Traffic management optimization of railway networks

Luan, Xiaojie

DOI
10.4233/uuid:4f6bf5d8-e75f-4627-ab0d-5e859f37a5fd

Publication date
2019

Document Version
Final published version

Citation (APA)
Luan, X. (2019). *Traffic management optimization of railway networks*. [Dissertation (TU Delft), Delft University of Technology]. TRAIL Research School. https://doi.org/10.4233/uuid:4f6bf5d8-e75f-4627-ab0d-5e859f37a5fd

Traffic Management Optimization of Railway Networks

Xiaojie LUAN

Delft University of Technology, 2019

Traffic Management Optimization of Railway Networks

Dissertation

for the purpose of obtaining the degree of doctor

at Delft University of Technology

by the authority of the Rector Magnificus, prof. dr. ir. T.H.J.J. van der Hagen,

chair of the Board for Doctorates

to be defended publicly on Monday, 1 July 2019 at 15.00 o'clock

by

Xiaojie LUAN

Master of Science in Traffic and Transportation Planning and Management

Beijing Jiaotong University, P.R.China

born in Yantai, ShanDong, P.R.China

This dissertation has been approved by the promotors:
Promotor: Prof. dr. ir. G. Lodewijks
Promotor: Prof. dr. ir. B. De Schutter

Composition of the doctoral committee:

Rector Magnificus	Chairman
Prof. dr. ir. G. Lodewijks	Delft University of Technology, promotor
Prof. dr. ir. B. De Schutter	Delft University of Technology, promotor

Other members:

Prof. dr. F. Corman	ETH Zürich

Independent members:

Prof. dr. R.M.P. Goverde	Delft University of Technology
Prof. dr. ir. R.P.B.J. Dollevoet	Delft University of Technology
Prof. dr. D. Huisman	Erasmus University Rotterdam
Prof. dr. D. Pacciarelli	Roma Tre University

This thesis is the result from a project funded by the China Scholarship Council [Grant number 201507090058], the section of Transpiration Engineering and Logistics, and the Delft Center for Systems and Control at the Department of Maritime and Transport Technology, Delft University of Technology in the Netherlands.

TRAIL Thesis Series no. T2019/10, The Netherlands TRAIL Research School

TRAIL
P.O. Box 5017
2600 GA Delft
The Netherlands
E-mail: info@rsTRAIL.nl

ISBN 978-90-5584-252-0

Printed in The Netherlands

Preface

I am always saying that I am so lucky and blessed to have so many people in my life, who believe in me, guide me, support me, and bear with me. I want to take this opportunity to express my sincere gratitude to them.

First of all, I would like to express my gratitude to my promoters and my supervisor:

Dear promoter Prof. Gabriel Lodewijks, thank you very much for giving me the opportunity to conduct research under your supervision. Your trust, professional guidance, inspiring discussions, and innovative ideas play a key role in my research. Especially thank you for your continues support after your migration to Australia.

Dear promoter Prof. Bart De Schutter, thank you for accepting me as your PhD student and allow me to work under your supervision. I think I was lucky enough to join your research team which helps me to go through the hardest time during my research and further promote my research to another level. Thank you very much for your timely feedback on my work. Your scientific way of thinking, attention on details, and broaden knowledge not only support my current research but also help me to become a better researcher.

Dear supervisor Prof. Francesco Corman, thanks for your support and guidance during the past years. When I was a master student, I got a chance to conduct research works under your supervision, which becomes the main motivation for me to pursue a PhD under your supervision. I enjoy your way of working. On one hand, you give me sufficient space and freedom to explore different ideas. On the other hand, you also carefully steer my research direction to make sure I am on the right track. The discussions we had onsite and via Skype are always inspiring and fruitful, helping me to solve my research problems and inspiring me with new insights and directions. Moreover, I am also grateful that we can keep our cooperation even when you work at IBM and move to Zurich.

In addition, I would like to express my gratitude to Prof. Linyun Meng, the supervisor of my master study at Beijing Jiaotong University. Thank you for always being supportive to me professionally and personally. You provide me lots of helpful advices, as well as great opportunities to reach out to the professionals in the field so that I get access to international education resources. Besides that, your words always encourage me when I am unconfident. The things I have achieved is due in no small part to your support and encouragement. I appreciate you so much and value everything that I have learned from you.

