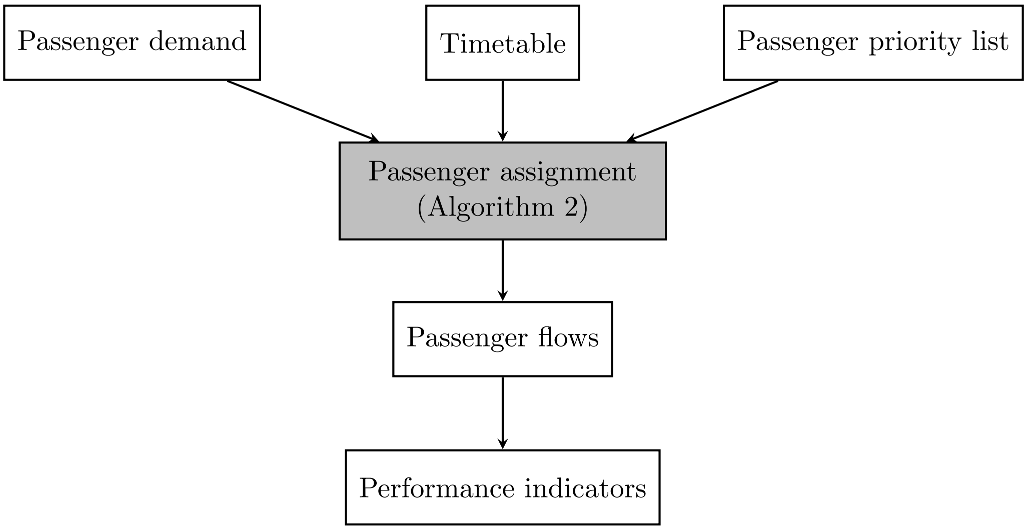


Figure 4.1: Inputs and outputs of the passenger assignment model.

Note that this framework assumes that crowding does not affect the passenger decisions, nor the priority list. In reality however, passengers may adjust their departure time or wait for the next vehicle if the current one is close to saturation. As a consequence, they can gain or lose priority in the system. As discussed in Section 4.5.3, the framework can easily be extended to include the impact of crowding on demand and priorities, using a fixed-point formulation.

4.5.1 Algorithm description

Algorithm 2: The passenger assignment algorithm.

```
Input: Set of passengers $p \in P$
       Passenger priority list $\Gamma : P \to \mathbb{R}$
       Timetable graph $G(V, A)$
Output: Passenger flows $f^p : A \to \{0, 1\}, \forall p \in P$

1  $O \leftarrow$ An ordering of $P$ according to $\Gamma$, breaking ties arbitrarily
2  $f^p(a) \leftarrow 0, \forall a \in A, \forall p \in P$

3  while $O \neq \varnothing$ do
4      Let $p$ be the first passenger in $O$ (i.e., the passenger with highest priority)
5      $O \leftarrow O \setminus \{p\}$
6      Let $o_p$ and $d_p$ be the origin and destination nodes of $p$, respectively
7      Obtain $SP \subset A$, the set of arcs in the shortest path between $o_p$ and $d_p$
8      foreach $a \in SP$ do
9          $f^p(a) = 1$
10         if $a \in A_{Dri} \cup A_{Wai}$ then
11             if $\sum_{p \in P} f^p(a) \geq q_a$ then
12                 $A \leftarrow A \setminus \{a\}$
```

The passenger assignment algorithm assigns the passengers on the network according to their order in the priority list. Algorithm 2 describes the pseudo-code of the method. The algorithm takes as input the elements depicted in Fig. 4.1 and returns an indicator $f^p : A \to \{0, 1\}$, which shows if passenger $p \in P$ uses arc $a \in A$ in the assignment. During the initialization phase (lines 1–2), an ordered (according to Γ) copy O of the set of passengers P is made and the flow indicator is set to zero for all arcs and for all passengers. The algorithm iterates as long as the set O is non-empty (line 3). At every iteration, the passenger $p \in O$ with highest priority is selected, and then removed from the set O (lines 4–5). In line 6, we denote by o_p and d_p the origin node and the destination node of passenger p, respectively. We then compute the shortest path (in terms of the generalized cost described by Eq. (4.1)) in the graph $G(V, A)$, from node o_p to node d_p (line 7). $SP \subset A$ denotes the set of arcs used by the passenger on his shortest path. For each arc a in the set SP, we update the indicator $f^p(a)$ to reflect the fact that passenger p uses arc a (line 9). Furthermore, if a is a driving or a waiting arc (line 10), we need to verify if the capacity of the arc is reached after the passenger is assigned on the arc (line 11). If it is the case, the arc is removed from the graph for the remaining passengers in O (line 12), that is, for passengers with lower (or equal) priority than p. The opt-out arc (with infinite capacity) linking every origin-destination pair guarantees that each passenger can always be assigned.

This algorithm has the advantage of being computationally efficient for two reasons. First, the sorting of the passengers (which determines the priorities in case of competition for the limited capacity) is done beforehand, hence the algorithm sweeps only once over the set of passengers. Second, the traveling strategy of the passengers is implemented as a shortest-path algorithm in each passenger graph. The latter being directed and acyclic, the shortest-path search can be performed in linear time. Dijkstra's algorithm is used to obtain the shortest path arcs. Hence, it is not necessary to construct the set $\Omega(o_p, d_p)$ of all paths linking the origin to the destination of passenger p. It is important to note that the route choice model in line 7 of Algorithm 2 can easily be replaced by a more complex one. If a probabilistic model is chosen, the assignment procedure becomes a simulator. The latter is not an issue in our framework, as the system needs to be simulated anyway when the priority lists are generated from a probabilistic model, as described in Section 4.5.2.

Further, the algorithm is guaranteed to terminate after $|P|$ iterations, thanks to the opt-out arc connecting every origin and destination node. Indeed, in the worst case, even if all driving arcs are removed from A, passenger p can still use the opt-out arc between o_p and d_p.

We conclude this description by noting that removing saturated arcs from the passenger graph of subsequent lower-priority passengers is a way to enforce hard priority rules among passengers. However, slightly different priority lists (e.g., due to stochasticity) may lead to very different passenger flows, due to the fact that capacity constraints are binary. We discuss the variability across realizations for a given priority specification in detail in Section 4.6.

4.5.2 Simulation of the assignment procedure

The assignment itself, as described in Algorithm 2, is purely deterministic (because the priority specifications are exogenous). However, both the traveler demand and the priority lists may be stochastic. In that case, the assignment procedure must be embedded in a simulation. Therefore, the passenger assignment algorithm is run (i.e., the system is simulated) multiple times, each time with the same timetable, but stochastically different passenger demand and passenger priority lists. In the following, we denote by r each realization of the simulation algorithm. We also define the flow indicator $f_r^p : A \to \{0, 1\}$, which shows if passenger $p \in P$ uses arc $a \in A$ in the r-th realization of the assignment simulation. Based on the flow indicator, we define several indicators that evaluate the performance of the assignment. The distribution of these indicators over the realizations can then be analyzed in detail, including the calculation of their mean, variance and various quantiles. R stands for the total number of realizations in the following.

4.5.3 Fixed-point formulation

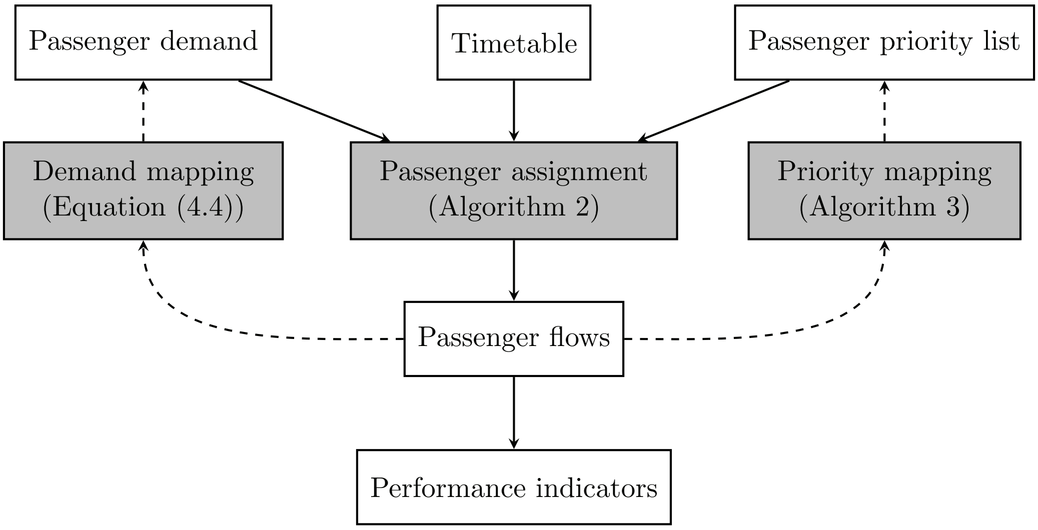

Passenger demand
Timetable
Passenger priority list
Demand mapping (Equation (4.4))
Passenger assignment (Algorithm 2)
Priority mapping (Algorithm 3)
Passenger flows
Performance indicators

Figure 4.2: Fixed-point problem of the capacitated passenger assignment.

The general framework for the passenger assignment algorithm presented in Fig. 4.1 assumes that the inputs (passenger demand and priority list) of the model do not depend on its output (passenger flows). In reality however, passenger flows can have a significant impact on passenger decisions. For instance, crowding has been shown to affect route choice in public transportation systems (see, e.g., Tirachini et al. (2013)). Also, it may happen that a passenger cannot board a vehicle that is full, because of the choices of the other passengers. The passenger priority list can also depend on passenger flows, thus introducing endogeneity. The rule giving priority to on-board passengers over boarding passengers, commonly used in the literature, is a classical example of an endogenous priority rule.

The dashed arrows in Fig. 4.2 depict the fixed-point problem that may arise in the case where the passenger demand and/or the priority list depend(s) on the outcome of the assignment. In the following, we define the fixed-point problem related to the passenger flow variables. Let $\mathbf{f} = \{f^p(a), \forall p \in P, \forall a \in A\}$ be the vector of passenger flows. We introduce the following mappings in order to describe how the passenger route choice and the priority list depend on the assignment:

Passenger demand mapping Crowding effects can be taken into account in our framework by including flow-dependent attributes in the generalized cost function given by Eq. (4.1). In order to model the discomfort level associated with travel in crowded conditions, travel time multipliers $\gamma(\mathbf{f})$ for in-vehicle and waiting time are introduced, as follows:

$$C^p_\omega(\mathbf{f}) = \gamma(\mathbf{f}) \cdot (VT^p_\omega + \beta_1 \cdot WT^p_\omega) + \beta_2 \cdot NT^p_\omega + \beta_3 \cdot ED^p_\omega + \beta_4 \cdot LD^p_\omega. \tag{4.4}$$

We use the values of Wardman and Whelan (2011), reported in Table 2.7, where the multipliers depend on the load factor of a transit vehicle, defined as the ratio between on-board passengers and the total number of seats. To incorporate these values into our framework, we define the multiplying factors based on the load factor of the arc they refer to:

$$\gamma(\mathbf{f}) = \gamma\left(\frac{\sum_{p\in P} f^p(a)}{s_a}\right),$$

where s_a is the total number of seats on driving or waiting arc $a \in A_{Dri} \cup A_{Wai}$.

Passenger priority mapping Endogenous passenger priority specifications, which depend on the passenger flow patterns, can also be used in our framework. We illustrate the procedure with the classical endogenous rule, giving priority to passengers on-board a train over passengers wishing to board. In this case, the flow-dependent passenger priority list $\Gamma(\mathbf{f})$ is constructed in the following manner.

1. We define a priority graph $\mathcal{G}$, where each node of the graph represents a passenger, and the existence of a directed arc from node p to node q in the graph indicates that passenger p has priority over passenger q. In our case, we assume that a passenger who is on-board a train has priority over every subsequent boarding passenger. Algorithm 3 formalizes the construction of the priority graph. Note that $\mathcal{G}$ defines only a partial order of the passengers, as the on-board priority might not apply to some pairs of passengers (e.g., the priority between two passengers who do not take the same train is irrelevant).

Algorithm 3: Partial passenger sorting.

```
Input: Set of passengers $p \in P$
       Passenger flows $f^p : A \to \{0,1\}, \forall p \in P$
       Timetable graph $G(V, A)$
Output: Priority graph $\mathcal{G}$

Sort arcs in $A_{Wai}$ according to time of their start node
foreach $a \in A_{Wai}$ do
    Let $a' \in A_{Dri}$ be the driving arc following $a$
    foreach $p \in P : f^p(a) = 1$ do
        foreach $q \in P : f^q(a') = 1$ and $f^q(a) = 0$ do
            Add arc $(p, q)$ to $\mathcal{G}$
```

2. A total order of the passengers is obtained by performing a topological sorting of $\mathcal{G}$ (see, e.g., Cormen et al., 2009). Note that the topological sorting can only be performed on an acyclic graph. As pointed out in Voegeli (2014), the problem of cycles appears when two different train lines share a least two stops but use different tracks in between. This situation is possible, but not common. In particular, it does not arise in our case study. In the general case, cycles have to be arbitrarily broken by removing one of the arcs

3. The topological sorting of $\mathcal{G}$ then defines the order in the priority list $\Gamma(\mathbf{f})$.

Given the mappings $C^p_\omega(\mathbf{f})$ and $\Gamma(\mathbf{f})$ above, we can define the fixed-point problem as follows. Let $\mathcal{A}(C^p_\omega(\mathbf{f}), G(V,A), \Gamma(\mathbf{f}))$ be the output of Algorithm 2, given a passenger flow $\mathbf{f}$. Then, a passenger flow $\mathbf{f}^*$ is said to be a fixed-point if

$$\mathcal{A}(C^p_\omega(\mathbf{f}^*), G(V,A), \Gamma(\mathbf{f}^*)) = \mathbf{f}^*. \tag{4.5}$$

Many algorithms exist in the literature in order to solve the fixed-point problem defined by Eq. (4.5) (see, e.g., Bottom, 2000; Magnanti and Perakis, 2004; Bierlaire and Crittin, 2006). Here we simply apply Banach iterations, but any more involved algorithm can easily be implemented. Let $\mathbf{f}_{(n)}$ be the passenger flow vector after the nth fixed point iteration. In iteration $n+1$, the generalized cost and the priority list are computed from $\mathbf{f}_{(n)}$, using the mappings defined above:

- $C^p_{\omega,(n+1)} = C^p_\omega(\mathbf{f}_{(n)})$,
- $\Gamma_{(n+1)} = \Gamma(\mathbf{f}_{(n)})$ is given by the priority graph $\mathcal{G}_{(n)}$, constructed from $\mathbf{f}_{(n)}$.

In this discussion, it is important to distinguish nonatomic and atomic user equilibria. In the former, each "commodity" (i.e., passenger) represents a large population of individuals, each of whom controls a negligible amount of traffic; in the latter, each commodity represents a single passenger who must route a significant amount of traffic on a single path. It can be shown that an equilibrium does not always exist in the atomic case (see, e.g., in Nguyen et al., 2001, the modeling discussion on *additional versus switching* passengers). As the fixed-point in Eq. (4.5) defines an atomic equilibrium, it may not exist, due to the discrete nature of the flow variables and of the priority list. We therefore introduce the following stopping criterion for the algorithm, in order to obtain a passenger flow vector which is *not too different* from one iteration to the next:

$$\Delta(\mathbf{f}_{(n)}) = \frac{\|\mathcal{A}(C^p_\omega(\mathbf{f}_{(n)}), G(V,A), \Gamma(\mathbf{f}_{(n)})) - \mathbf{f}_{(n)}\|_2}{\|\mathbf{f}_{(n)}\|_2} \leq \varepsilon, \tag{4.6}$$

where $\|\mathbf{f}_{(n)}\|_2$ is the Euclidean norm

$$\|\mathbf{f}_{(n)}\|_2 = \sqrt{\sum_{a \in A} \left[\left(\sum_{p \in P} f^p_{(n)}(a) \right)^2 \right]}$$

and ε is a parameter whose value depends on the precision required for the fixed-point problem.

4.6 Case study

4.6.1 Case description

We illustrate the algorithm on the network of the Swiss case study described in Section 2.3.2. All of the computational experiments were performed on a computer with a 2.4 GHz Intel Core i7 processor and 8 GB of RAM. The algorithms were implemented in Java. On average, one realization of the assignment algorithm (Algorithm 2) runs in about 0.2 seconds. For every priority specification, the passenger assignment is simulated a thousand times (i.e., $R = 1000$). Note that, for specification M, the computational time is slightly longer, because the set $\Omega(o_p, d_p)$ of all paths from origin to destination of every passenger needs to be constructed (once) to define the priority list. In order to limit the size of $\Omega(o_p, d_p)$, we forbid paths with the following properties: (a) paths passing twice through the same station, (b) paths where passengers transfer twice in the same station, (c) paths where passengers transfer more than twice in total. Using this strategy, and given the fact that $G(V, A)$ is sparse, the size of the set of paths is limited to about 20–100 paths per passenger.

The generalized travel time of the passengers is computed using the weights given in Table 2.6. The cost of the opt-out arc is the time horizon (four hours). We impose a minimal transfer time m of four minutes and a maximal transfer time M of fifteen minutes.

4.6.2 Results

The discussion of the results follows a topdown pattern: we begin by presenting aggregate passenger satisfaction indicators, such as average generalized travel time and unsatisfied demand, in Fig. 4.3. These allow us to compare the overall performance of the five priority specifications. In Fig. 4.4, we then investigate the variability among passengers for the same priority specification. In other words, instead of aggregating the results over passengers, we now consider each passenger and aggregate the performance indicators over the R realizations of the simulation. Finally, we present performance indicators at the origin-destination level in Figs. 4.5 and 4.6.

Fig. 4.3 presents, for the five priority specifications defined in Table 4.2, boxplots of the distributions of the following aggregate passenger satisfaction indicators:

- *Average generalized travel time* (in minutes), defined as the average (over $|P|$ passengers) of the generalized travel time, for passengers who do not take the opt-out path. More formally,

$$AVG(r) = \frac{1}{|P|} \sum_{p \in P} \sum_{a \in A \setminus A_{Opt}} f_r^p(a) \cdot c_a^p, \quad \forall 1 \leq r \leq R.$$

- *Unsatisfied demand*, defined as the number of passengers that are not served by the system, i.e., that need to take the opt-out path. Mathematically,

$$UNS(r) = \sum_{p \in P} \sum_{a \in A_{Opt}} f_r^p(a), \quad \forall 1 \leq r \leq R.$$

- *Average generalized passenger delay* (in minutes), defined as the average (over $|P|$ passengers) of the difference between the generalized cost of the path taken by a given passenger and the generalized cost of the shortest path for this passenger if capacity constraints are ignored. More formally,

$$DEL(r) = \frac{1}{|P|} \sum_{p \in P} \left(\sum_{a \in A} f_r^p(a) \cdot c_a^p - C_{\omega_1^p}^p \right), \quad \forall 1 \leq r \leq R.$$

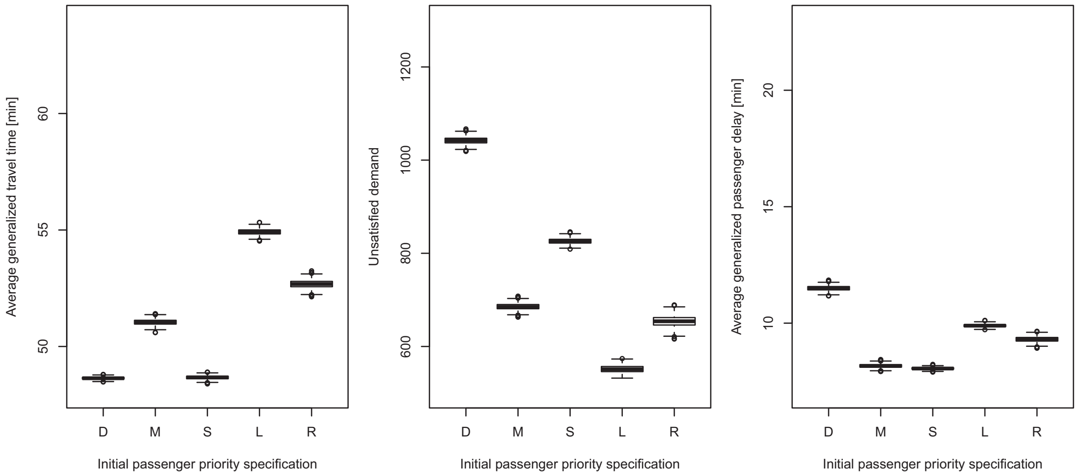


Figure 4.3: Distributions of aggregate passenger satisfaction indicators AVG(r), UNS(r) and DEL(r); for the five priority specifications.

The boxplots present the 25th (Q_1), 50th and 75th (Q_3) percentiles of the distributions of the three indicators over R realizations. The upper and lower whiskers are located at $\min(\max(\cdot(r)), Q_3 + \frac{3}{2}IQR)$ and $\max(\min(\cdot(r)), Q_1 - \frac{3}{2}IQR)$ respectively, where $IQR = Q_3 - Q_1$ is the interquartile range. The first observation that can be made for all specifications is the remarkable stability of the aggregate indicators across simulations. Indeed, the interquartile range shows limited deviation from the median of the distribution and only very few outliers (i.e., realizations situated outside the whiskers) are reported. Even in the pure random case (specification R), the output of the R simulations seems to be fairly robust.