Also, I would like to thank all the persons who are involved in my research during the past four years. Dear Yihui, Ton, and Jianrui, thank you so much for your assistance. The effort you put in was really appreciated.

I would like to thank all the committee members, Dear Prof. Dennis Huisman, Prof. Rob Goverde, Prof. Rolf Dollevoet, and Prof. Dario Pacciarelli, for your time and efforts on commenting my draft thesis. The quality of this thesis is definitely improved by incorporating your comments.

To all my colleges in the Transport Engineering and Logistics Section, thank you all and it has been a delight for me to work in such a friendly and inspiring section. Dear Dick, thank you for helping me translate the thesis Summary into Samenvatting. Dear Dineke and Patty, thanks for your support during my entire research in TU Delft.

I am also so grateful to my mentors and friends. Dear Peijuan, Nan, Xiaoyang, Qu, Yongqiu, Pengling, Ying, Xin, Wenjing, and Qingsong. Thanks for walking with me when I need a favor, walking ahead me when I am lost and need guidance, and walking behind me when I become demotivated. Thanks for making my study and my life more colorful and joyful.

Finally, I owe special thanks to my family. From the beginning of my life, my parents have always been taking care of me, unconditionally loving me, supporting me, and believing in me. You always think from my side and do everything for me but never ask anything from me. Because of my parents, I never feel hopeless no matter what happens. Words are never sufficient to express my love and gratitude to my parents. During the PhD journey, I surprisingly and luckily received a special and extraordinary gift, my husband Fan. You accompany me on the life journey and let me feel the warmth of the family when I am away from my parents. You stand by me when I am depressed. You undertake all the housework when I am busy and lazy. Thank you for everything!

Xiaojie LUAN
Delft, the Netherlands
May, 2019

Contents

Chapter 1

Introduction

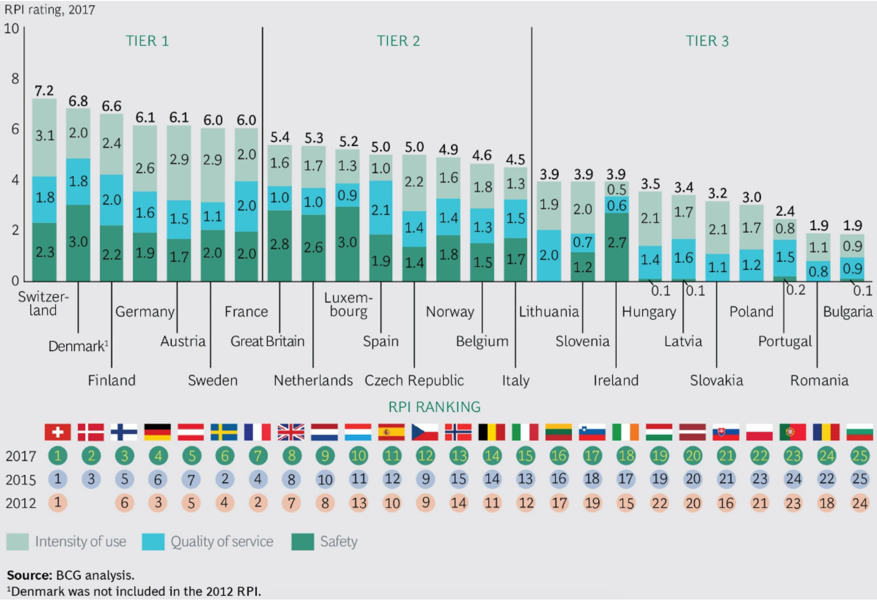
1.1 Background and motivations

Railway transport systems constitute a significant part of the transportation network and play an important role in addressing the ever-increasing mobility of people and goods. A safe, fast, punctual, reliable, and energy-efficient railway system is of crucial importance for the economic, environmental, and social objectives of a country.

Good performance of railway operations can help in attracting potential users and further increasing the share of railways. In 2017, Boston Consulting Group reported the European Railway Performance Index, as shown in Figure 1.1. This comprehensive benchmarking study of European railway operations focused on the three critical components of railway performance: intensity of use, quality of service, and safety. The result shows that Switzerland with a score of 7.2 is ranked as the best in Europe, alongside Denmark (6.8), Finland (6.6), etc. in tier 1, while a score of 5.3 for the Netherlands in tier 2. The conclusion of this study highlights the significance of safety and service quality (especially punctuality), i.e., they are the most important factors underlying changes in the performance of a railway system.

In fact, railway systems continue to face the challenge of maintaining and improving their service qualities. According to the survey of the European Commission (EC, 2018), only 59% of the rail passengers are satisfied with the railway services provided. To increase the satisfaction of passengers, improvements relevant to operations include making train services quicker, more frequent, more punctual, and more reliable for passengers. Providing such train services has been set as a goal for 2019 by the Dutch train operating company (Nederlandse Spoorwegen, NS).

A favorable market environment is the basis of providing high-quality services. Railways have developed as vertically-integrated (state-owned) organizations, which have been the most common structure for the rail sector in most countries, with responsibility for both the railway infrastructure facilities and train operations (Kurosaki, 2008). Since the 1990s, rail policy regulations in Europe (such as Directive 91/440/EC, 1991)



Source: BCG analysis.
[1]Denmark was not included in the 2012 RPI.

Figure 1.1: The 2017 European Railway Performance Index, courtesy of Boston Consulting Group (2017)

have fostered competition into the railway transport market. This has led to a vertical separation between infrastructure management and train operations. Such policies consider competition among train operating companies as a key element to achieve efficient operations. However, the common situation of a quasi-monopoly may result in discriminatory treatment among train operating companies. With the promulgation of Directive 2001/14/EC (EC, 2001), providing fair access to rail infrastructure for all competing operators is requested. Since then, the requirement of non-discrimination has been considered in both the tactical planning and the operational control process. In such a process, the infrastructure manager plays a role of making infrastructure available. With the growth of railway transport demand, the limited available capacity of infrastructure poses the most severe limitation on improving service quality and creates challenges to take non-discrimination actions.

Building or upgrading infrastructure can significantly increase the available capacity, but it costs huge amounts of money and it takes a long time. In the past years, several European countries have adopted ambitious investment plans for their railway systems. In 2016, Italy announced a ten-year program supported by planned investments of €100 billion, including €73 billion designated for infrastructure improvements. Belgium approved a €25 billion investment plan in 2013, and the project will be implemented over 12 years. Alternatives are needed, especially when large investments are impossible or where there are limited potentials of expansions. More promising solutions to make efficient use of existing infrastructure are, e.g., technological improvements, advanced planning and management, and efficient operational procedures. In addition, as advised by the European Commission (EU, 2015), potential infrastructure capacity can be exploited by better deployment and coordination. This reveals one promising direction, i.e., enhancing coordination among infrastructure users, in order to avoid unnecessary waste of capacity caused by poor coordination.

With the increase of the available capacity, typically reflected by adding additional train services, energy consumption will increase, leading to a higher cost and more carbon dioxide (CO_2) emissions. To foster sustainable development, the International Union of Railways (UIC, 2012) has set goals to reduce the CO_2 emissions and energy consumption from train operations by 50% and 30% respectively by 2030. This leads to a challenging research focus, achieving energy saving while at the same time maintaining a high quality of service. Energy-efficient train operation is seen as the most important strategy to reduce the environmental impacts and the costs used to power trains.

In line with the above setting, this research is motivated by achieving better performance of railway operations, in terms of punctuality, reliability, non-discrimination, capacity utilization, and energy efficiency. To achieve this goal, this research develops and implements optimization approaches. More specifically, the research is explored from the following five aspects:

- reducing delays and limiting their propagation by means of optimizing train orders, routes, and departure and arrival times at passing stations;
- dealing with conflicting requests of competing operators in a non-discriminatory manner by taking equity into account in the decision process;
- exploiting potentials of existing infrastructure by better coordination between the two single-problem decisions on traffic management and preventive maintenance planning;
- investigating potentials of energy efficiency in train operations by incorporating driving strategies into traffic management;
- promoting the application in practice of optimization approaches by developing distributed optimization methods.

In the remainder of this chapter, Section 1.2 describes the problem statements and discusses the challenges of the above five aspects. Section 1.3 proposes the research objectives and research questions to be solved in this dissertation. Section 1.4 summarizes the main contributions of this dissertation. Finally, Section 1.5 provides an outline of this dissertation, as well as a brief introduction of each chapter in the remainder of this dissertation.