Based on this observation, we can compare the performance of the different priority specifications on an aggregate level. Specifications D and S are the ones with the lowest average travel time. At the same time, they exhibit the highest number of unserved passengers. For priority specification D, 1042 passengers (6.9% of the total demand) on average need to take the opt-out path, whereas 826 (about 5.5% of the total demand) take it under priority specification S. Priority specifications D and S can be seen as "greedy" assignment specifications, where passengers who arrive first (D) or who travel for a short time (S) are prioritized. It corresponds to a situation where the average travel time is minimized, but the number of passengers who are worse off is substantial. Also, priority specification D has the highest value of average passenger delay. On the other hand, priority specification L has the highest average travel time and the lowest amount of unsatisfied demand. It can therefore be viewed as a priority specification that reduces the worst cases for the passengers. The price to pay is that, on average, the travel time is the highest among all priority specifications. Finally, the performance of priority specification M is a compromise between the "greedy" specifications D and S, and priority specification L. It is also a priority specification with low average passenger delay.

The aggregated passenger satisfaction indicators shown in Fig. 4.3 exhibit very little variability across R realizations. By contrast, Fig. 4.4 indicates a much higher variability for the travel time of every individual passenger across the realizations. The thick black line in the five figures (one for each priority specification) represents the average generalized travel time for every passenger (defined as $\frac{1}{R}\sum_{1\leq r\leq R}\sum_{a\in A} f^p_r(a) \cdot c^p_a, \forall p \in P$), ordered by increasing average travel time. Associated with the mean, we depict in light gray the .25 and .75 quantiles of the distributions. Three patterns of variability can be observed: (i) for priority specifications L and R, the variability appears to be fairly equally distributed among all passengers (the interquartile range increases smoothly with the mean travel time); (ii) for priority specifications D and M, passengers with higher average travel time exhibit a significantly higher variability in their travel times than passengers with lower average travel time; (iii) for priority specification S, the variability is extremely low (the assignment is almost deterministic).

It is interesting to note that the variability exhibited by priority specifications S and L

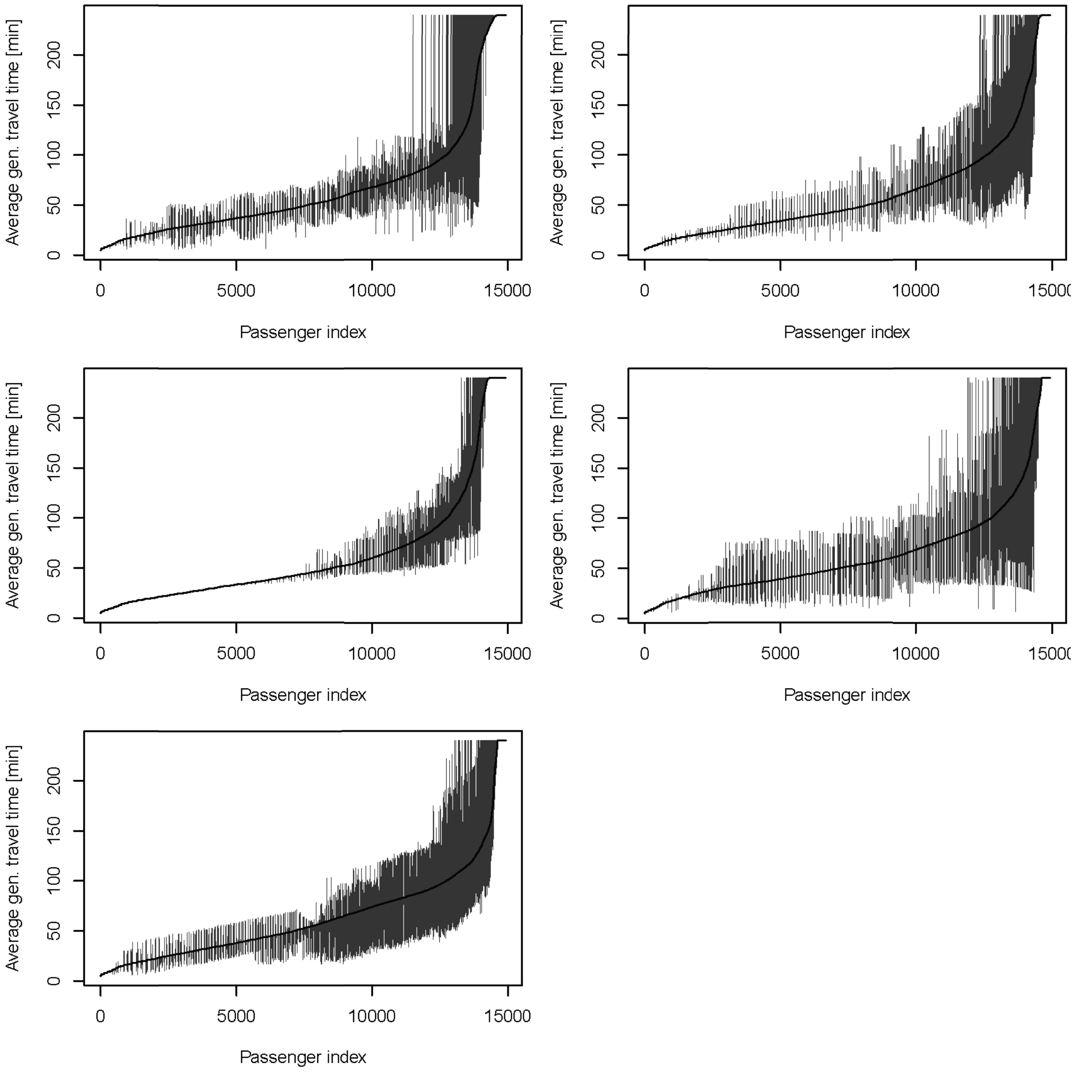


Figure 4.4: Mean, .25 and .75 quantile (over R realizations) of generalized passenger travel time for the five priority specifications (D top left, M top right, S mid left, L mid right, R bottom left).

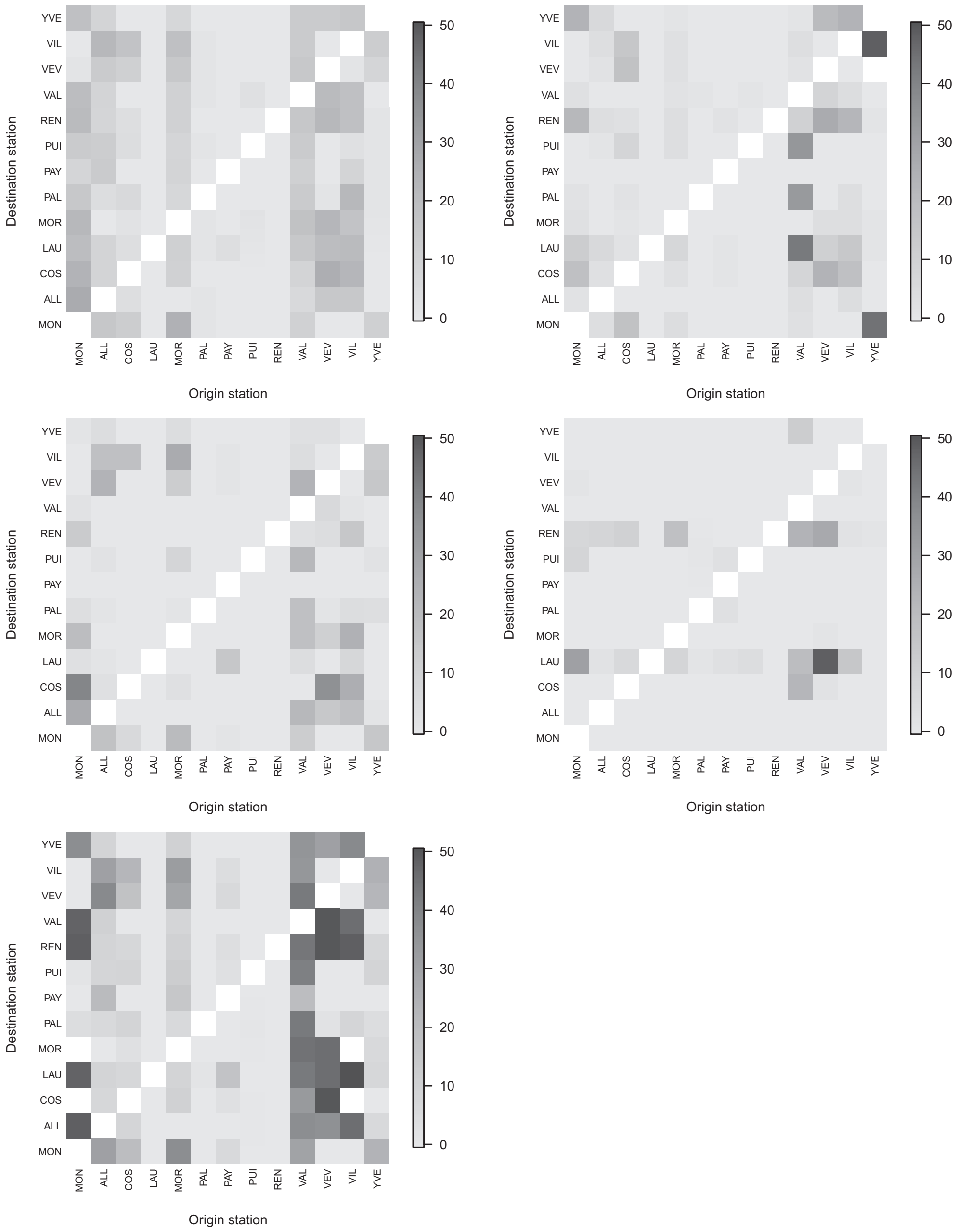


Figure 4.5: Average (for every origin-destination pair) of standard deviation (over R realizations) of generalized travel time, for the five priority specifications (D top left, M top right, S mid left, L mid right, R bottom left).

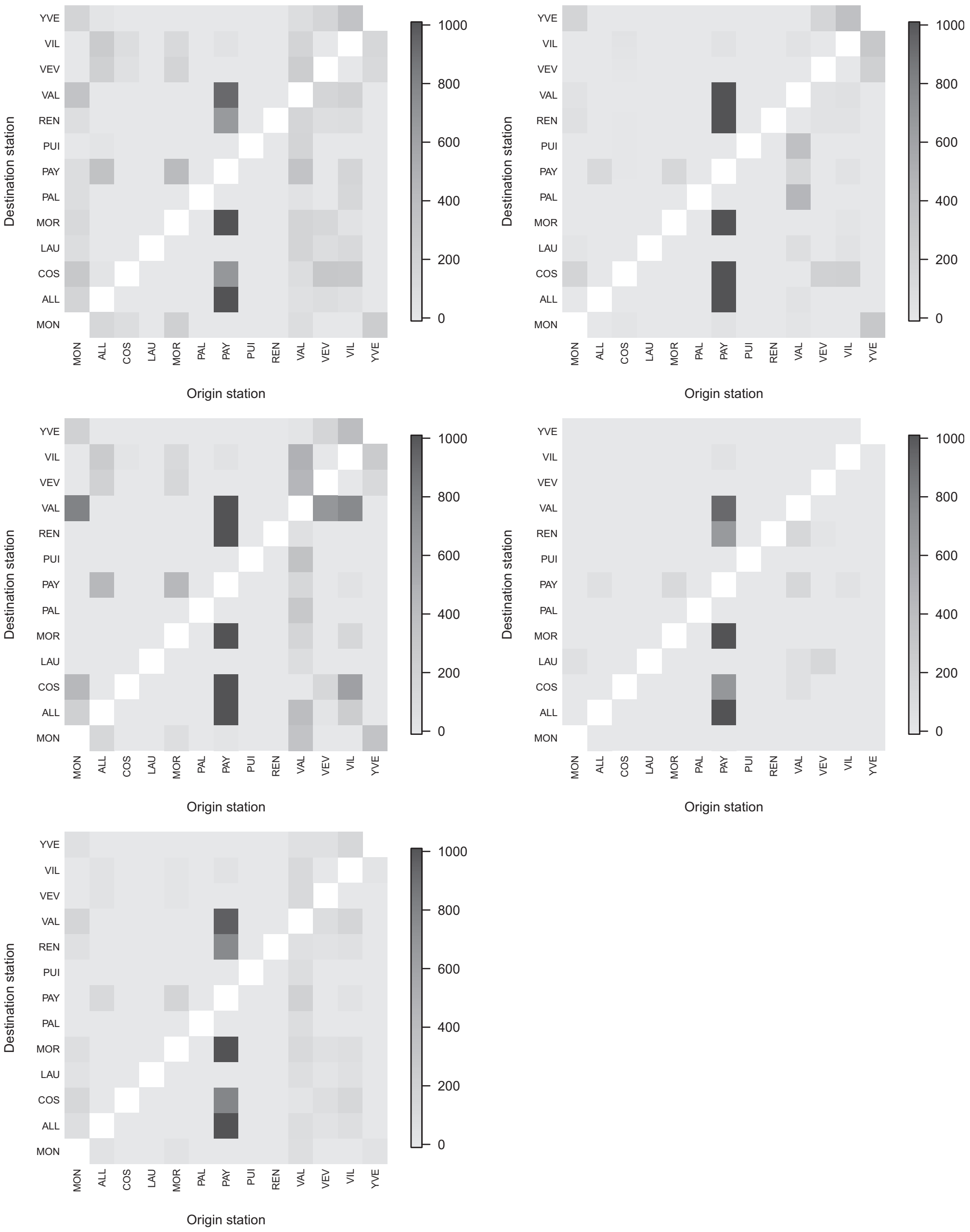


Figure 4.6: Average (for every origin-destination pair) of average (over R realizations) unsatisfied demand, for the five priority specifications (D top left, M top right, S mid left, L mid right, R bottom left).

is not symmetric. For the former, by giving priority to short origin-destination pairs, it is expected for priority specification S to obtain the lowest interquartile range for low values of average travel time. By contrast, for priority specification L, one might expect low values of interquartile range for higher values of average travel time. This is not the case however. Indeed, most passengers appear to have a certain range of variability. This can be explained by the fact that if priority is given to long-distance passengers, short-distance passengers will usually be able to find another way to travel (incurring an acceptable increase in travel time). If however short-distance travellers "clog up" trains, long-distance travellers may not be able to find another way to travel through the system in any reasonable time.

The difference in variability between the "greedy" priority specifications D and S is also noteworthy. Fig. 4.3 showed the highest level of unsatisfied demand for these two priority specifications. The involved passengers are very different however. If priority specification S is applied, 541 passengers take the opt-out path in every realization. In the case of priority specification D, only 311 passengers will always take the opt-out path. By comparing these values to the previously reported average levels of unsatisfied demand, we conclude that, for priority specification S, about 65.4% of unsatisfied demand is constituted of passengers that have no other choice than taking the opt-out path, whereas this number drops to 29.8% for priority specification D.

These insights may encourage railway operators to enforce a priority specification giving priority to short origin-destination pairs (similar to S). Thanks to an almost deterministic assignment, it is easy to pinpoint the passengers that are worse off under this priority specification. Also, low average travel times and low delays are an advantage of this priority specification.

Results at the origin-destination level are presented in Figs. 4.5 and 4.6. Each cell in these plots depicts the average over all passengers on a particular origin-destination pair. Fig. 4.5 compares the standard deviation (over R realizations) of the generalized travel time, for the five priority specifications. Formally, the value of a cell with origin $o \in S$ and destination $d \in S$ is given by

$$\frac{1}{|P(o,d)|} \sum_{p \in P(o,d)} \sigma \left(\sum_{a \in A \setminus A_{Opt}} f_r^p(a) \cdot c_a^p \right),$$

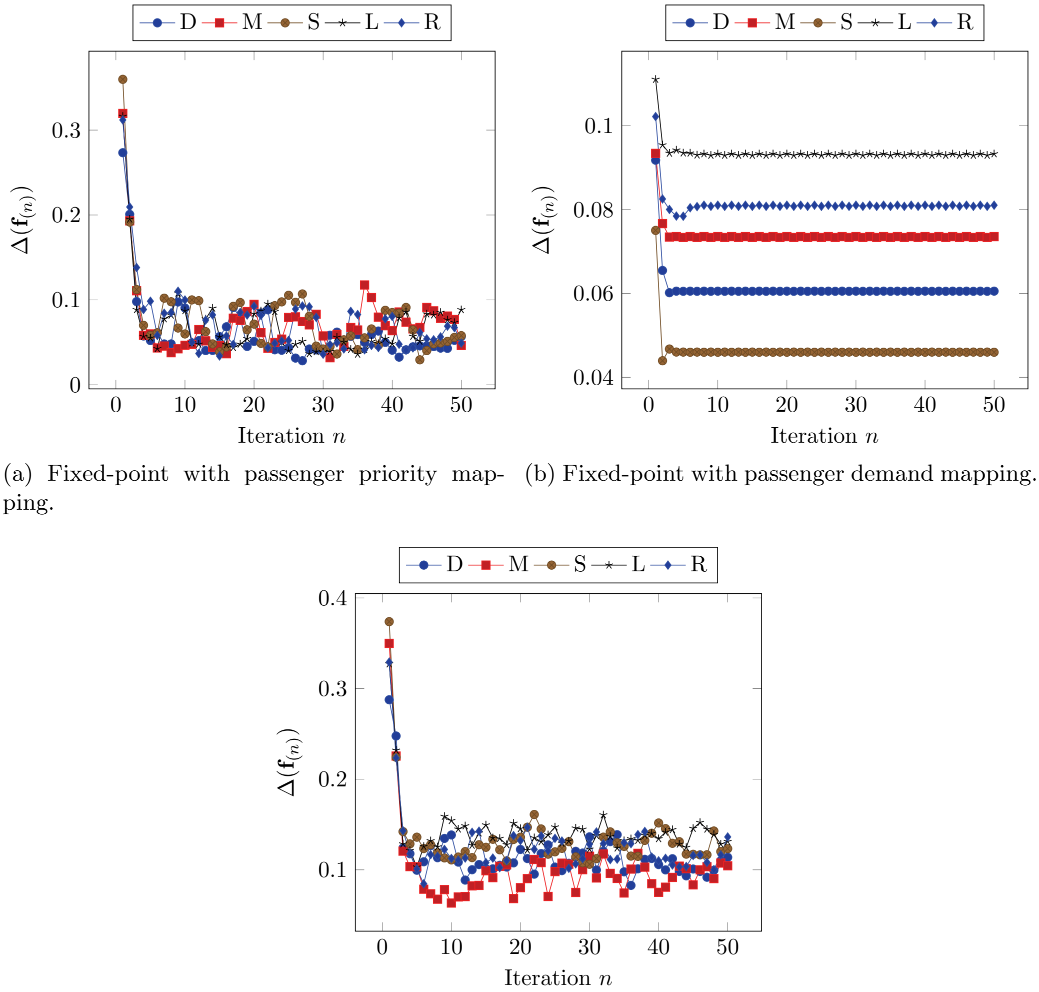
where $P(o,d) \subset P$ is the set of passengers traveling from o to d and $\sigma(\cdot)$ indicates the standard deviation of the distribution over R realizations. One can observe that the standard deviation of travel times is similarly distributed across all origin-destination pairs, except for two noteworthy points. First, for priority specification L, the standard deviation is low for almost all origin-destination pairs, except for passengers ending their journey in REN or LAU (the two main stations of the case study), and for passengers starting their journey in VAL. This priority specification might therefore not be appropriate in the morning rush hours, where passengers are assumed to travel towards the main stations, with little uncertainty on the travel time they can expect. Second, the origin-destination pairs originating in station LAU or REN all have a standard deviation of 0, for any priority specification, showing that there is more than enough capacity in the trains for them.

Fig. 4.6 compares the average (over R realizations) unsatisfied demand, for the five priority specifications, at the origin-destination level. Mathematically, the value of a cell with origin $o \in S$ and destination $d \in S$ is given by

$$\frac{1}{|P(o,d)|} \sum_{p \in P(o,d)} \left(\frac{1}{R} \sum_{1 \leq r \leq R} UNS(r) \right).$$

A clear pattern can be observed: origin-destination pairs originating from station PAY or VAL have a much higher level of unsatisfied demand. This pattern is even present for priority specifications L and R, where the overall level of unsatisfied demand is lowest. The explanation behind this phenomenon lies probably in the structure of the network: PAY and VAL are both stations situated at the periphery (see Fig. 2.4), therefore the train offer is less developed than in more central stations of the network. In accordance with what was observed in Fig. 4.5, there is no unsatisfied demand originating from station LAU or REN. In addition, the origin-destination pairs originating in the following stations are always served: PAL and PUI for priority specification S, COS and YVE for priority specification L.

4.6.3 Solving the fixed-point problem



(a) Fixed-point with passenger priority mapping.

(b) Fixed-point with passenger demand mapping.

(c) Fixed-point with both mappings.

Figure 4.7: Termination of the fixed-point problem in the first realization, for the five initial priority specifications (D, M, S, L, R).