1.2 Problem statements and challenges

The five aspects that this dissertation investigates are all in the scope of traffic management. The problem statements and the existing challenges are briefly discussed as follows, aspect-by-aspect:

- Railway traffic management

  As tactical plans, railway timetables are programmed and updated every year or every season to specify train routes, orders, and arrival and departure times at passing stations, with the common objective of maximizing the efficient use of the existing infrastructure. This is the so-called train timetabling problem or train scheduling problem (see the survey by Turner et al., 2016). In daily operations, perturbations unavoidably happen, which may affect the normal operations and cause a primary delay to the planned timetable. Due to the high interdependency between trains, the primary delay may further result in a snowball effect on other trains with consecutive delays, i.e., delay propagation. In the presence of delays, train dispatchers (controllers) are in charge of adjusting the affected schedules, with the aim of keeping the operations feasible and reducing potential negative consequences. This is the so-called real-time traffic management problem (also called traffic control, train dispatching, or train rescheduling problem), and we refer to the review paper by Cacchiani et al. (2014) and the book by Hansen and Pachl (2014) for more information. Ineffective traffic control could significantly downgrade the punctuality and reliability of train services. Therefore, in the presence of delays, how to efficiently generate effective train dispatching solutions leads to one research challenge.

- Non-discriminatory traffic control

  In daily operations, the available capacity can be reduced by delays and delay propagation, which may cause infeasibility of the planned train timetables. Train operating companies (TOCs) only look at maximizing their interests; however, their interests suffer from the negative consequences of delays, in terms of refund, penalties, and passenger dissatisfaction. In fact, the requirement of providing fair and non-discriminatory access to infrastructure for all TOCs has been mainly reflected in the timetable planning process (see Directive 2001/14/EC, 2001). This process follows a sequence of actions, i.e., applications of TOCs for infrastructure capacity, scheduling the requested applications, coordination of the conflicting requests, (if conflicts still exist, then) declaring the infrastructure congested, and employing non-discriminatory priority criteria to allocate the congested infrastructure. The rules for access and use of the infrastructure during real-time traffic management mainly focus on restoring the normal situation and do not require a special focus on non-discriminatory actions (see the review paper by Corman and Meng, 2015). The challenge that we face is then how to allocate this (reduced) capacity among competing TOCs without favoring any of them, i.e., how to provide non-discriminatory access to the limited capacity for the competing TOCs during disruptions. Additionally, in operations, penalties may be charged from TOCs for the actions that disrupt the normal operation, compensation may be granted for TOCs that seriously suffer from disruption, and TOCs may be rewarded for better than planned performance.

- Traffic control cooperating with a preventive maintenance planning

  In railway operations, infrastructure needs to be well-utilized (in terms of a train timetable) to meet passenger and goods transport demand; meanwhile, the infrastructure should be in a good shape (well-maintained by means of preventive maintenance) for ensuring that tracks are in the appropriate states for running trains. The former relates to the train scheduling problem (see the representative studies by Caprara et al., 2002; Bešinović et al., 2016), and the latter concerns a preventive maintenance time slots (PMTSs) planning problem (see the representative studies by Budai et al., 2008; Boland et al., 2013), which answers the question of in what time slots to perform the given preventive maintenance tasks, aiming at supporting railway services by preventing infrastructure failures. In practice, train schedules and PMTS plans are usually designed separately by different departments/planners. However, when generating a train schedule (or a PMTS plan), an unavoidable issue is to coordinate with PMTS plans (or train schedules), in order to ensure that the integrated plan of trains and PMTSs is conflict-free. Operating more trains leads to fewer time slots available for performing maintenance, and vice versa. The challenge is then to generate effective train schedules with joint consideration of PMTS plans, and more specifically, how to integrate the two objectives (i.e., trains and PMTS) with different properties into one single optimization problem and how to optimize them simultaneously.