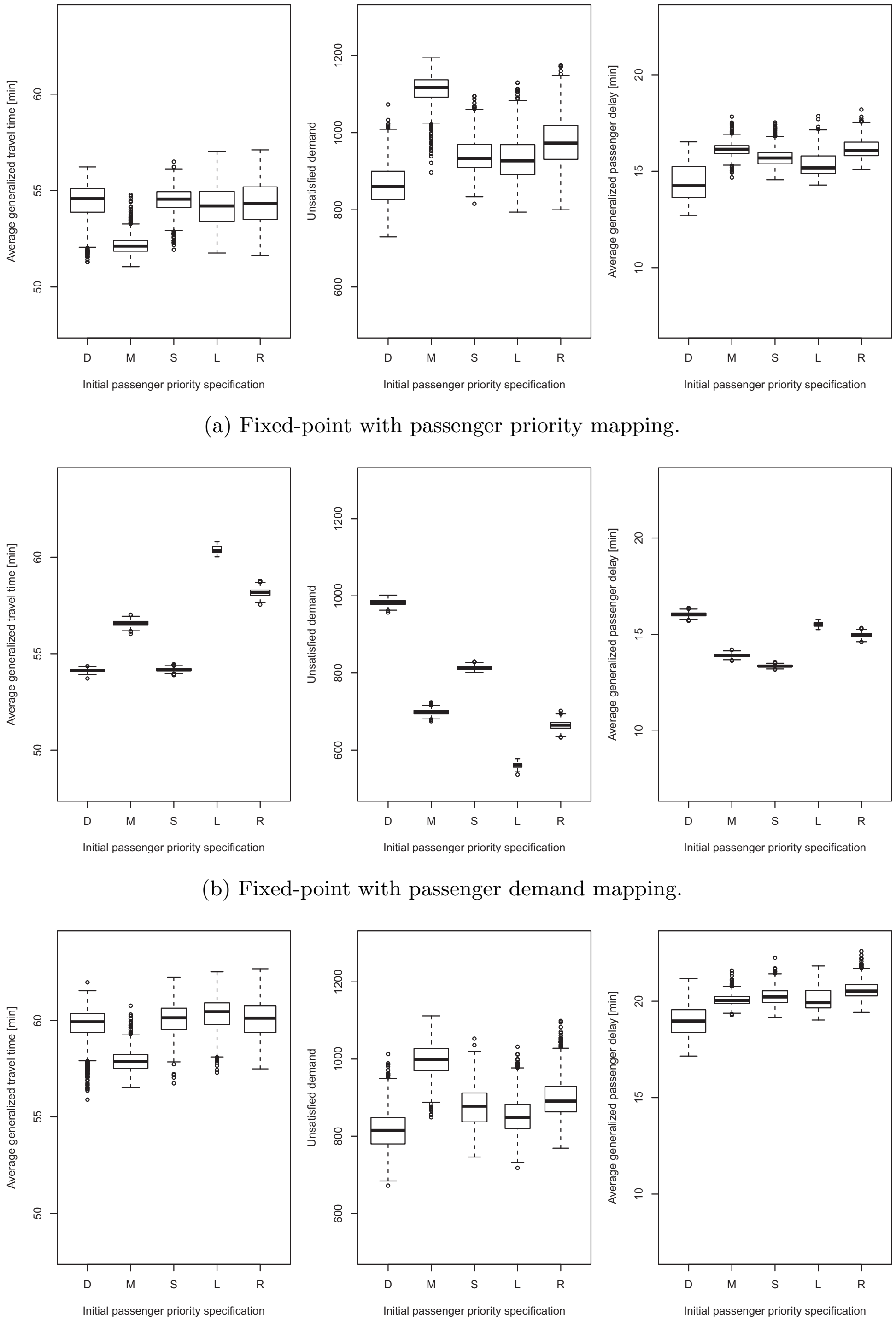
If we consider endogenous passenger demand or endogenous passenger priorities, a fixed-point problem has to be solved for every realization r. However, due to the discrete nature of the variables, this fixed-point may not exist. We use the gap function $\Delta(\mathbf{f}_{(n)})$ defined in Eq. (4.6), which indicates that the passenger flow vector has become not too different from one iteration to the next, to terminate the fixed-point iterations. In order to obtain an adequate value of the parameter ε for the stopping criterion, we depict in Fig. 4.7 the latter's evolution for the first realization of the random variables. The priority specification (D, M, S, L or R) indicates which exogenous priority specification is used as an initial solution to the fixed-point problem. Three fixed-point problems are investigated, according to the mappings defined in Section 4.5.3: in Fig. 4.7a, the fixed-point where the passenger priorities are updated at every iteration is considered; Fig. 4.7b depicts the fixed-point where the passenger demand is updated; whereas Fig. 4.7c shows the evolution of the gap function when both mappings are updated. In the three cases, it can be observed that the fixed-point cannot be solved exactly in fifty iterations, as the gap remains strictly positive. This behavior is expected with discrete variables. However, the gap drops in less than five iterations respectively below 5%, 10% and 12% in the first, second and third case. We assume that this precision is sufficient and set ε to 0.05 for the priorities fixed-point, to 0.08 for the demand fixed-point, and to 0.11 for the fixed-point where both mappings are updated, irrespective of the initial priority specification. We also set N, the maximal number of fixed-point iterations per realization, to fifty. If the stopping criterion is not met after N iterations (i.e., $\Delta(\mathbf{f}_{(N-1)}) > \varepsilon$), the realization is discarded in the results below.

Table 4.3: Fixed-point statistics after $R = 1,000$ realizations.

Updated mapping(s)	Initial priority specification	ε	# terminated realizations	Average number of fixed-point iterations	Average runtime [s]
Priorities	D	0.05	959	7.94	4.90
	M		972	7.46	5.09
	S		978	7.70	4.92
	L		991	6.98	4.81
	R		979	7.17	5.37
Demand	D	0.08	1,000	2.00	0.14
	M		1,000	2.03	0.14
	S		1,000	1.15	0.14
	L		249	3.10	0.15
	R		571	2.73	0.13
Both	D	0.11	989	5.06	5.17
	M		1,000	4.06	5.18
	S		730	7.12	4.98
	L		722	5.79	5.01
	R		917	5.81	5.61

Table 4.3 summarizes statistics about the fixed-point iterations across the realizations. The number of terminated realizations indicates how often the algorithm terminated (with respect to Eq.(4.6)), out of the total number of realizations. Except for initial priority specifications L and R for the demand fixed-point (respectively, S and L for the fixed-point with update of both mappings), the algorithm terminates in almost all realizations. The average number of fixed-point iterations is low in all cases, confirming the quick termination observed in Fig. 4.7. The last column indicates the average runtime of one fixed-point iteration. The need for a topological sorting of the priority graph explains the higher values for the first and the third fixed-point problems.

Fig. 4.8 presents the boxplots of the distributions of the three performance indicators defined in Section 4.6.2, over realizations where the fixed-point problem terminated (the



(a) Fixed-point with passenger priority mapping.

(b) Fixed-point with passenger demand mapping.

(c) Fixed-point with both mappings.

Figure 4.8: Distributions of aggregate passenger satisfaction indicators AVG(r), UNS(r) and DEL(r); for the five initial priority specifications (D, M, S, L, R). The width of the boxplots indicates the relative number of realizations where the fixed-point terminated.

width of the box is proportional to the number of terminated realizations). Figs. 4.8a–4.8c compare the results for the three fixed-point problems. In Fig.4.8a, it can be observed that the interquartile range is significantly higher than when no fixed-point is considered (compare with Fig. 4.3). This can be explained by the fact that the endogenous priority specification (giving priority to on-board passengers) assigns a passenger taking the train at an "earlier" station of a train line before a passenger taking the train at a "later" station. In many cases, the passenger with lower priority is therefore not able to board the saturated train, and an alternative path with a reasonable travel time may not exist, especially if it is a short origin-destination pair. The demand structure of the case study (50% of the passengers travel to LAU and 20% to REN) implies that many of these cases arise: for instance, passengers from stations VIL, MON, VEV, PUI and PAL that travel to REN have priority over passengers from LAU to REN. Due to the discrete nature of the priorities, the level of unsatisfied demand can thus vary considerably. The variability of the two other indicators can be explained by the high cost of the opt-out arc associated with unsatisfied demand. The results show that the median value of the distributions is almost independent of the priority specification used as an initial solution to the fixed-point problem, except for specification M. In general, the performance of the system worsens from the passenger perspective when the priorities fixed-point is considered (higher average generalized travel time, higher level of unsatisfied demand and higher average delay), motivating the implementation of a policy that explicitly controls the priority of passengers. By contrast, the interquartile range in Fig. 4.8b is as low as when no fixed-point is considered (Fig. 4.3), even for the initial priority specifications L and R, where only a fraction of the realizations terminate. For the demand fixed-point, the results are much more dependent on the initial priority specification. The values of the performance indicators are higher than without the fixed-point specification because the travel time multipliers introduced to model crowding are greater than one for load factors above 100%. Finally, Fig. 4.8c shows a hybrid combination of the effects when both mappings are updated at every realization: the median values of the indicators are higher and almost independent of the original priority specification, and the interquartile range increases significantly.

4.6.4 Discussion

In conclusion, results show a remarkable stability of aggregate passenger satisfaction indicators (such as passenger delay, average travel time or level of unsatisfied demand) from one simulation run to the next, when the fixed-point problem is disregarded. By comparing the five different priority specifications, a trade-off is highlighted: giving priority to long-distance passengers minimizes the level of unsatisfied demand, while giving priority to short-distance or early passengers minimizes the average travel time of all passengers. At the individual passenger level however, substantial variability is observed. By analyzing in detail the outcome of the passenger assignment model for every priority specification, we are able to identify clearly who gains and who loses in each case.

When considering passenger route choice or priority specifications that depend on the outcome of the assignment, an additional fixed-point problem needs to be solved. It was shown how our framework is able to accommodate such a specification for the passenger flows. In particular, we show that endogenous priority lists can easily be embedded in a fixed-point specification, as illustrated by the case study. It is, however, impossible to find an actual fixed-point, due to the discrete nature of the variables and the fact that an atomic equilibrium is defined. Convergence is therefore measured by means of a gap function, and reasonably low values are obtained in only a few iterations.

Results show that passengers experience a considerably lower level of service in the case where an endogenous priority specification and/or crowding are considered. The aggregate satisfaction indicators are consistently higher, more volatile, or both. The fixed-point problem where both mappings are updated simultaneously is the closest one to current practice in railway networks, as it considers crowding and mimics how passengers board trains in a FIFO manner. Unfortunately, it corresponds to the worst passenger satisfaction indicators reported in this chapter. Railway operators can use these insights in order to reduce the overall passenger inconvenience, by imposing specific priorities. For instance, one suggestion would be to give priority to short-distance and early arrival passengers (thus decreasing the average travel time), and to provide alternative solutions for those passengers who are disrupted by this priority specification.

Because of the variety of methodologies in the literature, and because of the size of solvable instances in the reported case studies, it is difficult to compare our results with a framework of the literature explicitly. In comparison with other works, our framework is one of the few that have been extensively tested on a case study of a realistic size. For instance, works such as Nguyen et al. (2001); Sumalee et al. (2009); Hamdouch et al. (2011, 2014) provide numerical case studies, acknowledging that algorithmic enhancements are necessary for application to a real case study.

In the following, we discuss how our framework performs with respect to three major issues identified by Hamdouch et al. (2014) to assess the applicability of a model in real-life examples: computational resource requirements, convergence conditions and computational time.

Our formulation is very lean in terms of memory usage, as space-time nodes are only included in the graph for departure and arrivals of transit vehicles from/to a station. By contrast, some formulations in the literature require variables indexed by station, time, transit line and travel strategy (Hamdouch et al., 2014). Obviously though, the size of the graph grows as the number of considered transit vehicles increases, thus effectively capping the total number of transit vehicles that can be considered. The number of passengers is not limiting in our formulation, as the size of the space-time graph representing the timetable does not depend on passenger demand.

As described in Section 4.5.1, the assignment algorithm we propose is guaranteed to terminate after one sweep over the set of passengers. It needs however to be embedded in a fixed-point formulation if one assumes endogenous passenger demand or priority rules, as discussed in Section 4.5.3. Convergence issues that arise in models using MSA-type algorithms to find equilibrium solutions (e.g., Poon et al., 2004; Hamdouch and Lawphongpanich, 2008; Hamdouch et al., 2014) can therefore not be excluded in this case. These are discussed when presenting the results of the fixed-point formulations, in Section 4.6.3.

Low computational time may be the most important asset for the applicability of a methodology in a real-life context. Our framework performs well in this respect. Thanks to its simplicity, and due to the fact that the priority lists are exogenous, our framework is very fast: one run of the assignment model takes about 0.2 seconds to simulate for the case study. As described throughout this chapter, our model can nevertheless easily be complexified (e.g., including crowding or more involved route choice models). In that case, a trade-off between computational time and model complexity would appear. We have shown that our framework can also accommodate a fixed-point formulation, and that the computational time to achieve convergence remains reasonable. Moreover, our formulation relies on simulation to account for the stochastic nature of the priority lists. Obviously, the computational time increases with the number of required simulations. The results showed that aggregate passenger satisfaction indicators are stable across simulations, thus limiting the number of simulations required to compute them. A higher number of simulations would however be required if one is interested in the fate of individual passengers.

4.7 Concluding remarks

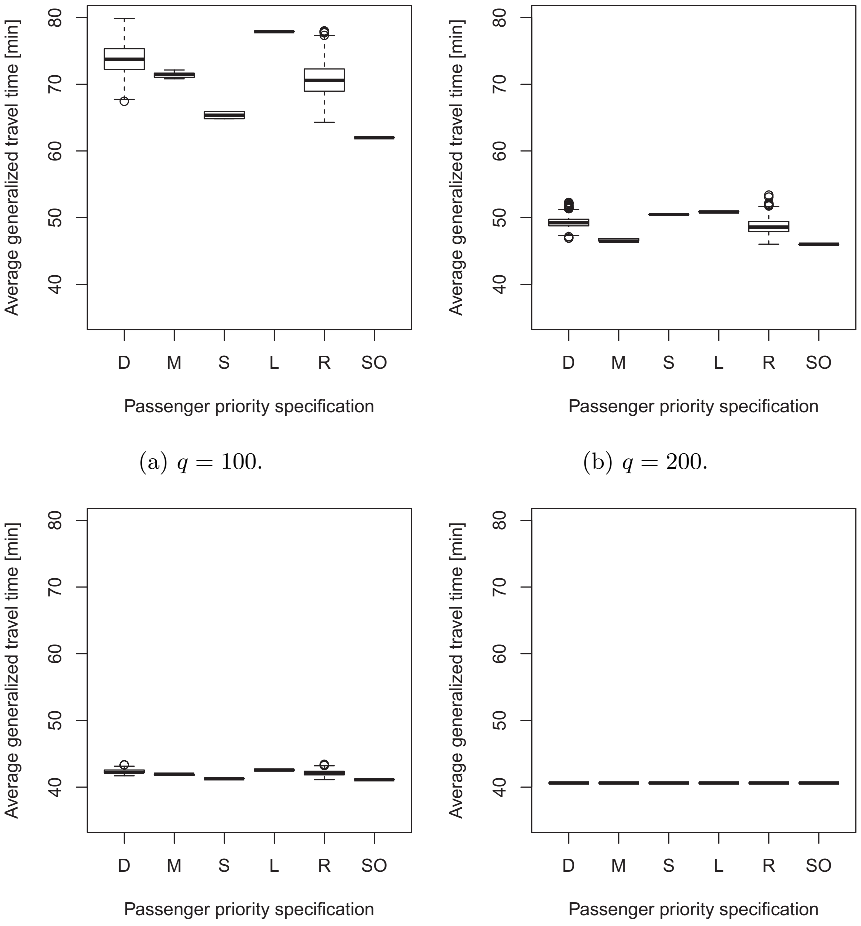
In this chapter, we introduced a new framework for the schedule-based passenger assignment problem in capacitated public transportation networks. The originality of the framework is the use of an exogenous passenger priority list in order to decide which passengers can board the train and which cannot. The priority lists are generated in advance, based on priority rules (e.g., prioritize long-distance passengers). Additionally, it was shown that the framework is flexible enough to accommodate endogenous priority specifications in a fixed-point formulation. Stochasticity appears in the generation process of the priority rules. Therefore, the assignment is run (i.e., the system is simulated) multiples times, in order to obtain the distribution of the performance indicators. Although various route choice models can be embedded in the framework, the use of a shortest path model allows for an efficient implementation of the framework.

Extensive computational experiments were performed on a realistic case study of the S-train network of Canton Vaud, Switzerland. One run of the assignment model takes about 0.2 seconds to simulate for our case study. Considering this short computational time, and the fact that a low number of simulations is sufficient (thanks to the stability of aggregate results), train operators can use the framework in practice to evaluate timetables from the passenger perspective in real-time. It can also be used in order to evaluate priority policies that the operator would like to enforce (e.g., giving priority to a set of "VIP" origin-destination pairs).

A natural extension of this work is the usage of the algorithm in an iterative optimization-simulation framework, where, in each iteration, the timetable is updated and then evaluated from the passenger perspective by the assignment model. Such a framework is introduced in the next chapter of this thesis.

Appendix

4.A Price of Anarchy



(a) $q = 100$.

(b) $q = 200$.

(c) $q = 300$.

(d) $q = 400$.

Figure 4.A.1: Price of Anarchy, for different values of capacity (q).

This appendix reports the Price of Anarchy occurring when applying the orderings defined in Table 4.2 to the Dutch case study. Fig. 4.A.1 reports the average generalized travel time, for the five priority specifications, and for the case of a system-optimal assignment (SO), as defined in Chapter 3. Similarly to Fig. 4.3, the boxplots represent the distribution over R realizations, for the five specifications. Regarding the system-optimal assignment, simulation is not necessary and the single value is thus reported.

4.B Scale factor

The scale factor μ captures the variability of the disturbances in a choice model. There are two limiting cases that result from extreme values of μ:

- As $\mu \to 0$, the variance of the disturbances approaches infinity (see Eq. (4.3)). In this case, the choice model (4.2) provides no information, so all alternatives are equally likely. In our framework, this translates to ordering the passengers completely randomly: as $\varepsilon_p \to \infty$ in (4.2), the systematic part of the importance function becomes irrelevant and passenger are sorted only using random disturbances.
- As $\mu \to \infty$, the variance of the disturbances approaches zero (see Eq. (4.3)) and a deterministic choice model is obtained because all the information in included in the systematic utilities. In our framework, this translates to ordering the passengers deterministically: as $\varepsilon_p \to 0$ in (4.2), the remaining part of the importance function is the systematic part and passengers are sorted accordingly.

These two limiting cases show that a choice of μ has to be made, and this for every priority specification, in order to control the level of randomness of the ordering. Obviously, this choice will always be arbitrary. In our case, we apply the procedure described in Algorithm 4 to determine an appropriate scale factor. The main idea is to compare the order of the passengers if they are ordered by the complete importance function (U_p), or only by its systematic part (V_p). We then impose the correlation between these two orderings to be greater than a given parameter α. We obtain the scale factors reported in Table 4.B.1 for a correlation of 95%.

Table 4.B.1: Scale parameters for the different priority specifications, with a correlation of $\alpha = 0.95$.

Priority specification ρ	μ_ρ
D	0.10
M	0.28
S	0.16
L	0.16

Algorithm 4: Determination of an appropriate scale factor for the different priority specifications.

```
Input: Set of passengers $p \in P$
       Priority rule $\rho \in \{D, M, S, L\}$
       Required correlation $\alpha$
Output: Scale factor $\mu_\rho$

$\mu_\rho := 0$
while $\mu_\rho < 10$ do
    foreach $p \in P$ do
        $V_p := V_p(\rho)$, according to Table 4.2
        $\Delta(p) = V_p$
        Draw $r \sim U(0,1)$
        $\varepsilon_p = -\frac{1}{\mu_\rho} \ln(-\ln r)$
        $U_p = V_p + \varepsilon_p$
        $\Gamma(p) = U_p$
    Let $V_\Delta$ be the vector of systematic utilities of passenger in $P$ ordered according to $\Delta$
    Let $V_\Gamma$ be the vector of systematic utilities of passenger in $P$ ordered according to $\Gamma$
    if $corr(V_\Delta, V_\Gamma) \geq \alpha$ then
        return $\mu_\rho$
    else
        $\mu_\rho := \mu_\rho + 0.01$
```

5

An adaptive large neighborhood meta-heuristic for railway timetable rescheduling

This chapter is based on the conference paper:

`Binder, S., Maknoon, Y., and Bierlaire, M. (2017). Efficient investigation of multiple dimensions of the railway timetable rescheduling problem. Proceedings of the 17th Swiss Transportation Research Conference May 17-19th, 2017.`

The work has been performed by the candidate under the supervision of Prof. Yousef Maknoon and Prof. Michel Bierlaire.

5.1 Introduction

In Chapter 3, an Integer Linear Programming formulation for the multi-objective railway timetable rescheduling problem was introduced. In order to maintain a tractable model, simplifying assumptions had to be made. Chapter 4 discussed the behavioral limitations of the model, and introduced an efficient passenger assignment algorithm for capacitated transit networks. The goal of this chapter is to apply the proposed assignment model, while extending the possibilities of the rescheduling model.

Timetable rescheduling is a difficult combinatorial optimization problem, made even harder by the consideration of multiple objectives in Chapter 3. The exact solution approach described in the latter provides optimal solutions for instances of limited size, that is, about 55 passenger groups and 20 trains in the case study. However, typical real-life instances may include thousands of passengers and hundreds of trains. Solution methodologies that can obtain "good" but not necessarily optimal solutions are therefore required for such large instances. Additionally, the solution methodology needs to be fast and efficient, in order to be applicable in a practical context. The main goal of this chapter is to propose such a solution methodology, based on the adaptive large neighborhood search (ALNS) meta-heuristic introduced by Ropke and Pisinger (2006).