- Traffic control integrating with train control

  In railway transport systems, the energy-efficiency is greatly influenced by the train operation strategy, which consists of the operating train timetables and the applied driving actions (Scheepmaker et al., 2017). The former relates to real-time traffic management, and the latter concerns train control (see the representative studies by Howlett and Pudney, 2012; Albrecht et al., 2013b), i.e., optimizing the sequence of driving regimes (maximum acceleration, cruising, coasting, and maximum braking) and the switching points between the regimes, with the aim of minimizing energy consumption. In fact, significant correlations exist between these two problems, as the traffic-related properties have an impact on the train-related properties and vice versa. Energy-efficient train operation (EETO) can be potentially achieved by jointly considering the two problems, i.e., (re-)constructing a timetable in a way that most effectively allows eco-driving (resulting in better energy performance). However, such a joint consideration leads to a very complex and difficult optimization problem, because not only the timetable should be well-defined for synchronizing the accelerating and braking actions of trains in the same block section, but also the driving actions should be controlled to reduce the tractive energy consumption under the speed limit, travel time, and distance constraints (Tuyttens et al., 2013). The research challenge is thus to integrate a rescheduling optimization problem with microscopic details with highly accurate real-time train speed trajectory optimization.

- Improvement in computation efficiency of the optimization approaches - investigation of distributed optimization approaches

  Optimization approaches often lead to large and complex optimization problems, especially when considering microscopic details or when integrating traffic management with other problems (e.g., train control). They mostly have excellent performance on small-scale cases, where optimality can be achieved in a short computation time. However, when enlarging the scale of the case, the computation time for finding a solution or for proving the optimality of a solution increases exponentially in general. Therefore, how to improve the computational efficiency of such optimization approaches for the real-time traffic management problem leads to an other research challenge.

In this section, the problem and challenge of each topic has been briefly discussed. A more detailed discussion will be presented in the corresponding chapter of each topic.

1.3 Research objectives and questions

One main research question and six sub-questions will be answered in this dissertation, in order to achieve the research objectives. The main question is

Are there benefits of incorporating equity policy, preventive maintenance planning, or train control into railway traffic management by means of optimization approaches?

Six sub-questions are given as follows:

(1) How to equitably deal with the conflicting requests of competing train operation companies while dispatching trains?

(2) How to jointly schedule trains and preventive maintenance tasks at the same time?

(3) Can the joint consideration of train scheduling and preventive maintenance planning bring any potential capacity of the existing infrastructure?

(4) How to incorporate driving actions (train control) into traffic management?

(5) Is an improvement in energy efficiency of train operations possible by means of integrating traffic management and train control?

(6) Which distributed optimization approaches can be used to reduce the computation time of the integrated problem of traffic management and train control for large railway networks?

1.4 Thesis contributions

This section describes the main contributions of this dissertation. A distinction is made between contributions that are of a scientific nature (either theoretical or methodological) and contributions that are of a societal nature.

1.4.1 Scientific contributions

The main scientific contributions of this dissertation are as follows:

1. An optimization approach for the non-discriminatory traffic management problem

   An optimization approach for the non-discriminatory traffic management problem will be developed in Chapter 3, where non-discrimination is quantified and incorporated into the traffic management problem. The optimization approach enables us to achieve an acceptable degree of equity while optimizing the train departure and arrival times, orders, and routes, and to explore the aspects related to delay equity, i.e., which controls the value of key performance indicators.

2. An integrated optimization approach for jointly considering the traffic management problem and the preventive maintenance time slots planning problem

   A formulation method to describe preventive maintenance tasks in train schedules will be proposed. With this formulation method, an integrated optimization approach will be further developed in Chapter 4, simultaneously determining train routes, orders, departure and arrival times at passing stations, as well as preventive maintenance time slots on relevant segments and stations.

3. Integrated optimization approaches for the integration of the traffic management problem and the train control problem

   An integrated modeling approach will be presented, and it incorporates the representation of microscopic traffic regulations and speed trajectories into a single optimization problem in Chapter 5. Three integrated optimization approaches for real-time traffic management, while explicitly including train control, will be developed, to deliver both a train dispatching solution (including train routes, orders, departure and arrival times at passing stations) and a train control solution (i.e., train speed trajectories). In these optimization approaches, train speed is considered variable, and the blocking time of a train on a block section dynamically depends on its real operating speed.

4. Approaches for introducing the minimization of energy consumption into the integrated optimization problem of traffic management and train control

Two approaches will be developed in Chapter 6 for including the minimization of energy consumption into the integrated optimization problems of traffic management and train control, with either nonlinear constraints or linearized constraints. These enable us to assess and optimize energy consumption and train delay of train operations simultaneously.

5. Distributed optimization approaches for the integrated optimization problem of traffic management and train control

   Three decomposition methods will be proposed to split the whole optimization problem (proposed in Chapters 5 and 6) into several subroblems. In order to deal with couplings among subproblems, three distributed optimization approaches will be introduced in Chapter 7. The approaches are proposed to improve computational efficiency of solving such optimization problems for large railway networks.