The introduction of a heuristic solution method is additionally motivated by the following reasons. Laumanns et al. (2006) notes that "the necessity of determining the constraint value a priori is [...] a serious drawback of the original ε-constraint method." Indeed, the values introduced in Chapter 3 (specifically, Table 3.3) are justified from a practical point of view but remain arbitrary. The heuristic introduced in this chapter does not require constraints on the objectives, thus circumventing the need for a manual setting and tuning of the values of the epsilons. Further, train operating companies may not be interested in the exact optimal solution, but rather in a solution that is easily implementable and understandable (both from the passengers' and the operator's point of view). Our framework considers this requirement, in the sense that timetables are modified by directly applying recovery actions that are used in practice (such as the ones described in the example of Section 2.1). Morevover, this allows to maintain a higher level of control over the modification of the timetable.

Note that the iterative procedure introduced in this chapter is purely heuristic: it does not necessarily find all non-dominated solutions, thus the obtained solutions are not guaranteed to be Pareto-optimal. Several algorithms exist to find the complete set of non-dominated solutions in a multi-objective context (e.g., Laumanns et al., 2006; Özlen and Azizoğlu, 2009; Bérubé et al., 2009; Mavrotas and Florios, 2013; Kirlik and Sayın, 2014). However, as we are motivated by the limited available time in a practical context, our main interest is the efficient exploration of the frontier, and the search for solutions that make sense from a practical point of view. Finding *all* non-dominated solutions is thus out of the scope of this work.

As the literature review of Chapter 2 has shown, the number of timetable rescheduling models considering passenger demand is limited. Cadarso et al. (2013), Kroon et al. (2015), Corman et al. (2016) and Veelenturf et al. (2017) also use a heuristic iterative framework to assign passenger flows during disruptions. In this context, the main contributions of this chapter are the following:

- The introduction of a meta-heuristic based on ALNS and simulated annealing for the multi-objective railway timetable rescheduling problem.

- The definition of a rich set of practice-inspired operators that modify the timetable.
- The application of the proposed framework to a real case study for the efficient investigation of the problem's Pareto frontier.

The remainder of this chapter is structured as follows. Section 5.2 describes the problem formally, and emphasizes the differences with Chapter 3. Section 5.3 introduces the heuristic solution algorithm and Section 5.4 provides the results of the computational experiments. Finally, Section 5.5 concludes the chapter.

5.2 Problem formulation

The problem addressed in this chapter is to generate a set of "good" and practice-oriented disposition timetables, and to quantify the trade-off between the objectives in order to assist railway operators in the design of such timetables. The multi-objective timetable rescheduling framework we present in this section builds upon the work of Chapter 3. For the sake of completeness, we recall here the main notations and assumptions. The following aspects differ from the previous chapters:

- The concept of train lines is introduced, and emergency trains are assumed to be (initially) scheduled along them.
- We still assume that the duration of the disruption is known in advance, but we also consider disruptions that are shorter than the whole time horizon of the problem.
- Instead of relying on the system-optimal passenger assignment of Chapter 3, the algorithm introduced in Chapter 4 is used, with a slight modification: the cost of the opt-out arc depends on the shortest path of the corresponding OD pair.
- In order to incorporate more realistic recovery decisions, we include the possibility for trains to be turned around along a train line at a given station, as well as the scheduling of additional emergency buses between disrupted stations.
- The definition of the deviation from the undisrupted timetable is refined in order to include the above additional recovery decisions.

5.2.1 Infrastructural model

Time is discretized into $n + 1$ time intervals of length τ (typically, one minute) and we introduce the set of time steps $H = \{0, \tau, 2\tau, \dots, n\tau\}$, where $n\tau$ is the considered planning horizon. We model the railway network at a macroscopic level. The infrastructure is represented by a set of stations $s \in S$ and a set of tracks $Q \subseteq S \times S$ connecting the stations. A track $(s, s') \in Q$ is an uninterrupted railway track linking s to s' directly, without passing in any other station. We define two stations $s, s' \in S$ to be neighboring if $(s, s') \in Q$ and $(s', s) \in Q$. There are therefore two tracks connecting any two neighboring stations in the undisrupted case. Tracks can be used by trains in both directions: each track can either be used by trains running in the same direction consecutively, or it can be used by trains scheduled in opposite directions alternatively. In any case, a certain headway has to be respected if two trains are scheduled on the same track. Between two neighboring stations, the running time $t(s, s')$, in minutes, and the distance $d(s, s')$, in kilometers, are known, symmetric (i.e., $t(s, s') = t(s', s)$, $d(s, s') = d(s', s)$) and equal for all trains. Trains cannot switch tracks between stations and overtakings occur only within stations (i.e., a platform in a station can be reached from any incoming/outgoing track).

The individual trains are operated along *train lines* in the network. A train line $\ell \in L$ is a succession of neighboring stations, that are visited in a given order. All trains are initially scheduled to operate along a train line, but their course may be altered during the rescheduling process. We consider two different types of trains: *original trains* and *emergency trains*. The set of original trains K_1 contains the trains that are operated in the undisrupted timetable. Their schedule is an input to the rescheduling model. The set of emergency trains K_2 represents trains that are ready to be scheduled if needed. All emergency trains are scheduled along a train line, and n_ℓ denotes the number of emergency trains available for train line ℓ. The set of all trains is denoted by $K = K_1 \cup K_2$, and all trains in K are characterized by their capacity q_k, defined as the maximal number of simultaneous on-board passengers.

5.2.1.1 Disruption and associated recovery decisions

We define a disruption in the network as the unavailability of at least one track between neighboring stations. Multiple tracks can become unavailable at the same time, and at different locations in the network. We assume that the duration of the disruption is known in advance, and that it is less or equal to the considered time horizon. Hence, we also consider what happens after the end of the disruption (i.e., once all tracks can be used again).

In order to recover from the disruption, we consider the following potential decisions to reschedule the timetable. The technical details and the choice mechanisms of the different decisions are described in Section 5.3.3.

Cancellation A train may be fully or partially canceled. A partially canceled train is only operated on a subset of the stations of its original train line and canceled afterwards, whereas a fully canceled train is not operated at all. Observe that a full cancellation is a special case of a partial cancellation.

Delay The departure of a train at a station may be delayed up to a maximal amount of time. A train may also be delayed only for a part of its route along the line.

Advance The departure of a train may be fully or partially advanced (i.e., scheduled earlier). Original trains are not allowed to run earlier than in the undisrupted timetable, as this is usually avoided in practice because passengers might miss their planned train.

Rerouting If a train is scheduled on an unavailable track, it may be rerouted through a path that circumvents the latter. The rerouting occurs between the stations at both ends of the disruption.

Turnaround Similarly, if a train is scheduled on an unavailable track, it may turn around at the last station before the disrupted track and run in the opposite direction back to the station where it started its journey.

Emergency train For every train line, a limited number of emergency trains is available. These may be scheduled as needed.

Emergency bus If the track between two neighboring stations is disrupted, an emergency bus may be scheduled to connect the two stations directly.

Note that multiple recovery decisions can be applied to a single train. For instance, a train can be turned around, and then delayed in order to wait for connecting passengers.

5.2.1.2 Timetable feasibility

The disposition timetables designed by our framework are feasible in terms of macroscopic operational constraints. A timetable is defined to be feasible if it does not contain any infeasibility of the two following types. A *headway-related infeasibility* occurs when two trains do not respect the minimal headway between each other, when both are scheduled to run on the same track. If they are running in the same direction, a minimal headway time h needs to be respected. If two trains are scheduled to run in opposite directions on a track (s, s'), it needs to be ensured that the track is vacated by the first train before the second one can use it. Again, we consider that a minimal headway time h needs to be respected between trains. Hence, the difference between the departure times of the two trains from stations s and s' has to be greater than $h + t(s, s')$. A *disruption-related infeasibility* occurs when a train is scheduled on a track that is not available due to the disruption. More precisely, for a disruption between stations s an s', from time t_1 to t_2, a train movement is said to be infeasible if its departure time from station s or s' lies between t_1 and t_2.

The initial timetable does not contain any headway-related infeasibilities, as it is the schedule run by the operator in undisrupted situations. Therefore, at the beginning of the rescheduling process, only disruption-related infeasibilities occur due to the fact that the track has become unavailable. Later on, both types of infeasibilities may appear, simultaneously and multiple times, after applying the recovery decisions presented in the previous section.

Observe that these constraints ensure the high-level feasibility of the schedule, but disregard details such as individual track assignments in stations. Also, it is assumed that train stations have an infinite number of platforms, and that each platform can be accessed by any train.

5.2.2 Passenger assignment model

Passenger demand is assumed to be known, in the form of an origin-destination (OD) matrix. The latter describes the number of passengers entering the system at a given origin station, at a certain time, and who wish to travel to a given destination station. Based on the OD matrix, a passenger p is denoted by a triplet (o_p, d_p, t_p), where $o_p \in S$ is the origin station, $d_p \in S$ the destination station, and $t_p \in H$ the desired departure time from the origin. The set of all passengers is denoted by P.

We use the passenger assignment model introduced in Chapter 4, where an exogenous ordering of the passengers is defined before assigning the passengers incrementally on the network. The algorithm takes the following information as an input: a train timetable, a passenger OD matrix, and a list defining assignment priorities among the passengers. Based on these inputs, it computes the passenger flows of the network. An indicator function $f^p : A \rightarrow \{0, 1\}$ shows if passenger $p \in P$ uses arc $a \in A$ in the output of the assignment.

Note that this assignment procedure relies on simulation to define the actual ordering of the passengers. Extensive numerical test have however shown that the variance across realizations of aggregate passenger satisfaction indicators (such as total generalized travel time) is very low. In the following, when assigning passenger demand, we thus assume that one run of the simulation is sufficient to obtain a good approximation of passenger satisfaction indicators. Further, we choose to apply priority specification R (see Section 4.4), in which passengers are assigned in a random order. For consistency, the passenger priority list is constructed in advance and thus passengers are always assigned in the same order (irrespective of the instance).

5.2.3 Mathematical formulation as a space-time graph

The timetable on which the passengers are assigned is mathematically represented by a directed space-time graph $G(V, A)$, as in Chapter 4. The reader can refer to the latter, especially Section 4.3.2, for a detailed description of the graph construction. The set of nodes V is equivalent. The set of arcs is given by $A = A_{Dri} \cup A_{Wai} \cup (\cup_{p \in P} A^p_{Acc}) \cup (\cup_{p \in P} A^p_{Egr}) \cup A_{Tra} \cup A_{Opt}$. There are six types of arcs in the graph, presented in Table 5.1. Driving and waiting arcs A_{Dri} and A_{Wai} model trains and passengers moving between stations and waiting within stations. Access and egress arcs A^p_{Acc} and A^p_{Egr} represent passenger p accessing and leaving the railway system. Potential passenger transfers in stations are modeled using the transfer arcs A_{Tra}. Finally, opt-out arcs A_{Opt} are introduced for passengers who cannot travel from origin to destination.

Table 5.1: Arc weights and capacities.

Name	Set	Start node	End node	c_a	c^p_a	q_a
Driving	A_{Dri}	(s, t, k)	(s', t', k)	$c \cdot d(s, s')$	$t' - t$	q_k
Waiting	A_{Wai}	(s, t, k)	(s, t', k)	0	$\beta_1 \cdot (t' - t)$	q_k
Access	A^p_{Acc}	o_p	(s, t, k)	-	$\beta_3 \cdot \max(0, (t_p - t)) +$ $\beta_4 \cdot \max(0, (t - t_p))$	1
Egress	A^p_{Egr}	(s, t, k)	d_p	-	0	1
Transfer	A_{Tra}	(s, t, k)	(s, t', k')	-	$\beta_2 + (t' - t)$	∞
Opt-out	A_{Opt}	o_p	d_p	-	$t(o_p, d_p) + C$	∞

With each arc a, a capacity q_a and two costs (c^p_a and c_a) are associated. The capacity of driving and waiting arcs is given by the capacity q_k of the train associated with the arc, as described in Section 4.3.2. Being passenger-specific, access and egress arcs have a capacity of one. Transfer and opt-out arcs have infinite capacity. The cost of using an arc a for a train (c_a), or for a passenger p (c^p_a), are listed in Table 5.1. We assume that the operational cost is proportional to the distance travelled by the trains. Therefore, only driving arcs have an operational cost c_a different from zero (c is the cost of running a train, per kilometer). An arc that is used by a passenger is weighted according to the generalized travel time defined in Eq. (4.1). Similarly to Kroon et al. (2015), it is assumed that the passengers have a "deadline" time: if the destination cannot be reached within this time, the passenger leaves the system. The deadline depends on the OD pair of the passenger. It is given by the generalized travel time on the shortest path between o_p and d_p, $t(o_p, d_p)$, plus a cutoff time C. The cost of a path in the graph for a passenger is obtained by summing the weights c^p_a of the arcs in the path.

The changes made to the timetable during the rescheduling process reflect directly in the graph. If a train is canceled, the space-time nodes associated with the train are removed from the graph. Additionally, all arcs incoming to and outgoing from the removed nodes are deleted. If a train is modified (i.e, delayed, advanced, rerouted or turned around) the space-time nodes associated with the train are removed from the graph, and a new set of nodes is added to the graph, as described in Section 4.3.2, to represent the timetable of the modified train. Access, egress and transfer arcs are updated according to the new train nodes. Finally, if an emergency train is added, new nodes and arcs are included in the graph, following the procedure described in Section 4.3.2.

5.2.4 Objective functions

Operational cost, passenger inconvenience and deviation from the undisrupted timetable are considered as three objectives to minimize and defined formally as follows, with respect to a timetable $G = G(V, A)$ and the undisrupted timetable $G^0 = (V^0, A^0)$. The *operational cost* (z_o) is the kilometric cost of all trains (original trains and emergency trains) operated in the timetable:

$$z_o(G) = \sum_{a \in A_{Dri}} c_a. \tag{5.1}$$

Passenger inconvenience (z_p) is defined as the total generalized travel time of all passengers and is given by

$$z_p(G) = \sum_{p \in P} \sum_{a \in A} c_a^p \cdot f^p(a). \tag{5.2}$$

The *deviation cost* (z_d) from the undisrupted timetable is computed by comparing the trains of the two timetables. Let K_G be the trains operated in timetable G, and K_{G^0} the trains in G^0. For convenience, we also define, for every train k, the ordered set of nodes $N_k \subset N$ (respectively, $N_k^0 \subset N^0$) visited in the timetable: $N_k(1)$ is the first node visited by train k in G, and $N_k(|N_k|)$ is the last one. Finally, let $t(n, n')$ be the time difference between space-time nodes n and n'. To compute the deviation cost from the undisrupted timetable, four weighting factors are used: penalties for canceling, delaying and rerouting trains, and the cost of operating emergency trains. We use the same values as in Veelenturf et al. (2016) (see also Chapter 3), where the aim is to operate as many original trains as possible. The canceled time of a train, i.e., the time difference between the original arrival time of the train at its last station and the time it is canceled, is weighted heavily by a factor of $\delta_c = 50$. Every delayed minute, for each train departure, is weighted only by a factor of $\delta_d = 1$. The time a rerouted train spends on a different geographical path than its train line is weighted by $\delta_r = 10$. These values ensure that rerouting a train is preferred over canceling it, while delaying is the least "costly" option. Finally, the penalty to operate an emergency train δ_e is 1000 plus the operated time of the emergency train.

The deviation cost is computed by sweeping over both sets of trains ($k \in K_G \cup K_{G^0}$) in the following manner.

- If $k \in K_{G^0}$ but $k \notin K_G$, the train k has been completely canceled. In this case, the deviation cost is proportional to the duration of the train journey that was canceled, given by $t(N_k^0(1), N_k^0(|N_k^0|))$, i.e., the time difference between the first and the last node of train k in G^0. Mathematically, $\Delta_c^k = \delta_c \cdot t(N_k^0(1), N_k^0(|N_k^0|))$.
- If $k \in K_G$ but $k \notin K_{G^0}$, the train k is scheduled as an emergency train. In this case, the deviation cost is the sum of the emergency train penalty and the duration of the train journey: $\Delta_e^k = \delta_e + t(N_k(1), N_k(|N_k|))$.
- If $k \in K_G$ and $k \in K_{G^0}$, the train k has been modified between G^0 and G, and we find the last node $n^* \subset N_k$ until which the train follows the sequence of stations of its train line, in order to compare the timetables. Let j be the index of this node in N_k, i.e., $N_k(j) = n^*$. Up to node n^*, the deviation cost is computed as a delay cost. For every departure node $N_k(i), 1 \leq i \leq j$, the time difference between the departure in G^0 and in G, $t(N_k^0(i), N_k(i))$, is computed. The delay cost is then proportional to this delay: $\Delta_d^k = \sum_{i \in \{1,\ldots,j\}} \delta_d \cdot t(N_k^0(i), N_k(i))$.

  After node n^*, there are three possibilities. If the train is partially canceled after n^*, the deviation cost is proportional to the canceled time between node n^* and $N_k^0(|N_k^0|)$: $\Delta_c^k = \delta_c \cdot t(n^*, N_k^0(|N_k^0|))$. If the train is rerouted after n^*, the deviation cost is proportional to the rerouted time between node n^* and $N_k(|N_k|)$: $\Delta_r^k = \delta_r \cdot t(n^*, N_k(|N_k|))$. Finally, if the train is turned around after n^*, we consider that an emergency train needs to be scheduled to compute the cost: $\Delta_e^k = \delta_e + t(n^*, N_k(|N_k|))$.

Given the definitions above, the overall deviation cost is the sum over all trains and all types of deviations:

$$z_d(G, G^0) = \sum_{k \in K_G \cup K_{G^0}} \Delta_c^k + \Delta_d^k + \Delta_r^k + \Delta_e^k. \tag{5.3}$$

For notational convenience, we denote the deviation cost $z_d(G, G^0)$ by $z_d(G)$ in the following.

5.3 Solution algorithm

In this section, a new solution algorithm for the problem defined above is introduced. It is based on Adaptive Large Neighborhood Search (ALNS), a meta-heuristic initially proposed by Ropke and Pisinger (2006) to solve the Pickup and Delivery Problem with Time Windows. It has also been used recently in railway timetable scheduling (see, e.g., Barrena et al., 2014; Robenek et al., 2017). ALNS is a type of large neighborhood search in which a number of fairly simple operators compete to transform the current solution.

At every iteration of the algorithm, the operators add, modify or remove trains from/to the timetable. We use simulated annealing as the search guiding meta-heuristic, as it seems to be the preferred approach in the ALNS literature.

Note that, according to its original definition, Adaptive Large Neighborhood Search relies on a set of destroy and repair operators that are chosen simultaneously at every iteration. By contrast, our heuristic selects a single operator from a set of addition, modification and removal operators. It can therefore also be seen as a hyper-heuristic, i.e., a "search method or learning mechanism for selecting or generating heuristics to solve computational search problems" (Burke et al., 2013).

Meta-heuristics commonly rely on a large set of parameters, whose values affect the outcome and the performance of the algorithm. In the following description, we provide a range or an initial value for the parameters that we introduce. The rationale behind the suggested values helps the reader to tailor the parameters to specific implementations. The values that are used in the case study are reported in Table 5.4.

5.3.1 General framework

The pseudo-code of the ALNS framework is shown in Algorithm 5. The sections referenced in the algorithm indicate where the different concepts are explained in further detail. The algorithm assumes the existence of an initial solution. In this case, a solution is defined as a timetable, i.e., the departure and arrival times of all trains at all stations. In the following, the notions of timetable and solution are therefore used interchangeably. The initial solution does not need to be feasible, hence the undisrupted timetable G^0 is used as an input. In the context of multi-objective optimization, it is common to define dominance relations among solutions in order to consider the trade-offs between the differents objectives (see Section 5.3.2 for formal definitions). The algorithm produces an archive of feasible non-dominated timetables, $\mathcal{A}$.

5.3.2 Multi-objective simulated annealing

Kirkpatrick et al. (1983) initially proposed the meta-heuristic known as simulated annealing (SA). Its characteristic feature is a probabilisitic criterion for the acceptance of worsening solutions in the search for a global optimum. In contrast with simple descent heuristics that only accept solutions which improve the value of the objective function, this allows to avoid that the procedure gets trapped in a local optimum. As SA is a well-known heuristic, we focus here on its extension to a multi-objective problem, based on the work of Suppapitnarm et al. (2000).

Algorithm 5: ALNS framework.

```
Input: Initial timetable $G^0$
       Initial (final) temperatures $T_i^{start}$ ($T_i^{end}$), $i \in \{z_p, z_o, z_d\}$
       Set of general operators $\Pi$ (Section 5.3.3)
       Set of infeasibility operators $\Pi_{Inf}$ (Section 5.3.3)
       $\rho_1 > \rho_2 > \rho_3 \geq 0$ (Section 5.3.3.4)
Output: Archive of feasible non-dominated timetables $\mathcal{A}$

$T_i \leftarrow T_i^{start}, i \in \{z_p, z_o, z_d\}$, $\mathcal{A} \leftarrow \varnothing, G, G' \leftarrow G^0$
while $T_i > T_i^{end}, i \in \{z_p, z_o, z_d\}$ do
    /* Timetable modification */
    if G is feasible (Section 5.2.1.2) then
        $\pi \leftarrow$ Select operator from $\Pi$ (Section 5.3.3.4)
        $G' \leftarrow$ Apply $\pi$ to $G$
    if $G'$ is not feasible (Section 5.2.1.2) then
        while $G'$ is not feasible do
            $\pi_{Inf} \leftarrow$ Select infeasibility operator from $\Pi_{Inf}$ (Section 5.3.3.4)
            $G'' \leftarrow$ Apply $\pi_{Inf}$ to $G'$
            Update score of $\pi_{Inf}$ (Section 5.3.3.4)
            $G' \leftarrow G''$
        Update weights of $\pi \in \Pi_{Inf}$ (Section 5.3.3.4)
    Assign passengers on $G'$ (Section 5.2.2)
    Evaluate $z_p(G'), z_o(G'), z_d(G')$ (Section 5.2.4)
    /* Archiving */
    if $G'$ can be archived (Section 5.3.2.1) then
        Add $G'$ to $\mathcal{A}$ (Section 5.3.2.1)
        $G \leftarrow G'$
        Update the score of $\pi$ by $\rho_1$ (Section 5.3.3.4)
    else
        /* Acceptance/Rejection */
        if $G'$ is accepted by the SA criterion (Eq. (5.4)) then
            $G \leftarrow G'$
            Update the score of $\pi$ by $\rho_2$ (Section 5.3.3.4)
        else
            Update the score of $\pi$ by $\rho_3$ (Section 5.3.3.4)
    Periodically, update $T_i$ according to the annealing schedule (Eq. (5.5))
    Periodically, update weights of general operators and reset scores (Section 5.3.3.4)
    Periodically, select $G$ randomly in $\mathcal{A}$ (Section 5.3.2.3)
return $\mathcal{A}$
```

5.3.2.1 Archiving and acceptance criterion

In order to address the multi-objective aspect of the problem, we define formally the concepts of dominance and Pareto optimality. Let $\mathcal{G}$ be the set of all feasible timetables, and G_1, G_2 two particular timetables. We say that G_1 dominates G_2 (equivalently, G_2 is dominated by G_1) if

$$z_i(G_1) \leq z_i(G_2), \forall i \in \{p, o, d\} \quad \text{and} \quad \exists i \in \{p, o, d\} : z_i(G_1) < z_i(G_2),$$

where z_p, z_o, z_d are defined in Eqs. (5.1)–(5.3). In other words, G_1 is not worse than G_2 in any objective and G_1 is strictly better than G_2 in at least one objective. We denote this property by $G_1 \prec G_2$ (the notation is oriented in this direction because the objectives are *minimized*). A timetable $G^* \in \mathcal{G}$ is said to be Pareto optimal if it is not dominated by any other timetable: $\nexists G \in \mathcal{G} : G \prec G^*$. The set of all Pareto optimal timetables is $\mathcal{P}^* = \{G^* \in \mathcal{G} | \nexists G \in \mathcal{G} : G \prec G^*\}$.

The derivation of the entire set $\mathcal{P}^*$ is beyond the scope of this research. Instead, we maintain an *archive* $\mathcal{A}$ of non-dominated solutions in Algorithm 5, as an approximation of $\mathcal{P}^*$. We begin with an empty archive (line 1). Then, at every iteration, the current timetable G' is a candidate for archiving (line 17). The following cases need to be considered:

a) G' is dominated by at least one solution in $\mathcal{A}$ (i.e., $\exists G \in \mathcal{A} : G \prec G'$);

b) G' is not dominated by any solution in $\mathcal{A}$ (i.e., $\nexists G \in \mathcal{A} : G \prec G'$).

In the first case, G' is not added to the archive and $\mathcal{A}$ is not modified. In the second case, G' is added to $\mathcal{A}$, and all solutions in $\mathcal{A}$ that are dominated by G' are removed from the archive (lines 17–20).

Given this archiving procedure, we define the following acceptance criterion for the current timetable G'. If G' is archived, it is automatically accepted (line 19). If not, it is accepted with probability

$$\prod_{i \in \{p,o,d\}} \min \left\{ \exp \left(-\frac{z_i(G') - z_i(G)}{T_i} \right), 1 \right\}. \tag{5.4}$$

This acceptance probability is the product of individual acceptance probabilities for each objective, with its associated temperature T_i. It is therefore not necessary to scale the objectives with respect to each other to form a composite objective function. Note that the individual acceptance probability, $\exp(-(z_i(G') - z_i(G))/T_i)$, may be greater than unity (if the difference is negative). We therefore take the minimum between the calculated value and one.

5.3.2.2 Temperature update

In SA, the "temperature" of the system is a control parameter that indicates how likely it is that a worsening solution is accepted in the search process. For high values of T_i in Eq. (5.4), virtually all new solutions are accepted, irrespective of the sign of the numerator. Conversely, only small increases in the objective function z_i are accepted for small values of T_i. Thus, the search is initiated with high values for the temperature, in order to allow for a broad exploration of the search space, and to escape local minima. The temperature is then gradually decreased as the search intensifies around a (hopefully global) minimum.

We use the cooling schedule proposed in Bierlaire (2015) to update the temperatures (line 28). The temperatures $T_i^{start} > 0$ are initially set to a sufficiently high value, which guarantees that virtually all solutions are accepted in this phase. The cooling of the temperatures needs to be related to typical values of the objective functions in order to obtain meaningful acceptance probabilities. To that end, Suppapitnarm et al. (2000) suggests to compute, after a predetermined number of iterations N_1, the standard deviation of the respective objective functions, $\sigma(z_i)$, over all previously accepted solutions. The authors suggest $N_1 = 1000$ iterations, but this value can be reduced if the standard deviations stabilize earlier. After this warm-up phase, the temperatures are lowered M times, according to the following schedule:

$$T_i(m) = -\frac{\sigma(z_i)}{\ln(p_0 + \frac{p_f - p_0}{M} m)}, \qquad m = 0, \ldots, M. \tag{5.5}$$

This cooling strategy has the advantage that the acceptance probability decreases linearly, with respect to m, between the initial (p_0) and the final (p_f) acceptance probability. The algorithm is run K times at this temperature, before the next cooling step. Bierlaire (2015) reports results of the application of simulated annealing to the knapsack problem with $M \cdot K = 5000$, $p_0 = 0.999$ and $p_f = 0.00001$.

5.3.2.3 Return-to-archive strategy

In order to explore more intensively the search space close to the Pareto frontier, Suppapitnarm et al. (2000) suggests to implement an occasional selection of a solution in the archive, from which to recommence the search. The rate at which to come back to a solution from the archive is decided in the following way. In the warm-up phase before the iteration count hits N_1, we do not perform any return-to-archive moves, as the goal is to explore the search space as freely as possible at that time. Thereafter, the rate of return to solutions of the archive is increased in the following manner: $N_j^B = r_B \cdot N_{j-1}^B, j = 2, 3, 4, \ldots,$ where N_j^B is the number of iterations to be executed prior to the jth return-to-archive and $0 < r_B < 1$ is a parameter controlling the rate of return. Suitable values of the parameters found by Suppapitnarm et al. (2000) are $r_B = 0.9$ and $N_1^B = 2K$.

The choice of the solution in the archive is performed in a random manner (line 30). Note that this might not be a particularly efficient approach, as the search is more likely to return to well-explored parts of the search space, where there are already many archived solutions. Alternative selection processes may easily be implemented instead: choosing a solution based on its degree of isolation (Suppapitnarm et al., 2000), or using "biased fitness" or solution diversity (Vidal et al., 2012) to guide the selection process.

5.3.3 Operators

Operators are used in the ALNS framework in order to modify the current solution. An operator is a combination of two building blocks: a *neighborhood structure* that defines how to modify a timetable to generate a neighbor timetable, and a *search strategy* that modifies the current timetable by selecting one of the neighbors. In the following, we first introduce ten neighborhood structures, and then the search strategies that are associated with each of them.

5.3.3.1 Neighborhood structures

The neighborhood structures introduced in this section define the potential modifications of a timetable in order to obtain a neighbor timetable. Observe that these are directly inspired from rescheduling operations applied in practice, as illustrated in Chapter 2. Table 5.2 summarizes the effects of the different neighborhood structures on the current timetable.

Cancellation neighborhood structures (CC and PC) select a train $k \in K$ and cancel it, either completely or partially. In the case of a partial cancellation, the train is canceled after a station $s \in S_k$, where $S_k \subset S$ is the ordered set of stations visited by train k.

Delay neighborhood structures (CD and PD) modify a train $k \in K$ by delaying it, either completely or partially. The time by which the train is delayed is a multiple of 5 minutes, and cannot exceed 30 minutes. We introduce the set of potential delay times, $H_D = \{5, 10, 15, 20, 25, 30\}$. In the case of a complete delay, train k is delayed by $t \in H_D$ minutes on its entire line, whereas for a partial delay after station $s \in S_k$, train k is operated on time up to station s, and delayed by $t \in H_D$ minutes thereafter.

Similarly, advance neighborhood structures (CA and PA) modify a train $k \in K$ by scheduling it earlier, either completely or partially. The following additional conditions need to be respected: (i) if k is an original train, no departure time can be scheduled

Table 5.2: List of neighborhood structures.

Name	Label	Parameters	Description
Complete Cancel	CC	k	Select train $k \in K$ and cancel it completely
Partial Cancel	PC	k, s	Select train $k \in K$ and cancel it after its arrival in station $s \in S_k$
Complete Delay	CD	k, t	Select train $k \in K$ and delay it by $t \in H_D$ minutes on its complete line
Partial Delay	PD	k, t, s	Select train $k \in K$ and delay it by $t \in H_D$ minutes after station $s \in S_k$
Complete Advance	CA	k, t	Select train $k \in K$ and advance it by $t \in H_A^k$ minutes on its complete line
Partial Advance	PA	k, t, s	Select train $k \in K$ and avance it by $t \in H_A^k$ minutes after station $s \in S_k$
Insert Train	IT	ℓ, t	Insert an emergency train along train line $\ell \in L$, starting its trip at time $t \in H_I^\ell$
Reroute	RR	k, s, s'	Select train $k \in K$ and reroute it between neighboring stations $s, s' \in S_k$
Turn Around	TA	k, s	Select train $k \in K$ and turn it around at station $s \in S_k$
Insert Bus	IB	s, s', t	Insert a bus between neighboring stations $s, s' \in S$, starting at time $t \in H_B$

earlier than in the undisrupted timetable; (ii) a train cannot depart from a station earlier than its departure time from the previous station, plus the running time between the stations and the minimal dwell time at the station. The set of potential advance times, $H_A^k \subseteq H_D$, includes these restrictions. In the case of a complete advance, train k is advanced by $t \in H_A^k$ minutes on its entire line, whereas for a partial advance after station $s \in S_k$, train k is not modified up to station s, and advanced by $t \in H_A^k$ minutes thereafter.

The neighborhood structure inserting an emergency train (IT) schedules a new train along a train line $\ell \in L$. An additional train can be scheduled every 15 minutes from the first station of the train line. The set of potential departure times from the first station is given by H_I^ℓ.

The rerouting neighborhood structure (RR) modifies a train $k \in K$ by rerouting it between neighboring stations $s, s' \in S_k$. The set of all possible reroutings between stations s and s' is generated in advance, and the rerouting with the lowest additional travel time is selected. Let $\{s_j\}_{j=1}^N, N \geq 1$, be the sequence of stations visited in this rerouting between stations s and s'. Train k is then modified in the following way: up to station s, there is no modification to its schedule; between s and s', the train is rerouted along stations $\{s_j\}_{j=1}^N$; after s', the train runs on its original path but is delayed because of the rerouting. The departure times at stations after s are updated according to running and minimal dwell times.

The turn-around neighborhood structure (TA) modifies a train $k \in K$ by turning it around after station $s \in S_k$. Train k is modified in the following way: up to station s, there is no modification to its schedule; then a turn-around time of 10 minutes is considered, after which it departs in opposite direction, back to the station where it started its journey. The departure times at stations after s are updated according to running and minimal dwell times.

The neighborhood structure inserting an emergency bus (IB) schedules a new bus between two neighboring stations $s, s' \in S$. A bus can be scheduled to depart every 5 minutes. The set of potential departure times is given by H_B. The bus travel time and cost are doubled compared to operating a train between the stations.

5.3.3.2 Search strategies

The choice of parameters (k, s, s', t, ℓ) of the neighborhood structures described above determines which neighbor timetable G' is constructed from timetable G. For a given neighborhood structure, the list of potential combinations of parameters is given by Table 5.2. For instance, for neighborhood structure PD, the set of neighbor timetables of G, $N(G)$, is given by all possible combinations of $k \in K$, $t \in H_D$ and $s \in S_k$. Four search strategies are used in order to choose the combination of parameters from this list: local search, random selection, disruption mitigation and feasibility restoration.

Local search A local search algorithm is run to select a combination of parameters that minimizes locally one of the objectives (z_p, z_o or z_d). This search strategy is applied in association with the following neighborhood structures: CC, PC, CD, PD, CA, PA or IT. Ideally, the set of all possible neighborhood timetables $N(G)$ should be constructed, by applying one of the above neighborhood structures with all possible parameter combinations. However, the construction of the complete set $N(G)$ becomes computationally too demanding and we therefore impose an upper bound $\nu_{\max}$ on the number of evaluated neighbors. For a given neighborhood structure, the local search evaluates successively the feasible parameter combinations introduced in Table 5.2, and adds the neighbor timetable to $N(G)$. Once $|N(G)| = \nu_{\max}$, the neighbor timetable $G' \in N(G)$ with the lowest value of the corresponding objective function is selected.

Note that applying delay and advance neighborhood structures does not affect the operational cost of the timetable (see Eq. (5.1)). The local search minimizing z_o is therefore not defined for CD, PD, CA and PA.

Random selection This search strategy is applied in association with all neighborhood structures, except RR and TA. It is mostly used for diversification purposes at the beginning of the algorithm. For a given neighborhood structure, the combination of parameters is chosen by drawing each parameter uniformly from the sets described in Table 5.2.

Disruption mitigation This search strategy is applied exclusively in association with the bus insertion neighborhood structure. Its purpose is to schedule a bus that "bridges the gap" between the two sides of the disruption. For a disruption on track (s_1, s_2), the neighborhood structure IB is therefore used with $s = s_1$ and $s' = s_2$. Two operators are defined: the departure time t is either drawn uniformly from H_B, or a local search is applied to determine the value $t \in H_B$ that generates the neighbor timetable with the lowest passenger inconvenience. Again, an upper bound $\nu_{\max}$ is imposed on the number of evaluated timetables in the local search.

Feasibility restoration This search strategy is applied in association with all neighborhood structures, except IT and IB. Starting from an infeasible timetable G' (according to the definition of Section 5.2.1.2), its purpose is to generate a feasible neighbor timetable G''. The search strategy uses the first infeasibility of the timetable to determine the appropriate combination of parameters for a given neighborhood structure. If the infeasibility is headway-related, let $k_1, k_2 \in K$ be the two involved trains, and $(s_1, s_2) \in Q$ the track where the minimal headway is not respected. In this case, the parameters of the neighborhood structure are set (when applicable) to $k = k_1$, $s = s_1$, $s' = s_2$ and $t = h$ (the minimal headway time). If the infeasibility is disruption-related, let $k_1 \in K$ be the train that is scheduled on the disrupted track $(s_1, s_2) \in Q$. In that case, the parameters of the neighborhood structure are set (when applicable) to $k = k_1$, $s = s_1$ and $s' = s_2$. The delay (respectively, advance) time t is selected (in H_D or $H_A^{k_1}$) so as to train k_1 passes track (s_1, s_2) just before the start or after the end of the disruption. If no such time exists, t is set to the maximal delay or advance time.

Note that one application of a feasibility restoration operator does not necessarily result in a feasible neighbor timetable. A loop is therefore introduced in Algorithm 5 (lines 8–12), in order to ensure that the final neighbor timetable G'' is feasible before its potential inclusion in the archive.

5.3.3.3 Summary and operator classification

Based on these definitions, Table 5.3 summarizes the operators that are used in the ALNS framework. Each row corresponds to a neighborhood structure, and each column indicates the search strategy that determines the parameters of the neighborhood structure. A total of 35 operators is thus defined.

We classify the operators into two sets: *general operators*, Π, and *infeasibility operators*, Π_{Inf}. The former try to improve the current solution, whereas the latter attempt to make the current solution feasible. The set of general operators com-

Table 5.3: List of operators of the ALNS framework.

	Search strategy					
	Local search			Random selection	Disruption mitigation	Feasibility restoration
	z_p	z_o	z_d			
CC	CC_P	CC_O	CC_D	CC_R	-	CC_{Inf}
PC	PC_P	PC_O	PC_D	PC_R	-	PC_{Inf}
CD	CD_P	-	CD_D	CD_R	-	CD_{Inf}
PD	PD_P	-	PD_D	PD_R	-	PD_{Inf}
CA	CA_P	-	CA_D	CA_R	-	CA_{Inf}
PA	PA_P	-	PA_D	PA_R	-	PA_{Inf}
IT	IT_P	IT_O	IT_D	IT_R	-	-
RR	-	-	-	-	-	RR_{Inf}
TA	-	-	-	-	-	TA_{Inf}
IB	-	-	-	IB_R	$IB_{Dis,P}, IB_{Dis,R}$	-

prises cancel operators ($\Pi_C = \{CC_P, CC_O, CC_D, CC_R, PC_P, PC_O, PC_D, PC_R\}$), delay operators ($\Pi_D = \{CD_P, CD_D, CD_R, PD_P, PD_D, PD_R\}$), advance operators ($\Pi_A = \{CA_P, CA_D, CA_R, PA_P, PA_D, PA_R\}$), train insertion operators ($\Pi_I = \{IT_P, IT_O, IT_D, IT_R\}$) and bus insertion operators ($\Pi_B = \{IB_R, IB_{Dis,P}, IB_{Dis,R}\}$). The set of infeasibility operators is given by $\Pi_{Inf} = \{CC_{Inf}, PC_{Inf}, CD_{Inf}, PD_{Inf}, CA_{Inf}, PA_{Inf}, RR_{Inf}, TA_{Inf}\}$.

5.3.3.4 Operator weight and score update

For general operators, a two-level roulette-wheel mechanism governs the choice of the operator (line 5). First, the type of neighborhood structure Π' is chosen uniformly in $\{\Pi_C, \Pi_D, \Pi_A, \Pi_I, \Pi_B\}$, thus ensuring an equitable use of cancellation, insertion, delay, advance and bus insertion operators. Second, the operator is selected in the following manner. Given the set of operators Π', operator $\pi \in \Pi'$ is chosen with probability $w_\pi / \sum_{\pi' \in \Pi'} w_{\pi'}$, where w_π is the non-negative weight associated with operator π. The operator weights are updated using information from earlier iterations: the idea is to keep track of a score s_π for every operator in order to compute its weight. This score measures the recent performance of the given operator; a higher score corresponds to a better performance. The ALNS algorithm is divided into segments of J iterations. The score of all operators is set to zero at the beginning of every segment. Then, at every iteration, the score of the selected operator is updated. We distinguish three cases: if the last iteration of the algorithm

- resulted in a new solution added to the archive (line 20), the operator is rewarded with a score increase $\rho_1 > 0$;

- resulted in a solution that was not added to the archive (line 25), but was accepted by the SA criterion, the operator score is increased by $\rho_2 < \rho_1$;
- resulted in a solution that was neither added to the archive, nor accepted by the SA criterion (line 27), the score of the selected operator is not increased ($\rho_3 = 0$).

The values of ρ_1, ρ_2 and ρ_3 need to be tuned together, in order to model the relative importance of the rewards. For instance, the parameter tuning of Ropke and Pisinger (2006) resulted in $\rho_1 = 33$, $\rho_2 = 9, \rho_3 = 13$.

At the beginning of the algorithm, all weights are set to one. Then, at the end of every segment (line 29), the weights are updated using the recorded scores in the following manner. After segment j, the weight of operator π in segment $j+1$, w_π^{j+1}, is given by

$$w_\pi^{j+1} = (1-\eta)w_\pi^j + \eta\frac{s_\pi}{n_\pi},$$

where $0 < \eta < 1$ is the reaction factor that controls how quickly the weight adjustment responds to changes in the effectiveness of the operators and n_π is the number of times operator π was used in the previous segment (if $n_\pi = 0$, we assume that the weight of operator π remains unchanged in segment $j+1$). A lower bound $w_{\min}$ is enforced on the weights. Its purpose is to avoid that the probability of choosing an operator becomes too small. Its value needs therefore to be adjusted a posteriori, once the maximal weights of all operators are known.

For infeasibility operators, a single-level roulette-wheel mechanism is implemented. In that case, an infeasibility operator's score is increased by ρ_1 if it produces a feasible solution, and by ρ_2 if it reduces the number of infeasibilities compared to the previous solution (line 11). Note that we choose to use the same score increases ρ_1 and ρ_2 for general and infeasibility operators, but this is not required. For infeasibility operators, the concept of segments of iterations with a constant weight is not relevant. Their weights are therefore updated once all infeasibilities of a solution are resolved (line 13), and the scores reset to zero.

5.4 Computational experiments

Computational results are presented in the following manner. In Section 5.4.1, the test instances on which the algorithm is applied are introduced. Section 5.4.2 then describes the parameter settings. The contribution of the different operators of the algorithm are discussed in Section 5.4.3, and the computational results are presented in Section 5.4.4. Finally, the practical applicability of the framework is discussed in Section 5.4.5.

The algorithm is implemented in Java and all numerical experiments have been carried out on a 3.33 GHz Intel Xeon X5680 server running 64-bit Ubuntu 16.04.2.

5.4.1 Test instances

Two types of instances are considered in order to test the algorithm and its operators.

First, a large set of "small" instances is created from the Dutch case study introduced in Section 2.3.1. Its main purpose is to calibrate the algorithm's parameters, and to evaluate the contribution of its operators in finding non-dominated solutions. The Dutch case study comprises 18 double tracks between stations. For each of these, the following disruptions are considered:

- complete blockade for the entire time horizon (0–120 minutes);
- complete blockade for the first half of the time horizon (0–60 minutes);
- partial blockade for the entire time horizon (0–120 minutes).