1.4.2 Societal relevance

The main contributions to society of this dissertation are as follows, from the viewpoints of passengers and operators respectively:

- From the passenger perspective,

  (1) The investigation of the non-discriminatory traffic control problem has a practical impact on providing a fair market environment to multiple competing train operation companies (TOCs), so that they can gain fair access to railway infrastructure. Such a non-discriminatory treatment can encourage TOCs to positively participate in the competition, e.g., improving passengers' satisfaction by means of providing higher-quality services, with a final purpose of increasing their ridership and raising their revenue.

  (2) The study of the traffic management problem has practical relevance with regards to providing more punctual and reliable services for passengers. This can enhance the control of passengers on personal affairs and avoid missed appointments caused by delays, while also reducing unexpected dwell time in journeys.

  (3) The exploitation of potential infrastructure capacity is practically relevant to the frequency of train services. The increase of the available capacity can bring more frequent train services to passengers, which can further lead to more options in train connections and can reduce the total travel time of passengers.

- From the operational perspective,

    (1) The approach developed for delivering non-discriminatory traffic control solutions is practically relevance to setting up market regulation to protect the rights of interests of TOCs and to guarantee the normal operation of the railway transport market.

    (2) The methods developed in this dissertation have (directly or indirectly) practical relevance in terms of improving service quality, e.g., punctuality, reliability, high frequency, flexible connections, and short travel time. Providing a high-quality service can increase the attractiveness of railways to potential users, which can further increase the share of railways and raise revenue of the railway sector.

    (3) The developed approach for improving the energy efficiency of train operations is practically relevant with regards to reducing energy consumption and CO_2 emissions of railway operations. The reduction of energy consumption can lead to lower operating costs of the railway sector. The saving of energy and the reduction of CO_2 emissions facilitate sustainability of railway operations and also contribute to sustainable development of the transportation system.

1.5 Thesis outline

This dissertation consists of 8 chapters. The outline of this dissertation is illustrated in Figure 1.2 with a clarification of the connections between the chapters. The main contents of Chapters 2-8 are briefly introduced as follows:

Chapter 2 presents the preliminaries of the following chapters and introduces the traffic management problem and reviews the state-of-the-art on the relevant topics.

Chapter 3 focuses on generating non-discriminatory train dispatching solutions (i.e., achieving an satisfactory degree of equity while dispatching). An optimization approach is proposed to explicitly consider delay equity among multiple train operation companies or trains, in addition to minimizing average (consecutive) train delay time.

Chapter 4 proposes an optimization approach to integrate the two processes of train scheduling and preventive maintenance planning, by means of a novel virtual-train-based modeling technique. A Lagrangian-relaxation-based solution framework is proposed to deal with the complicating track capacity constraints, so that the original complex optimization problem can be decomposed into a sequence of single-train-based subproblems. A standard label correcting algorithm is employed for finding the time-dependent least cost path of each train on a time-space network.

Chapter 5 addresses the integration of real-time traffic management and train control by using optimization methods, determining both traffic-related properties (i.e., a

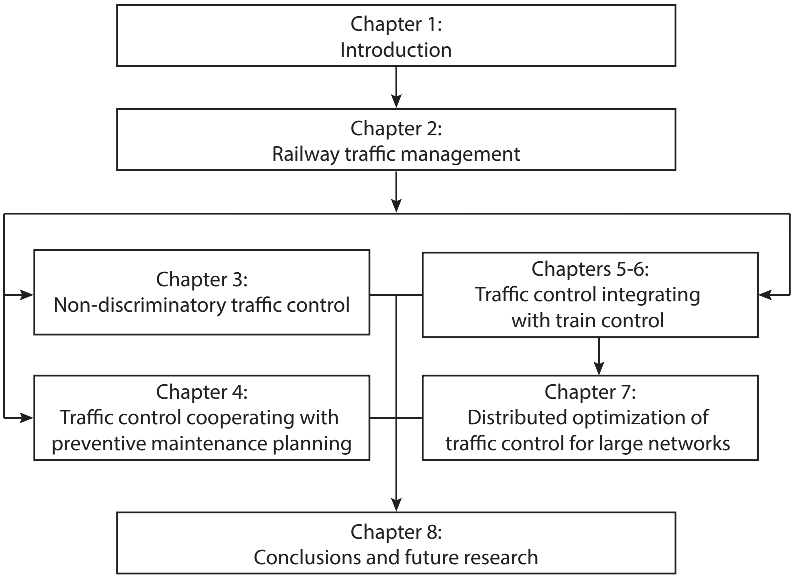

Chapter 1:
Introduction
Chapter 2:
Railway traffic management
Chapter 3:
Non-discriminatory traffic control
Chapters 5-6:
Traffic control integrating
with train control
Chapter 4:
Traffic control cooperating with
preventive maintenance planning
Chapter 7:
Distributed optimization of
traffic control for large networks
Chapter 8:
Conclusions and future research

Figure 1.2: Thesis outline

set of times, orders, and routes to be followed by trains) and train-related properties (i.e., speed trajectories) at once. A mixed-integer nonlinear programming approach (MINLP) is first proposed and solved by a two-level solution approach. This MINLP problem is then reformulated by approximating the nonlinear terms with piecewise affine functions, resulting in a mixed-integer linear programming (MILP) problem. In addition, a preprocessing method is further considered to generate the possible speed profile options for each train on each block section, one of which is further selected by a proposed MILP problem (i.e., the third optimization approach) with respect to safety, capacity, and speed consistency constraints. A custom-designed two-step solution approach is proposed to solve this MILP problem.

Subsequently, Chapter 6 focuses on the train control part of the proposed integrated optimization approaches while including energy-related formulations. A set of nonlinear constraints is proposed to calculate the energy consumption, which is further reformulated as a set of linear constraints and approximated by using piecewise constant functions. Moreover, formulations are presented to calculate the utilization of the regenerative energy obtained through braking trains.

In Chapter 7, three decomposition methods, namely a geography-based decomposition, a train-based decomposition, and a time-interval-based decomposition, are proposed to split the whole optimization problem (proposed in Chapters 5 and 6) into several subproblems. Three distributed optimization approaches are further introduced to handling the couplings among subproblems, i.e., solving subproblems sequentially and iteratively through coordination with other subproblems or with respect to the available solutions of other subproblems. The three algorithms under consideration include an alternating direction method of multipliers (ADMM) algorithm, a priority-rule-based (PR) algorithm, and a cooperative distributed robust safe but knowledgeable (CDRSBK) algorithm.

The conclusions of this dissertation and the promising directions for future work are summarized in the final Chapter 8.

Chapter 2

Railway traffic management

This chapter reviews the state-of-the-art in real-time railway traffic management, equitable capacity allocation in train timetabling and equitable control of air and road traffic, joint scheduling of trains and preventive maintenance tasks, and interaction of traffic management and train control. Then, a brief explanation on relevant terms, e.g., tracks, stations, nodes, and cells, is given, and two formulation methods for railway traffic management, namely a time-instant formulation and flag-variable-based formulation, are presented, which are the basis of the optimization problems proposed in the later chapters.

2.1 Introduction

Railway timetables are programmed and updated every year or every season to specify train routes, orders, and arrival and departure times at passing stations, with the objectives of maximizing the effective use of the existing infrastructure and of being robust to small disturbances, in order to accommodate the railway transport demand into attractive and highly safety and reliable services.

When a planned timetable is put into practice, perturbations, caused by bad weather, infrastructure failures, extra passenger flows, etc., unavoidably occur. Although timetables are designed with one objective of making operations robust and resilient to small perturbations, perturbations still often result in primary delays that affect the normal operations due to the high traffic density. The primary delays may further result in a snowball effect on other trains with consecutive delays, i.e., delay propagation, due to the high interdependency among trains. In the presence of delays, train dispatchers (controllers) are in charge of adjusting the affected schedules, aiming at keeping the operations feasible and reducing potential negative consequences.

2.2 State-of-the-art

In this section, we present a detailed literature review. We first review the literature on the real-time traffic management problem in Section 2.2.1. Then, we discuss the literature on the equitable capacity allocation (of the train timetabling problem) and on the equitable control of air traffic and road traffic in Section 2.2.2. Section 2.2.3 focuses on the joint scheduling of trains and preventive maintenance tasks in railway systems. Section 2.2.4 further reviews the literature that considers the interaction or integration of traffic and train control in some way.