A total of 54 test instances is thus created. For each of these, the algorithm is run once and results are reported in Section 5.4.3.

Second, two disruption scenarios are constructed for the realistic Swiss case study introduced in Section 2.3.2. The first scenario is a full blockade of the tracks between stations MON and VEV (see Fig. 2.4), between 05:30am and 06:30am. The main purpose of this scenario is to analyze the disposition timetables constructed by our framework, and to validate the latter by comparing the output of the algorithm with recovery procedures that might be implemented in practice. Indeed, this scenario is easily solved "by hand": there are only three trains using the disrupted tracks in the normal timetable, and the fact that there are no other railway tracks linking MON to VEV limits the number of rescheduling possibilities. The second scenario illustrates a more severe case, with the two following disruptions happening simultaneously:

- complete blockade of the tracks between LAU and VEV, from 5:00am to 8:00am;
- complete blockade of the tracks between LAU and PUI, from 5:00am to 7:00am.

For the two scenarios, the algorithm is run ten times, and the archives of non-dominated solutions of every run are merged together, according to the dominance rules described in Section 5.3.2.1. By applying this procedure, all dominated solutions are removed. The results of the merged archives are reported in Sections 5.4.4.1 and 5.4.4.2.

5.4.2 Parameter settings

Dayarian et al. (2016) note that, similarly to most meta-heuristics, the value of the parameters in an ALNS framework may affect to a high extent the performance of the algorithm, but not its correctness. An extensive tuning of the parameters, using for instance an optimizer such as Opal (Audet et al., 2014), is out of the scope of our research. Parameters are however tuned by performing numerous trial runs of the algorithm on the 54 test instances described above. Table 5.4 summarizes the values we use for the parameters of the algorithm, first SA-related parameters, followed by ALNS-related parameters. Initial values are either borrowed from the literature or based on preliminary trial-and-error combinations.

Table 5.4: Algorithmic parameters.

Symbol	Description	Value	Source
T_i^{start}	Initial temperature	10^8	Trial-and-error
p_0	Initial acceptance probability	0.999	Bierlaire (2015)
p_f	Final acceptance probability	0.001	Bierlaire (2015)
M	Number of temperature changes	100	Bierlaire (2015)
K	Iterations per temperature level	50	Bierlaire (2015)
N_1	Warm-up iterations	300	Trial-and-error
N_1^B	Iterations before first return-to-archive	$2K$	Suppapitnarm et al. (2000)
r_B	Decrease parameter	0.99	Suppapitnarm et al. (2000)
J	Weight segment length	$2K$	Trial-and-error
η	Reaction factor	0.5	Ropke and Pisinger (2006)
ρ_1	Reward	10	Trial-and-error
ρ_2	Reward	2	Trial-and-error
ρ_3	Reward	0	Trial-and-error
$w_{\min}$	Minimal operator weight	0.3	Trial-and-error
$\nu_{\max}$	Maximal number of evaluated solutions	20	Trial-and-error

The initial temperatures (T_i^{start}) are set sufficiently high to ensure that all generated solutions during the warm-up phase of the algorithm (before N_1) are accepted. Initial and final acceptance probabilities (p_0 and p_f), as well as the number of temperature changes M and the number of iterations per temperature level K, obviously have a direct impact on the running time of the algorithm, as they determine its total number of iterations. We use the values suggested by Bierlaire (2015), but if more (or less) iterations are required, these can be adjusted easily. The number of “warm-up iterations” N_1 is tuned so as to obtain a meaningful value for the standard deviation of the three objectives before starting the cooling schedule. Given these parameters, one run of the algorithm comprises 5300 iterations. Finally, parameters related to the return-to-archive procedure (N_1^B and r_B) are borrowed from Suppapitnarm et al. (2000), but adjusted because they generated too frequent returns.

The weights of the ALNS operators are updated every two temperature updates: $J =$

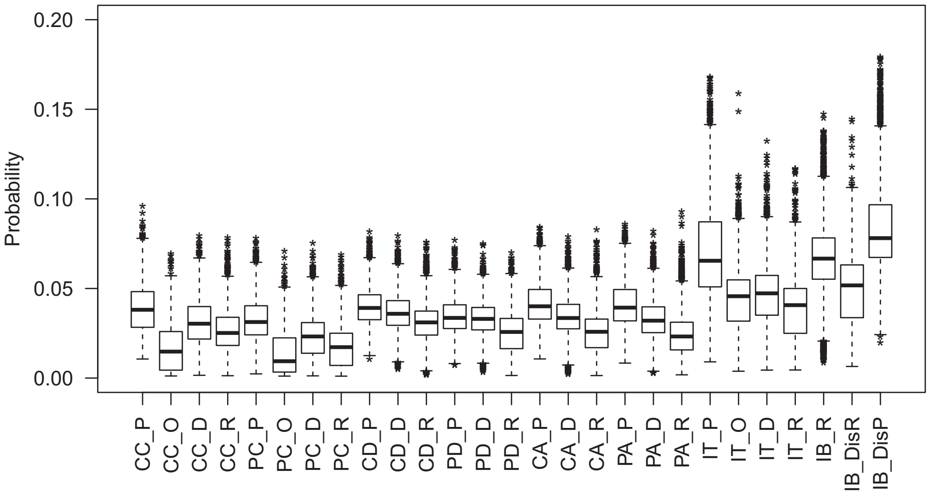
$2K$. Ropke and Pisinger (2006) proposed a very low value of the reaction factor η; we use a higher value in order to increase the impact of newly gained information on the operator weights. The rewards ρ_1, ρ_2, ρ_3 were tuned together, and we decided to boost the importance of finding a new non-dominated solution by giving it a (relatively) higher reward than finding an accepted solution. The minimal operator weight $w_{\min}$ was set so as to ensure a non-infinitesimal probability of choosing each operator, while avoiding that the weight mechanism becomes completely irrelevant. Finally, the maximal number of evaluated solutions $\nu_{\max}$ defines the trade-off between computational time and optimality of the local search operators.

5.4.3 Operator usage statistics

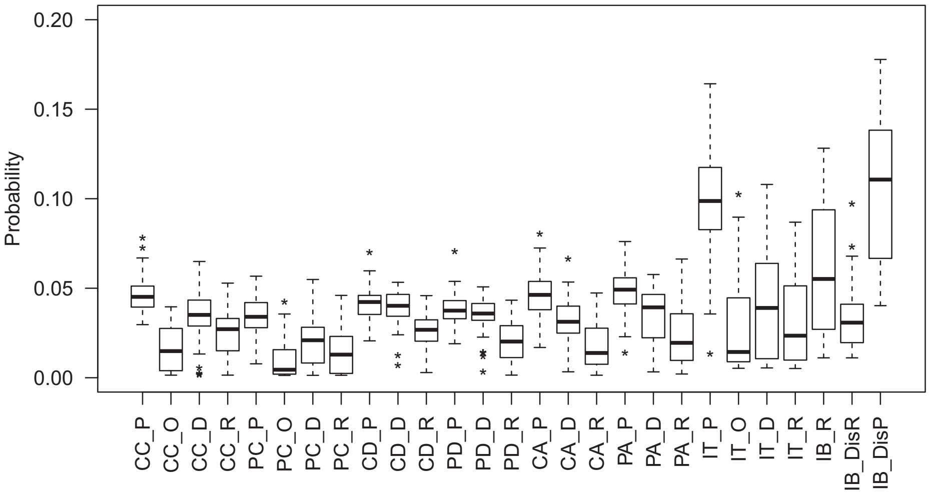
This section discusses the usefulness of all the operators of the algorithm, by applying the latter on the 54 test instances introduced above. First, the probabilities of choosing each operator are examined in Fig. 5.1, both during and at the end of the algorithm. We then examine the contribution of the operators to the search for non-dominated solutions in Table 5.5, and the capacity of infeasibility operators to restore timetable feasibility in Table 5.6. Finally, computational times of the operators are reported in Fig. 5.2.

Recall that the algorithm is divided into segments of length J, at the end of which the weights (from which the probability to be selected are computed) of all general operators are updated simultaneously. Fig. 5.1 reports the distribution of probabilities that a general operator is chosen: Fig. 5.1a shows the distribution of probabilities at the end of all segments in the 54 instances, whereas Fig. 5.1b limits the analysis to the final segment of each of the instances. The boxplots are defined identically to Section 4.6.2. The probabilities reported in Fig. 5.1a indicate that each operator is useful at some point, when considering results across all instances. Indeed, the median probability to select an operator is significantly different from zero across all segments, and every operator exhibits high values of its maximal probability of being chosen. At the same time, the high variability of the probabilities to select an operator demonstrates the usefulness of the weight adaptation scheme of the ALNS framework. Even for operators such as CC_O, PC_O, PC_R, which exhibit very low final probabilities, there are instances where the probability is significantly higher. An operator can therefore be very efficient in the case of one instance, but not contribute significantly to another, thus suggesting that the weight updating scheme is capable to adapt to the specificities of each instance. Note however that a danger of this procedure is to exclude operators that are useless when far from the solution, but useful close to the optimal solution. If they perform poorly in the first iterations, they will be seldom used, and the algorithm will not have the opportunity to change its mind about their performance.

Moreover, it is interesting to note that final probabilities (Fig. 5.1b) accentuate the



(a) Distribution of operator choice probabilities at the end of all segments.



(b) Distribution of operator choice probabilities at the end of the final segment.

Figure 5.1: Distribution of probabilities of choosing each operator, over the 54 test instances.

overall trend observed during the algorithm (Fig. 5.1a). It can thus be observed which operators become essential, and which are not that useful anymore, in the final intensifying iterations of the algorithm. Operators minimizing passenger inconvenience appear to have consistently higher probabilities of being selected for the groups of can-

Table 5.5: Usage statistics and success rates of general operators.

General operator	# of non-dominated solutions found	# of usages	Success rate
CC_P	4572	10,672	0.43
CC_O	1229	4863	0.25
CC_D	3119	8581	0.36
CC_R	2113	7373	0.29
PC_P	3016	9105	0.33
PC_O	718	3941	0.18
PC_D	1448	6357	0.23
PC_R	923	5116	0.18
CD_P	4077	11,108	0.37
CD_D	3411	10,393	0.33
CD_R	2121	8750	0.24
PD_P	2715	9517	0.29
PD_D	2626	9325	0.28
PD_R	1360	7006	0.19
CA_P	3508	11,424	0.31
CA_D	1944	9644	0.20
CA_R	446	3382	0.13
PA_P	2876	9548	0.30
PA_D	1500	7743	0.19
PA_R	210	1228	0.17
IT_P	4299	19,716	0.22
IT_O	931	12,542	0.07
IT_D	1118	13,310	0.08
IT_R	661	10,894	0.06
IB_R	317	18,504	0.02
$IB_{Dis,R}$	179	13,892	0.01
$IB_{Dis,P}$	1746	23,476	0.07

cel, advance, and train and bus insertion operators (CC_P, PC_P, CA_P, PA_P, IT_P and $IB_{Dis,P}$). Regarding delay operators, there appears to be no clear preference between minimizing passenger inconvenience or the deviation from the undisrupted timetable (see CD_P, CD_D and PD_P, PD_D). It is important to remember however that these operators should not be overused either: they induce a greedy optimization along one objective and require higher computational times than other operators, as will be shown in Figs. 5.2 and 5.4.

Table 5.5 reports over the 54 test instances, for each general operator, the number of times it found a non-dominated solution, the number of times it was used, and its success rate, defined as the ratio between the two previous values. Note that the number of non-dominated solutions is an overestimation of the total number of solutions in the final archive, as a non-dominated solution in an early iteration might become dominated by a later solution. The fact that each operator is able to find a significant amount of non-dominated solutions corroborates the usefulness of the different operators described above. Further, it can be observed that operators with low success rates (such as IT_O, IT_D, IT_R, IB_R and $IB_{Dis,R}$) are not necessarily the ones with the lowest number of usages, indicating that those should not be discarded as useless, but rather considered as diversification opportunities that pave the path for a later significant improvement leading to a new non-dominated solution. Likewise, operators with a very high number of usages are not the ones with the best success rate.

Table 5.6: Usage statistics of infeasibility operators.

Infeasibility operator	# of resolved infeasibilities	# of reduced infeasibilities	# of usages
CC_{Inf}	19,587	725	20,312
PC_{Inf}	3435	1021	4456
CD_{Inf}	2250	232	5223
PD_{Inf}	660	333	1664
CA_{Inf}	0	0	236
PA_{Inf}	0	1	13
RR_{Inf}	3821	149	4798
TA_{Inf}	2597	830	3795

Table 5.6 reports the usage statistics of infeasibility operators, as well as their capacity to restore feasibility. The number of resolved infeasibilities indicates how many times an operator was able to generate a feasible neighbor timetable, whereas the number of reduced infeasibilities reports the number of times the operator generated a neighbor timetable with a lower number of infeasibilities than the incumbent solution. One can observe that the complete cancellation operator is by far the most used, which is explained by the fact that resolving an infeasibility is highly rewarded in the weighting scheme of the algorithm. The other infeasibility operators are nevertheless also used frequently, except for the advance operators (CA_{Inf} and PA_{Inf}). The latter results from the nature of the considered disruptions: all disruptions start at the beginning of the time horizon, and it is therefore impossible to schedule a train to pass before the disruption happens. This demonstrates once more the usefulness of the weight adaptation scheme.

Finally, Fig. 5.2 reports over the 54 test instances, for every operator, the distribution of average computational times during a segment. Local search operators obviously exhibit longer computational times than the operators with a different search strategy. This becomes even more evident for the larger case studies discussed below. Overall, computational times are very low, at less than one tenth of a second per operator usage.

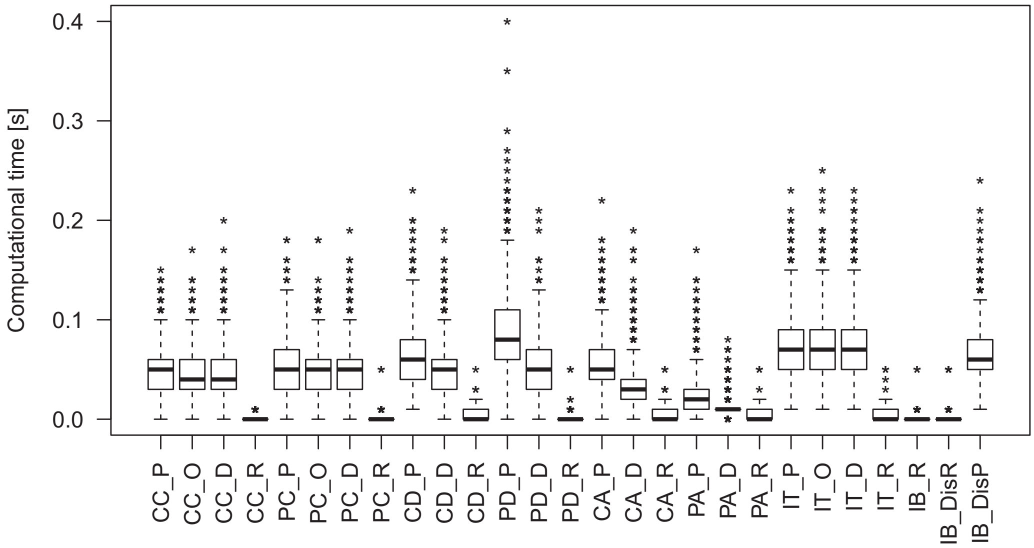


Figure 5.2: Distribution of computational times of the operators over the 54 test instances.

5.4.4 Computational results

5.4.4.1 Disruption MON–VEV

As introduced in Section 5.4.1, this section focuses on the analysis of the disposition timetables constructed by our framework for a disruption scenario that is easy to solve “by hand”, due to the limited rescheduling possibilities. Indeed, a full blockage of the tracks between stations MON and VEV from 5:30am to 6:30am is considered. In the normal timetable, three trains are scheduled on the disrupted tracks:

- Train **S2 12211**, leaving VEV at 06:21am and reaching MON at 06:30am;
- Train **S2 12216**, leaving MON at 06:29am and reaching VEV at 06:37am;
- Train **S3 12314**, leaving MON at 05:55am and reaching VEV at 06:03am.

For this disruption scenario, the merged archive of non-dominated solutions obtained by our framework contains 952 disposition timetables. Table 5.7 summarizes the recovery strategies applied to the three disrupted trains in each of the timetables: column CC indicates the number of timetables in which the train is completely canceled; column PC the number of timetables in which the train is partially canceled, and after which

Table 5.7: Summary of recovery strategies applied to trains S2 12211, S2 12216 and S3 12314 in disruption MON–VEV.

Train	Recovery strategy usage				
	CC	PC	CD	PD	TA
S2 12211	440	14 (COS) 7 (LAU) 15 (MON) 5 (REN) 278 (VEV)	38	23 (LAU) 92 (VEV)	1 (COS) 39 (VEV)
S2 12216	337	363 (MON)	227	12 (MON)	13 (MON)
S3 12314	343	481 (MON)	1	11 (MON)	116 (MON)

station; column CD the number of timetables in which the train is completely delayed; column PD the number of timetables in which the train is partially delayed, and after which station; and column TA the number of timetables in which the train is turned around, and after which station.

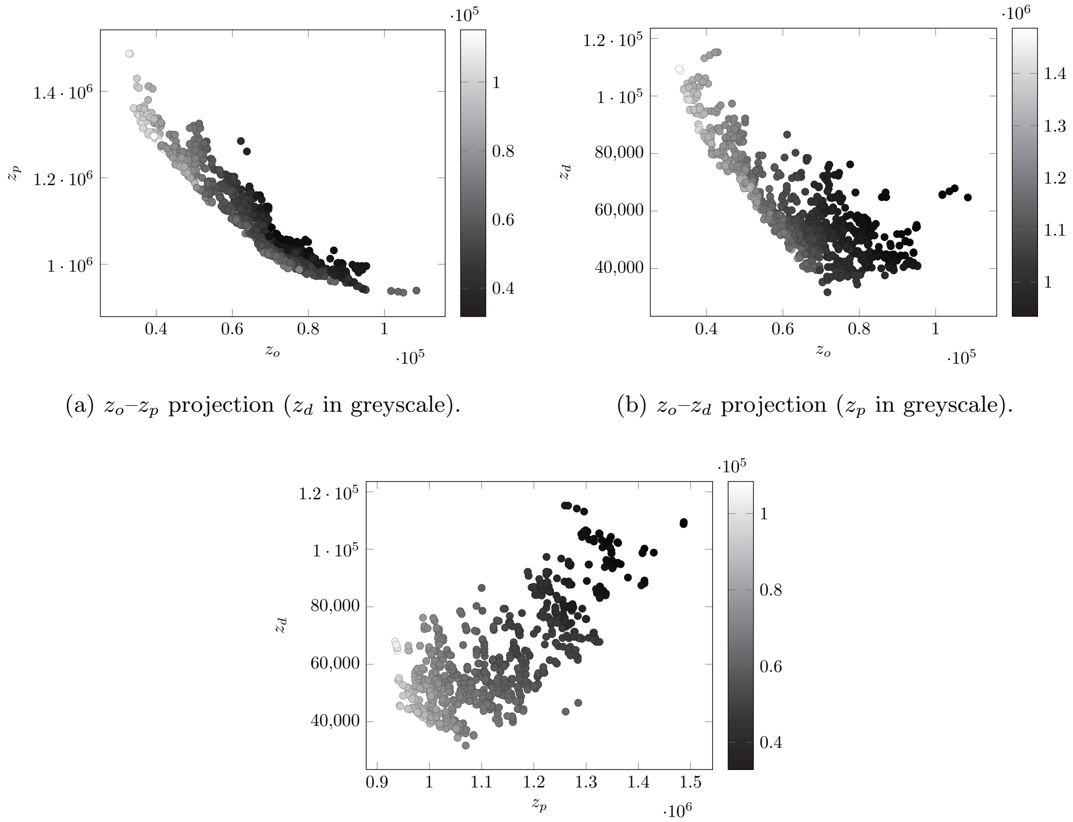
It can be observed that "common sense" recovery actions are properly reproduced by our algorithm. For instance, trains S1 12211 and S2 12216, which are scheduled in the normal timetable to cross the disrupted tracks only very shortly before the disruption ending time (9 and 1 minutes, respectively), are much more likely to be delayed (partially or completely) than train S3 12314. Indeed, by delaying these trains only by a few minutes, the timetable easily becomes feasible again. In particular, it is noteworthy that the algorithm proposes timetables in which these two trains are delayed so as to depart exactly at 06:30am (the ending time of the disruption) from stations MON or VEV. Further, the most common station for partial cancellations and turn arounds is the station right before the disruption (VEV for the first train, MON for the two others), which, again, makes sense from a practical point of view.

Conversely, the algorithm also produces less "intuitive" timetables, such as turning around train S2 12211 in station COS or delaying train S3 12314 by more than 35 minutes in order to pass the disrupted area. The fact that these solutions are non-dominated indicates that train operating companies might want to implement (or at least evaluate) them.

Additionally to the recovery actions applied to the three trains, we also examined the emergency buses and trains scheduled in the disposition timetables. On average, 5.8 buses are scheduled, and this number varies between 0 and 16, depending on the timetable. In 553 out of the 952 timetables, emergency trains are scheduled on the disrupted tracks after the disruption is resolved. Interestingly, no emergency train is scheduled before the beginning of the disruption at 05:30am, thus showing that the algorithm adapts to the low passenger demand at this time.

This in-depth analysis of the timetables constructed by our framework highlighted the importance of running the algorithm multiple times from the initial solution. Indeed, the manner in which the initial infeasibilities related to the disruption are handled influences the future iterations of the algorithm. For instance, if all disrupted trains are canceled in the first iteration by the feasibility restoration operators, they will not appear in most subsequent timetables.

5.4.4.2 Disruption LAU–VEV–PUI



(a) z_o–z_p projection (z_d in greyscale).

(b) z_o–z_d projection (z_p in greyscale).

(c) z_p–z_d projection (z_o in greyscale).

Figure 5.3: Pareto frontier projections for disruption LAU–VEV–PUI.

This section is dedicated to the discussion of the most severe disruption scenario introduced in Section 5.4.1. Results are presented at the aggregate level: timetables in the archive of non-dominated solutions are first compared using the Pareto frontier of the three objectives — passenger inconvenience (z_p), operational cost (z_o) and deviation from the undisrupted timetable (z_d). Then, the three timetables in the archive with the lowest value of z_p, z_o and z_d, respectively, are reported in Table 5.8 and compared in further detail. The complexity of the Pareto frontiers obtained by our framework requires a sound methodology for practitioners to choose which timetable to implement among the numerous non-dominated solutions. Two such methodologies are proposed in Section 5.4.5.

For this disruption scenario, the merged archive of non-dominated solutions contains 987 disposition timetables. Fig. 5.3 compares the performance of each of the timetables, in terms of the objective functions identified above. As the Pareto frontier is three-dimensional, we project it on the three planes ($z_d = 0, z_p = 0, z_o = 0$) for an easier visual interpretation, and the third objective is represented in greyscale.

It can be observed in Fig. 5.3a that the starkest trade-off between objectives is achieved on the z_o–z_p front: an increase in operational cost results (almost) automatically in a decrease in passenger inconvenience, regardless of the value of z_d. Also, by increasing the deviation from the undisrupted timetable, the two other objectives can be improved simultaneously. Note that it is not possible to identify a clear inflection point of the frontier (as in Fig. 3.1a). This may be explained by the fact that the cost of the opt-out arc has been lowered in this chapter, thus effectively limiting the sharp increase in passenger inconvenience when canceling trains (i.e., decreasing operational cost).

The trade-off in the two other frontiers is less clear. In the z_o–z_d projection (Fig. 5.3b), the minimal value of deviation from the undisrupted timetable lies at about $z_o =$ 70 kCHF. From this point, z_d can be increased in two ways: either by decreasing operational cost (thus increasing passenger inconvenience), or vice versa. The former situation corresponds to timetables where many trains are canceled, resulting in poor passenger satisfaction, whereas in the latter additional trains and buses are scheduled, increasing passenger satisfaction, but coming at a cost. Finally, in the z_p–z_d projection (Fig. 5.3c), it can be observed that, for a given value of operational cost, an increase in z_d results in a decrease in passenger inconvenience. This indicates that the operators introduced in our framework are capable of scheduling trains that are adjusted to passenger demand.

Table 5.8: Values of the objective functions, and level of unsatisfied demand, for the timetables with lowest value of z_p, z_o and z_d in $\mathcal{A}$.

z_p [min]	z_o [CHF]	z_d [-]	Unsatisfied demand
934,290.0	104,970.0	67,883.0	2155
1,486,869.5	**32,850.0**	109,497.0	10,526
1,069,571.5	71,670.0	**31,716.0**	3524

Table 5.8 reports the values of the objective functions at the minimal extremes of the Pareto frontier, as well as the level of unsatisfied demand, defined as the number of passengers that need to take the opt-out path. For comparison, in the undisrupted

timetable, $z_p = 901,924.5$ [min], $z_o = 92,160.0$ CHF and $z_d = 0$. For the timetable with lowest passenger inconvenience (first row of Table 5.8), z_p is only about 4% higher than in the undisrupted case. This level of service however requires a high operational cost. Indeed, 18 buses and 19 emergency trains are scheduled in this timetable. Even in this case, 2155 out of 14,920 passengers (14%) take the opt-out path. The timetable with minimal operational cost (second row of Table 5.8) is associated with an extremely low level of service for passengers: on average, travel times increase by about 64% and more than two thirds of the passengers need to find an alternative solution to travel. This timetable is “cheap” for the railway operator because only 30 original and 6 emergency trains are operated. In addition, all emergency trains run on train lines with at most two stations. The deviation cost associated with this timetable is very high, because many original trains are completely canceled. Finally, the timetable with the lowest deviation cost (third row of Table 5.8) represents an interesting compromise: z_p increases by 18% with respect to the undisrupted case, and the associated operational cost remains low. In fact, in this timetable, only one bus is scheduled, and not a single emergency train. Railway operators could therefore use this timetable as an initial feasible solution, and then include additional trains or buses in order to increase the level of service provided, especially to serve the unsatisfied demand.

5.4.5 Practical applicability

In conclusion, the results show that the proposed framework is capable of constructing high-quality feasible disposition timetables that make sense from a practical point of view. In addition, as can be seen in Fig. 5.4, the computational time of the algorithm remains low, even for the real case study based on the Swiss S-train network. Again, the highest computational times occur obviously with the local search operators, especially those minimizing passenger inconvenience, as the passenger assignment model needs to be rerun on each timetable in this particular case.

The algorithm described in this chapter can therefore be seen as a tool for railway operators to compare disposition timetables at an aggregate level. Its output is a list of timetables which are non-dominated. In order to choose the “best” among them, train operators wishing to use the framework need to decide the relative importance of the objectives to their eyes, or the maximal value of each objective that they are willing to accept (these values can be relative to the objective functions of the undisrupted timetable, as in the ε-constraints of Chapter 3). If one objective is significantly more important than the others, the operator would choose a timetable which minimizes this objective, while respecting the bounds on the other objectives. Otherwise, if the objectives are equivalently important, the timetable which is closest (e.g., in terms of Euclidian distance in the three-dimensional Pareto frontier) to the undisrupted timetable would be preferred.

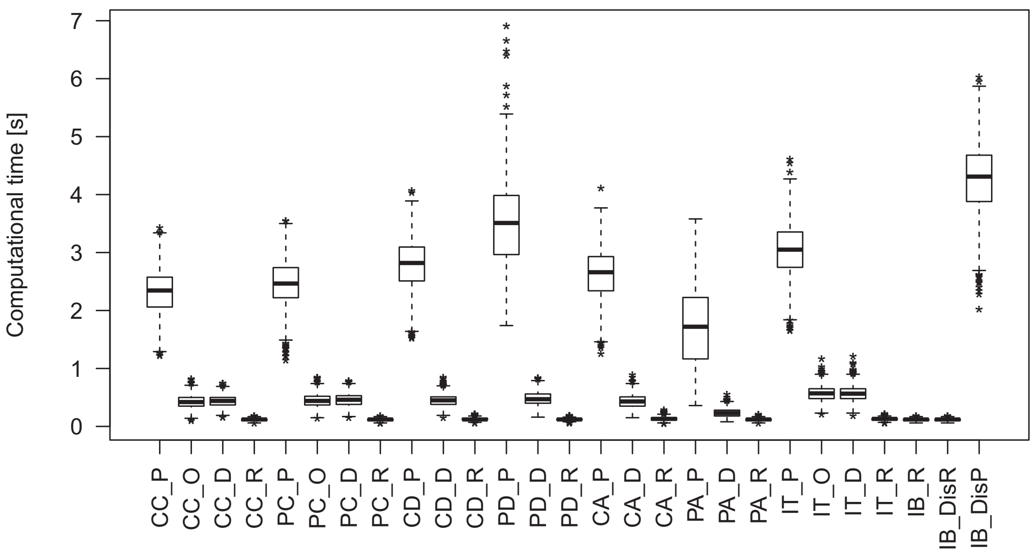


Figure 5.4: Distribution of computational times for disruption LAU–VEV–PUI.

We wish to stress here that each timetable in the archive of non-dominated solutions is the result of successive applications of simple recovery operators that are directly inspired from practice. It is therefore very easy for railway operators to implement a timetable from the archive, by retracing the elementary steps that led to it. Finally, note that operator-defined upper bounds on the objective functions can easily be included in the framework. This has the advantage of limiting the search for timetables to a solution space in which operators are guaranteed to obtain timetables in line with their requirements.

5.5 Concluding remarks

In this chapter, a meta-heuristic framework based on adaptive large neighborhood search was introduced for the multi-objective railway timetable rescheduling problem. Operators inspired directly from practice are implemented in order to modify the timetable. The algorithm constructs an approximation of the Pareto frontier of the problem, by populating an archive of non-dominated solutions.

Extensive computational experiments were performed on a case study of the S-train network of Canton Vaud, Switzerland. Our analysis indicates that each of the operators contributes at some point to find new non-dominated solutions. Additionally, it was shown that the framework performs well with respect to computation time and solution quality in many cases. Indeed, it reproduces recovery actions that railway operators use in practice for disruption management.

6

Concluding remarks

6.1 Main findings

The main goal of this thesis was to study the macroscopic impact of the inclusion of passenger demand in the railway timetable rescheduling problem. The results of this research indicated that taking the passengers into account is necessary for the design of “good” disposition timetables. Indeed, it was shown throughout this thesis that adopting a demand-oriented approach for the management of disruptions not only is possible, but may lead to significant improvement in passenger satisfaction (compared to a purely operator-centric framework), associated with a low operational cost of the disposition timetable. The presented models thus provide a quantitative understanding of the objectives of the various stakeholders of the problem, and therefore can be used for the improvement of the level of service offered to railway passengers.

In this thesis, we have proposed a novel methodology for the timetable rescheduling problem. Its major originality resides in the explicit consideration of the multiple objectives of the problem: passenger inconvenience, operational cost and deviation from the undisrupted timetable are three conflicting objectives that need to be minimized in the design of disposition timetables. The problem is defined formally as an Integer Linear Program in Chapter 3. By introducing ε-constraints into the problem, the three-dimensional Pareto frontier is explored to understand the trade-offs between the objectives. This exact solution approach provides optimal solutions for instances of limited size. However, solution methodologies that can obtain “good” but not necessarily optimal solutions are required for larger and more realistic instances. Additionally, a fast and efficient solution methodology is required, in order to be able to apply the framework in a practical context. A heuristic solution approach is therefore presented in Chapter 5: an adaptive large neighborhood meta-heuristic is adapted to the multi-objective context in order to approximate the Pareto frontier. A rich set of practice-inspired operators competes to find non-dominated timetables. The proposed solution approach performs well on large-scale practical problems, in terms of computational time and solution quality. Indeed, a detailed analysis of the solutions showed that the algorithm reproduces common railway operator recovery techniques (e.g., delaying a train by just the right amount of time so it can pass the disruption, or scheduling an additional train in case of high demand) to a large extent.

Another contribution of this thesis is the introduction of a flexible network loading framework for the capacitated passenger assignment problem arising in public transportation. The novelty of the approach, described in Chapter 4, is to define an exogenous ordering of the passengers before the assignment, thus extracting the complexity from the actual assignment problem. The impacts of five different priority rules are studied. For a given ordering, results show a remarkable stability across simulations of aggregate passenger satisfaction indicators, such as average travel time or level of unsatisfied demand. The realism of these orderings is also discussed, when comparing results with the FIFO ordering that takes place in practice. Railway operators can use these insights in order to reduce the overall passenger inconvenience, by imposing specific priorities. The stability of aggregate results and the efficiency of the assignment justify the use of the assignment algorithm in an iterative rescheduling framework, as well as in a practical context.

6.2 Applicability in practical context

One of the motivations for this work was the general need for more practice-oriented frameworks in the field of railway disruption management, in order to bridge the gap between practice and academic research identified by Caimi et al. (2012). In that respect, this thesis represents a significant step in this direction, in the following ways:

- it introduces models that are tested extensively on two realistic case studies;
- it provides a tool for the aggregate quantification of the objective functions of the problem;
- it uses recovery actions that are directly inspired from practical experience (e.g., turning around a train before the disrupted area);

- it generates a set of “good” disposition timetables in reasonable computational times.

It needs however to be emphasized that the achievements discussed so far rely on a macroscopic representation of the problem. In particular, conflict resolution is addressed at a low level of detail, ensuring sufficient headways between trains using the same track. Unfortunately, as the common saying goes, “the devil is in the detail”. Microscopic aspects, such as the block signalling system or the assignment of trains to single tracks in stations, are ignored in this thesis, and may therefore lead to potential conflicts among trains, especially in busy train stations, such as Lausanne (the main station of the Swiss case study). Also, the integration with the rolling stock and crew rescheduling problems is left to later stages of the disruption management process (see, e.g., Dollevoet et al., 2017, for an integrated version of the problem).

In this sense, this thesis can be seen as a high-level decision support tool for train operators faced with severe disruptions. Despite its ability to construct quickly a large set of “good” disposition timetables, it does not claim to replace nor automate the work of train dispatchers, but should rather be considered as a valuable tool to evaluate a priori the consequences of their decisions. In addition, the real disruption example described in Section 2.1 highlighted that proper coordination and communication is of critical importance for a successful management of the disruption. Questions such as what to do with potentially broken-down rolling stock, how to guide efficiently (disrupted) passengers, etc., need to be answered quickly and reasonably. Also, based on the severity of a disruption (which is difficult to define in a mathematical way), an experienced dispatcher might know in advance which recovery actions are feasible, implementable, and publicly acceptable, and which are not. The local expertise of operators is thus crucial in these time-constrained moments, and our framework can do only so much to help in that case. It nevertheless makes significant contributions to solve a highly complex problem.

6.3 Directions for future research

The main objective of this thesis was to include passenger demand in the design of railway disposition timetables in case of severe disruptions. It therewith lays an initial stepping stone for a broad and challenging research agenda. We have identified four directions along which further research may be organized.

Modeling assumptions A model is always a simplified representation of a real phenomenon or process, with the purpose of studying its characteristics. This thesis is no exception: a number of simplifying assumptions had to be made in order to model the problem.

First, it is assumed that the start and end times of the disruption are known in advance, which is rarely the case in practice. We note that the uncertain duration of the disruption can be considered by embedding our model in a rolling horizon framework (see, e.g., Nielsen et al., 2012): in this case, rescheduling decisions are only considered if they are within a given time horizon from the beginning of the disruption, and the schedule is then revised as time passes and new information becomes available. In the case of disruptions of unknown length, it may also be necessary to consider the so-called "Traveler's Route Choice Problem", defined in Schmidt et al. (2017), where a traveler in a railway system is faced with a decision problem under uncertainty: he can wait until the disruption is (hopefully) over, or take a longer detour route as an alternative.

Second, the passenger assignment model may not be ideal for disrupted situations. For instance, it was assumed that the passenger demand does not change when the disruption occurs. In reality however, passengers might adjust their destination, their desired departure time, or even their chosen travel mode in a disrupted situation. Hence, an interesting extension would be to account for the shift in the passenger demand following the announcement of the disruption. In particular, considering the "indifference threshold" (Liu and Zhou, 2016) up to which a passenger prefers to stick with his current choice instead of selecting another (potentially shorter) alternative, would be an interesting extension.

Third, the passengers are considered to be homogeneous in this research. In reality however, the behavior of each passenger may be different. This can be captured by varying the multiplying factors of the generalized travel time defining the route choice of every passenger, and by introducing socio-economic variables such as income, age or education. Passenger heterogeneity can thus be treated as a future extension; analyzing how passengers react to the disruption, based on their socio-economic characteristics provides an interesting field of future research. The use of more advanced choice models would be required in this case.

Finally, we assumed that the stochastic parts of the importance function ordering the passengers in Chapter 4 are independent and identically distributed across the passengers. Relaxing this assumption may allow to model more complex behavioral assumptions.

Methodological extensions In the exact formulation of the problem, we have ignored the solution methodology side of the problem and rather focused on the general concept. Hence, a natural extension would be to aim at a more efficient solving of the problem. We see decomposition methods as one promising option that could allow to solve problems with larger time horizons and, critically, more passengers.

Further, our exploration of the Pareto frontier is partial and might miss non-dominated solutions. The use of an exact algorithm to explore the three-dimensional Pareto frontier is beyond the scope of this thesis, but would definitely be an interesting direction for future research.

Regarding the meta-heuristic, a potential methodological extension to increase the algorithm's efficiency would be to approximate the values of the objective functions instead of calculating them exactly at every iteration. Another extension would be to normalize the probability of selecting an operator by the time it requires to find a neighbor solution, thus reducing the use of overly time-consuming operators.

Data requirements While working on this thesis, the difficulty to obtain passenger demand data from railway operators was noticed recurrently. Eventually, a non-homogeneous Poisson process with realistic assumptions was used to generate dynamic origin-destination passenger demand. We can only hope that demand data becomes more largely available, as the popularity and visibility of passenger-centric transportation systems grow. Open data platforms such as `https://tfl.gov.uk/info-for/open-data-users/` and `https://data.sbb.ch` are a welcome step in this direction for the future of transportation research. In addition, data obtained with new collection techniques, such as automated fare collection systems or smartphone tracking, is expected to increase significantly in the coming decades. The gathering of socio-economic information about the passengers is also conceivable with these novel techniques, as long as privacy concerns are properly addressed.

Additionally, the close collaboration with a railway operator is suggested as a future extension of this work. Indeed, this thesis requires a number of generic parameter values (e.g., train driving cost per kilometer, cost of canceling or delaying a train, or scheduling an additional train). The input of the railway operator is necessary to obtain the "true" values of these parameters, and thus be able to quantify correctly the objectives of the problem. Also, including in the model the cost of the increasingly common "compensation payments", following the introduction of Directive 2001/14/EC by the European Communities, may be of great interest to railway operators.

Applications to other fields The modeling framework presented in this thesis is directly suited for passenger-centric railway timetable rescheduling. Thanks to its flexibility, it can however easily be transferred to other fields with similar characteristics. In particular, it can be applied for (re)scheduling in any field where human factors appear and multi-objective optimization is required, such as bus networks, airline networks, hospital shift (re)scheduling, etc., by incorporating the constraints that are specific to that field.

Bibliography

Acuna-Agost, R., Michelon, P., Feillet, D. and Gueye, S. (2011). A MIP-based local search method for the railway rescheduling problem, *Networks* **57**(1): 69–86.

Albrecht, A., Panton, D. and Lee, D. (2013). Rescheduling rail networks with maintenance disruptions using Problem Space Search, *Computers & Operations Research* **40**(3): 703–712.

Audet, C., Dang, K.-C. and Orban, D. (2014). Optimization of algorithms with OPAL, *Mathematical Programming Computation* **6**(3): 233–254.

Barrena, E., Canca, D., Coelho, L. C. and Laporte, G. (2014). Single-line rail rapid transit timetabling under dynamic passenger demand, *Transportation Research Part B: Methodological* **70**: 134–150.

Ben-Akiva, M. E. and Lerman, S. R. (1985). *Discrete choice analysis: theory and application to travel demand*, number 9 in *MIT Press series in transportation studies*, MIT Press, Cambridge, Mass.

Bérubé, J.-F., Gendreau, M. and Potvin, J.-Y. (2009). An exact epsilon-constraint method for bi-objective combinatorial optimization problems: Application to the Traveling Salesman Problem with Profits, *European Journal of Operational Research* **194**(1): 39–50.

Bierlaire, M. (2015). *Optimization: Principles and Algorithms*, EPFL Press, Lausanne.

Bierlaire, M. and Crittin, F. (2006). Solving Noisy, Large-Scale Fixed-Point Problems and Systems of Nonlinear Equations, *Transportation Science* **40**(1): 44–63.

Boccia, M., Mannino, C. and Vasilyev, I. (2013). The dispatching problem on multitrack territories: Heuristic approaches based on mixed integer linear programming, *Networks* **62**(4): 315–326.

Bottom, J. A. (2000). *Consistent anticipatory route guidance*, phdthesis, Massachusetts Institute of Technology.

Brucker, P., Heitmann, S. and Knust, S. (2002). Scheduling railway traffic at a construction site, *OR Spectrum* **24**(1): 19–30.

Burke, E. K., Gendreau, M., Hyde, M., Kendall, G., Ochoa, G., Özcan, E. and Qu, R. (2013). Hyper-heuristics: a survey of the state of the art, *Journal of the Operational Research Society* **64**(12): 1695–1724.

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

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

Caimi, G., Fuchsberger, M., Laumanns, M. and Lüthi, M. (2012). A model predictive control approach for discrete-time rescheduling in complex central railway station areas, *Computers & Operations Research* **39**(11): 2578–2593.

Charkhgard, H. (2016). *Theory and Algorithms for Multi-objective Integer Programming*, phdthesis, The University of Newcastle.

Cominetti, R. and Correa, J. (2001). Common-Lines and Passenger Assignment in Congested Transit Networks, *Transportation Science* **35**(3): 250–267.

Corman, F. (2010). *Real-time railway traffic management: dispatching in complex, large and busy railway networks*, PhD thesis, Technische Universiteit Delft, Delft, The Netherlands.

Corman, F., D'Ariano, A., Hansen, I. A., Pacciarelli, D. and Pranzo, M. (2011). Dispatching trains during seriously disrupted traffic situations, *IEEE International Conference on Networking, Sensing and Control*, Delft, pp. 323–328.

Corman, F., D'Ariano, A., Marra, A. D., Pacciarelli, D. and Samà, M. (2016). Integrating train scheduling and delay management in real-time railway traffic control, *Transportation Research Part E: Logistics and Transportation Review* .

Corman, F., D'Ariano, A., Pacciarelli, D. and Pranzo, M. (2009). Evaluation of green wave policy in real-time railway traffic management, *Transportation Research Part C: Emerging Technologies* **17**(6): 607–616.

Corman, F., D'Ariano, A., Pacciarelli, D. and Pranzo, M. (2010a). Centralized versus distributed systems to reschedule trains in two dispatching areas, *Public Transport* **2**(3): 219–247.

Corman, F., D'Ariano, A., Pacciarelli, D. and Pranzo, M. (2010b). A tabu search algorithm for rerouting trains during rail operations, *Transportation Research Part B: Methodological* **44**(1): 175–192.

Corman, F., D'Ariano, A., Pacciarelli, D. and Pranzo, M. (2012). Bi-objective conflict detection and resolution in railway traffic management, *Transportation Research Part C: Emerging Technologies* **20**(1): 79–94.

Corman, F., D'Ariano, A., Pacciarelli, D. and Pranzo, M. (2014). Dispatching and coordination in multi-area railway traffic management, *Computers & Operations Research* **44**: 146–160.

Corman, F. and Quaglietta, E. (2015). Closing the loop in real-time railway control: Framework design and impacts on operations, *Transportation Research Part C: Emerging Technologies* **54**: 15–39.

Cormen, T. H., Leiserson, C. E., Rivest, R. L. and Stein, C. (2009). *Introduction to algorithms*, 3rd ed edn, MIT Press, Cambridge, Mass.

D'Ariano, A., Corman, F., Pacciarelli, D. and Pranzo, M. (2008). Reordering and Local Rerouting Strategies to Manage Train Traffic in Real Time, *Transportation Science* **42**(4): 405–419.

D'Ariano, A., Pacciarelli, D. and Pranzo, M. (2007). A branch and bound algorithm for scheduling trains in a railway network, *European Journal of Operational Research* **183**(2): 643–657.

Dayarian, I., Crainic, T. G., Gendreau, M. and Rei, W. (2016). An adaptive large-neighborhood search heuristic for a multi-period vehicle routing problem, *Transportation Research Part E: Logistics and Transportation Review* **95**: 95–123.

de Cea, J. and Fernández, E. (1993). Transit Assignment for Congested Public Transport Systems: An Equilibrium Model, *Transportation Science* **27**(2): 133–147.

de Keizer, B., Geurs, K. and Haarsman, G. (2012). Interchanges in timetable design of railways: A closer look at customer resistance to interchange between trains, *40th European Transport Conference*, Glasgow.

Dollevoet, T. (2013). *Delay Management and Dispatching in Railways*, PhD thesis, Erasmus University, Rotterdam.

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

Dollevoet, T., Huisman, D., Schmidt, M. and Schöbel, A. (2012). Delay Management with Rerouting of Passengers, *Transportation Science* **46**(1): 74–89.

Dorbritz, R. (2012). *Methodology for assessing the structural and operational robustness of railway networks*, PhD thesis, ETH Zurich.

European Commission (2013). Europeans' satisfaction with rail services, *Technical Report Flash Eurobarometer 382a*.

Eurostat (2017). Modal split of passenger transport (Data table tsdtr210).
URL: *http://ec.europa.eu/eurostat/tgm/table.do?tab=table&plugin=1&language=en&pcode=tsdtr210*

Fu, Q., Liu, R. and Hess, S. (2012). A Review on Transit Assignment Modelling Approaches to Congested Networks: A New Perspective, *Procedia - Social and Behavioral Sciences* **54**: 1145–1155.

Gao, Z., Sun, H. and Shan, L. L. (2004). A continuous equilibrium network design model and algorithm for transit systems, *Transportation Research Part B: Methodological* **38**(3): 235–250.

Gendreau, M. (1984). *Etude approfondie d'un modèle d'équilibre pour l'affectation des passagers dans les réseaux de transport en commun*, PhD thesis, Université de Montréal, Centre de recherche sur les transports, Montréal, Canada.

Hamdouch, Y., Ho, H., Sumalee, A. and Wang, G. (2011). Schedule-based transit assignment model with vehicle capacity and seat availability, *Transportation Research Part B: Methodological* **45**(10): 1805–1830.

Hamdouch, Y. and Lawphongpanich, S. (2008). Schedule-based transit assignment model with travel strategies and capacity constraints, *Transportation Research Part B: Methodological* **42**(7-8): 663–684.

Hamdouch, Y., Marcotte, P. and Nguyen, S. (2004). Capacitated transit assignment with loading priorities, *Mathematical Programming* **101**(1).

Hamdouch, Y., Szeto, W. and Jiang, Y. (2014). A new schedule-based transit assignment model with travel strategies and supply uncertainties, *Transportation Research Part B: Methodological* **67**: 35–67.

Hansen, I. A. and Pachl, J. (eds) (2014). *Railway timetabling & operations: analysis, modelling, optimisation, simulation, performance evaluation*, Eurailpress, Hamburg.

Hirai, C., Kunimatsu, T., Tomii, N., Kondou, S. and Takaba, M. (2009). A Train Stop Deployment Planning Algorithm Using a Petri-net-based Modelling Approach, *Quarterly Report of RTRI* **50**(1): 8–13.

Kirkpatrick, S., Gelatt, C. D. and Vecchi, M. P. (1983). Optimization by simulated annealing, *Science* **220**(4598): 671–680.

Kirlik, G. and Sayın, S. (2014). A new algorithm for generating all nondominated solutions of multiobjective discrete optimization problems, *European Journal of Operational Research* **232**(3): 479–488.

Kroon, L., Maróti, G. and Nielsen, L. (2015). Rescheduling of Railway Rolling Stock with Dynamic Passenger Flows, *Transportation Science* **49**(2): 165–184.

Krzywinski, M., Schein, J., Birol, I., Connors, J., Gascoyne, R., Horsman, D., Jones, S. J. and Marra, M. A. (2009). Circos: An information aesthetic for comparative genomics, *Genome Research* **19**(9): 1639–1645.

Laumanns, M., Thiele, L. and Zitzler, E. (2006). An efficient, adaptive parameter variation scheme for metaheuristics based on the epsilon-constraint method, *European Journal of Operational Research* **169**(3): 932–942.

Liu, J. and Zhou, X. (2016). Capacitated transit service network design with boundedly rational agents, *Transportation Research Part B: Methodological* **93**: 225–250.

Louwerse, I. and Huisman, D. (2014). Adjusting a railway timetable in case of partial or complete blockades, *European Journal of Operational Research* **235**(3): 583–593.

Lusby, R. M., Larsen, J., Ehrgott, M. and Ryan, D. M. (2013). A set packing inspired method for real-time junction train routing, *Computers & Operations Research* **40**(3): 713–724.

Magnanti, T. L. and Perakis, G. (2004). Solving variational inequality and fixed point problems by line searches and potential optimization, *Mathematical Programming* **101**(3): 435–461.

Mannino, C. and Mascis, A. (2009). Optimal Real-Time Traffic Control in Metro Stations, *Operations Research* **57**(4): 1026–1039.

Mascis, A. and Pacciarelli, D. (2002). Job-shop scheduling with blocking and no-wait constraints, *European Journal of Operational Research* **143**(3): 498–517.

Mavrotas, G. (2009). Effective implementation of the epsilon-constraint method in Multi-Objective Mathematical Programming problems, *Applied Mathematics and Computation* **213**(2): 455–465.

Mavrotas, G. and Florios, K. (2013). An improved version of the augmented epsilon-constraint method (AUGMECON2) for finding the exact pareto set in multi-objective integer programming problems, *Applied Mathematics and Computation* **219**(18): 9652–9669.

Meng, L. and Zhou, X. (2014). Simultaneous train rerouting and rescheduling on an N-track network: A model reformulation with network-based cumulative flow variables, *Transportation Research Part B: Methodological* **67**: 208–234.

Min, Y.-H., Park, M.-J., Hong, S.-P. and Hong, S.-H. (2011). An appraisal of a column-generation-based algorithm for centralized train-conflict resolution on a metropolitan railway network, *Transportation Research Part B: Methodological* **45**(2): 409–429.

Narayanaswami, S. and Rangaraj, N. (2013). Modelling disruptions and resolving conflicts optimally in a railway schedule, *Computers & Industrial Engineering* **64**(1): 469–481.

Ngatchou, P., Zarei, A. and El-Sharkawi, A. (2005). Pareto Multi Objective Optimization, IEEE, pp. 84–91.

Nguyen, S. and Pallottino, S. (1988). Equilibrium traffic assignment for large scale transit networks, *European Journal of Operational Research* **37**(2): 176–186.

Nguyen, S., Pallottino, S. and Malucelli, F. (2001). A Modeling Framework for Passenger Assignment on a Transport Network with Timetables, *Transportation Science* **35**(3): 238–249.

Nielsen, L. K., Kroon, L. and Maróti, G. (2012). A rolling horizon approach for disruption management of railway rolling stock, *European Journal of Operational Research* **220**(2): 496–509.

Nielsen, O. A. (2004). A Large Scale Stochastic Multi-Class Schedule-Based Transit Model with Random Coefficients, *in* N. H. M. Wilson and A. Nuzzolo (eds), *Schedule-Based Dynamic Transit Modeling: theory and applications*, Vol. 28, Springer US, Boston, MA, pp. 53–77.

Nuzzolo, A., Crisalli, U. and Rosati, L. (2012). A schedule-based assignment model with explicit capacity constraints for congested transit networks, *Transportation Research Part C: Emerging Technologies* **20**(1): 16–33.

Nuzzolo, A., Russo, F. and Crisalli, U. (2001). A Doubly Dynamic Schedule-based Assignment Model for Transit Networks, *Transportation Science* **35**(3): 268–285.

Özlen, M. and Azizoğlu, M. (2009). Multi-objective integer programming: A general approach for generating all non-dominated solutions, *European Journal of Operational Research* **199**(1): 25–35.

Papola, N., Filippi, F., Gentile, G. and Meschini, L. (2009). Schedule-based transit assignment: new dynamic equilibrium model with vehicle capacity constraints, *Schedule-Based Modeling of Transportation Networks: Theory and applications*, Springer US, Boston, MA, pp. 1–26.

Poon, M., Wong, S. and Tong, C. (2004). A dynamic schedule-based model for congested transit networks, *Transportation Research Part B: Methodological* **38**(4): 343–368.

Rijden de Treinen (2017). Number of disruptions and total duration.
URL: *https://www.rijdendetreinen.nl/en/statistics*

Robenek, T., Maknoon, Y., Azadeh, S. S., Chen, J. and Bierlaire, M. (2016). Passenger centric train timetabling problem, *Transportation Research Part B: Methodological* **89**: 107–126.

Robenek, T., Sharif Azadeh, S., Maknoon, Y. and Bierlaire, M. (2017). Hybrid cyclicity: Combining the benefits of cyclic and non-cyclic timetables, *Transportation Research Part C: Emerging Technologies* **75**: 228–253.

Ropke, S. and Pisinger, D. (2006). An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows, *Transportation Science* **40**(4): 455–472.

Samà, M., D'Ariano, A., Corman, F. and Pacciarelli, D. (2017). A variable neighbourhood search for fast train scheduling and routing during disturbed railway traffic situations, *Computers & Operations Research* **78**: 480–499.

Sato, K., Tamura, K. and Tomii, N. (2013). A MIP-based timetable rescheduling formulation and algorithm minimizing further inconvenience to passengers, *Journal of Rail Transport Planning & Management* **3**(3): 38–53.

Schachtebeck, M. and Schöbel, A. (2010). To Wait or Not to WaitAnd Who Goes First? Delay Management with Priority Decisions, *Transportation Science* **44**(3): 307–321.

Schmidt, M., Kroon, L., Schöbel, A. and Bouman, P. (2017). The Travelers Route Choice Problem Under Uncertainty: Dominance Relations Between Strategies, *Operations Research* **65**(1): 184–199.

Schöbel, A. (2001). A Model for the Delay Management Problem based on Mixed-Integer-Programming, *Electronic Notes in Theoretical Computer Science* **50**(1): 1–10.

Schöbel, A. (2009). Capacity constraints in delay management, *Public Transport* **1**(2): 135–154.

Small, K. A. (1982). The scheduling of consumer activities: Work trips, *The American Economic Review* **72**(3): 467–479.

Spiess, H. and Florian, M. (1989). Optimal strategies: A new assignment model for transit networks, *Transportation Research Part B: Methodological* **23**(2): 83–102.

Sumalee, A., Tan, Z. and Lam, W. H. (2009). Dynamic stochastic transit assignment with explicit seat allocation model, *Transportation Research Part B: Methodological* **43**(8-9): 895–912.

Suppapitnarm, A., Seffen, K. A., Parks, G. T. and Clarkson, P. J. (2000). A simulated annealing algorithm for multi-objective optimization, *Engineering Optimization* **33**(1): 59–85.

Swiss Federal Railways (2015). SBB: Facts and Figures 2015, *Technical report*.

Tirachini, A., Hensher, D. A. and Rose, J. M. (2013). Crowding in public transport systems: Effects on users, operation and implications for the estimation of demand, *Transportation Research Part A: Policy and Practice* **53**: 36–52.

Toletti, A., Laumanns, M., De Martinis, V. and Weidmann, U. (2017). Multicriteria train rescheduling by means of an adaptive epsilon-constraint method, *Proceedings 6th International Conference on Railway Operations Modelling and Analysis (RailLille2017), Lille, France, April 4-7, 2017*.

Toletti, A. and Weidmann, U. (2016). Modelling customer inconvenience in train rescheduling, *Swiss Transport Research Conference (STRC2016)*.

Tong, C. and Wong, S. (1999). A stochastic transit assignment model using a dynamic schedule-based network, *Transportation Research Part B: Methodological* **33**(2): 107–121.

Törnquist, J. and Persson, J. A. (2007). N-tracked railway traffic re-scheduling during disturbances, *Transportation Research Part B: Methodological* **41**(3): 342–362.

Törnquist Krasemann, J. (2012). Design of an effective algorithm for fast response to the re-scheduling of railway traffic during disturbances, *Transportation Research Part C: Emerging Technologies* **20**(1): 62–78.

Veelenturf, L. P., Kidd, M. P., Cacchiani, V., Kroon, L. G. and Toth, P. (2016). A Railway Timetable Rescheduling Approach for Handling Large-Scale Disruptions, *Transportation Science* **50**(3): 841–862.

Veelenturf, L. P., Kroon, L. G. and Maróti, G. (2017). Passenger oriented railway disruption management by adapting timetables and rolling stock schedules, *Transportation Research Part C: Emerging Technologies* **80**: 133–147.

Vidal, T., Crainic, T. G., Gendreau, M., Lahrichi, N. and Rei, W. (2012). A Hybrid Genetic Algorithm for Multidepot and Periodic Vehicle Routing Problems, *Operations Research* **60**(3): 611–624.

Voegeli, M. (2014). *Integrated Timetabling and Passenger Assignment for Capacitated Public Transportation Networks*, Master thesis, Swiss Federal Institute of Technology, Zurich.

Wardman, M. (2004). Public transport values of time, *Transport Policy* **11**(4): 363–377.

Wardman, M. and Whelan, G. (2011). Twenty Years of Rail Crowding Valuation Studies: Evidence and Lessons from British Experience, *Transport Reviews* **31**(3): 379–398.

Xu, P., Corman, F., Peng, Q. and Luan, X. (2017). A train rescheduling model integrating speed management during disruptions of high-speed traffic under a quasi-moving block system, *Transportation Research Part B: Methodological* .

Zhan, S., Kroon, L. G., Zhao, J. and Peng, Q. (2016). A rolling horizon approach to the high speed train rescheduling problem in case of a partial segment blockage, *Transportation Research Part E: Logistics and Transportation Review* **95**: 32–61.

Zhan, S., Kroon, L., Veelenturf, L. P. and Wagenaar, J. C. (2015). Real-time high-speed train rescheduling in case of a complete blockage, *Transportation Research Part B: Methodological* **78**: 182–201.

Stefan Binder

Transport and Mobility Laboratory – TRANSP-OR
School of Architecture, Civil and Environmental Engineering, Station 18
Ecole Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland

☎ (+41) 21 693 2532 • ✉ s.binder@epfl.ch
Website: http://transp-or.epfl.ch/personnal.php?Person=BINDER

Education

Ecole Polytechnique Fédérale de Lausanne — **Lausanne, Switzerland**
Doctoral degree (PhD) — *2013–2017*

Georgia Institute of Technology — **Atlanta, GA**
Master thesis — *August 2012–December 2012*
An application of targeted marketing data to model emissions failures

Ecole Polytechnique Fédérale de Lausanne — **Lausanne, Switzerland**
MSc in Environmental Sciences and Engineering — *2010–2013*
Minor : Urban Planning and Territorial Development

KTH Royal Institute of Technology — **Stockholm, Sweden**
Erasmus exchange year — *2008–2009*

Ecole Polytechnique Fédérale de Lausanne — **Lausanne, Switzerland**
BSc in Physics — *2006–2009*

Institut Mont-Olivet — **Lausanne, Switzerland**
Federal maturity — *2003–2006*
Bilingual mention: French/German

Professional Experience

Transitec Ingénieurs-Conseils SA — **Lausanne, Switzerland**
Industrial internship — *Summer 2012*
Analysis of the effects of a new residential area on the surrounding road network

Weinmann-Energies SA — **Echallens, Switzerland**
Industrial internship — *Spring 2010 and Spring 2012*
Development of energetic certificates for new and renovated buildings

Languages

English: Full professional proficiency — *TOEFL*
French: Native — *Father tongue*
Geman: Native — *Mother tongue*

Computer skills

Maths: Matlab
Languages: C#, Java

Statistics: R
Other: LaTeX, Office, Lesosai, Simapro

Publications

Papers in international journals

1. Binder, S., Maknoon, Y., and Bierlaire, M. (2017). Exogenous priority rules for the capacitated passenger assignment problem, Transportation Research Part B: Methodological **105**:19-42.
2. Binder, S., Maknoon, Y., and Bierlaire, M. (2017). The multi-objective railway timetable rescheduling problem, Transportation Research Part C: Emerging Technologies **78**:78-94.
3. Binder, S., Macfarlane, G. S., Garrow, L. A., and Bierlaire, M. (2014). Associations among Household Characteristics, Vehicle Characteristics and Emissions Failures: An Application of Targeted Marketing Data, Transportation Research Part A: Policy and Practice **59**:122-133.

Book chapters

1. Binder, S., and Ravalet, E. (2017). Quel est le prix d'une minute gagnée dans nos déplacements ?, In Bierlaire, Kaufmann, Rérat (ed) La mobilité en questions pp.37-52. Presses Polytechniques et Universitaires Romandes (PPUR).
2. Audikana, A., and Binder, S. (2017). Faut-il penser les transports pour qu'ils soient les plus rapides possibles ?, In Bierlaire, Kaufmann, Rérat (ed) La mobilité en questions pp.23-36. Presses Polytechniques et Universitaires Romandes (PPUR).

Papers in conference proceedings

1. Binder, S., Maknoon, Y., and Bierlaire, M. (2017). Efficient investigation of multiple dimensions of the railway timetable rescheduling problem. Proceedings of the 17th Swiss Transportation Research Conference, May 17-19th, 2017.
2. Binder, S., Maknoon, Y., and Bierlaire, M. (2015). Passenger-oriented railway disposition timetables in case of severe disruptions. Proceedings of the 15th Swiss Transportation Research Conference, April 15-17th, 2015.
3. Binder, S., Maknoon, Y., Sharif Azadeh, S., and Bierlaire, M. (2015). Generation and evaluation of passenger-oriented railway disposition timetables in case of severe disruptions. Proceedings of the 6th International Conference on Railway Operations Modelling and Analysis (RailTokyo2015), March 23-26th, 2015.
4. Bonnet, F., Binder, S., Elias de Oliveira, M., Halloy, J., and Mondada, F. (2014). A Miniature Mobile Robot Developed to be Socially Integrated with Species of Small Fish. Proceedings of the IEEE International Conference on Robotics and Biomimetics, December 5-10th, 2014.
5. Binder, S., Jiang Hang, C., and Bierlaire, M. (2014). Generation and evaluation of passenger-oriented railway disposition timetables. Proceedings of the Swiss Transportation Research Conference (STRC) May 14-16th, 2014.

Teaching

Optimization and simulation
EPFL, PhD course *Spring 2017*

Decision-aid methodologies in transportation
EPFL, Master course *Spring 2017*

Global Issues: Mobility
EPFL, Bachelor course *Spring 2013–Spring 2017*

Modeling of transportation systems
EPFL, Master course *Fall 2014*

Student project supervision

Master theses

1. *Demand-based rolling stock allocation (2017)*. Salma Deruiche (EPFL).
2. *Optimisation des tournées de ramassage des employés de l'aéroport de Genève (2014)*. Isabel Tovar (EPFL).

Semester projects

1. *On the optimization of CAPEX and OPEX for the design of a full electric large capacity urban bus system (2017)*. Guillaume Mollard (EPFL).
2. *Price of anarchy in public transit networks (2016)*. Marc-Edouard Schultheiss (EPFL).
3. *Online estimation of pedestrian origin-destination demand in train stations using Kalman Filtering (2016)*. Marc Solsona Bernet (EPFL).
4. *Routing of a mixed fleet of electric and diesel trucks: Analysis of solution approaches (2015)*. Noortje Verstegen (EPFL).
5. *Routing of a mixed fleet of electric and internal combustion trucks (2014)*. Thomas Cibils (EPFL).
6. *Train Management in SNCF Application (2014).* Cao Huu-An (EPFL).