2.2.1 Real-time traffic management: delay recovery

The real-time railway traffic management problem has been attracting much attention in the last years. Advances in scheduling theory make it possible to solve real-life train scheduling instances, in which not only departure/arrival times (Ginkel and Schöbel, 2007; D'Ariano et al., 2007a), but also train orders, routes, and further operational freedom are considered as variables (e.g., Törnquist and Persson, 2007; Corman et al., 2010, 2012; Meng and Zhou, 2014). For more information, we refer to the review papers by Narayanaswami and Rangaraj (2011), Corman and Meng (2015), Cacchiani et al. (2014), Fang et al. (2015), and the book by Hansen and Pachl (2014).

To formulate the railway network topology (infrastructure), traffic situation, and traffic constraints, several approaches based on operations research techniques are available in the scientific literature. A stream of studies considers the alternative graph model, which uses a combination of job shop and alternative graph techniques (D'Ariano et al., 2007a). In the alternative graph model, each block section is formulated as a single capacity server with further no-store constraints[1] and blocking constraints relating to the processing over multiple adjacent block sections (D'Ariano et al., 2007a). Some studies employ the alternative graph based formulation to deal with the problem of rerouting trains by developing meta-heuristics, e.g., a tabu search algorithm proposed by Corman et al. (2010); considering multiple classes of running traffic (Corman et al., 2011a); determining the Pareto frontier of the bi-objective problem of reducing delays and maintaining as many passenger connections as possible (Corman et al., 2012); investigating the impact of the levels of detail and the number of operational constraints on the applicability of models, in terms of solution quality and computational efficiency (Kecman et al., 2013); and rescheduling high-speed traffic based on a quasi-moving block system, which integrates the modeling of traffic management measures and the supervision of speed, braking, and headway (Xu et al., 2017).

Another stream of studies focuses on developing macroscopic models based on an

[1] The no-store constraint requires that a train, having reached the end of a track segment, cannot enter the subsequent segment if the latter is occupied by another train, thus preventing other trains from entering the former segment.

event-activity network[2], which allows for faster resolution and larger geographical scope. Schöbel (2007) proposed an event-activity based integer programming model to solve the delay management problem[3]. The model was further extended to address a discrete time/cost trade-off problem of maintaining service quality and reducing passengers' inconvenience (Ginkel and Schöbel, 2007); and by including headways and capacity constraints and testing multiple pre-processing heuristics in order to fix integer variables and to speed up the computations (Schachtebeck and Schöbel, 2010). In their proposed models, connections are decided to be maintained or dropped by minimizing the number of missed connections, while minimizing the sum of all delays of all events. Dollevoet et al. (2012) presented an event-activity based model to address the problem of rerouting passengers in the delay management process. Zhan et al. (2015) employed the event-activity network to reschedule the operations, when a segment of a high speed railway was totally blocked without considering rerouting, aiming to minimize the number of canceled and delayed trains.

Other approaches have also been proposed for solving the same problem. Rodriguez (2007) presented two constraint programming models for the rescheduling and rerouting of trains running through a junction, considering a fixed speed and a variable speed respectively. The latter does not consider proper speed variation dynamics, but it constrains train running times to be coherent with train braking and acceleration in the case of conflict. Törnquist and Persson (2007) described a mathematical model for rescheduling traffic to minimize the consequences of a single disturbance, which can be an infrastructure failure, a vehicle malfunction, or a personnel availability problem. Different strategies to reschedule trains were considered, such as a change to the track used by a train or a modified train order, in order to reduce computation time depending on the size of the instance. To improve the computational efficiency, a greedy heuristic approach was further developed by Törnquist (2012), based on the same formulation of the problem. The idea was to obtain reasonably good feasible solutions in a very short time and to use the rest of the predefined computation time to improve the obtained feasible solution by backtracking and reversing decisions made in the first stage. In Mu and Dessouky (2011), a simultaneous freight train routing and scheduling problem was formulated as a mixed-integer linear programming (MILP) model with macroscopic details, which was solved via heuristic procedures based on clustering trains according to their entrance time in the network. Meng and Zhou (2014) investigated the benefits of simultaneous train rerouting and rescheduling compared to sequential approaches in general rail networks. Network-wide cumulative flow variables were used to implicitly model capacity constraints, which enabled an easy problem decomposition mechanism. The decomposed subproblems were then solved by an adapted time-dependent least-cost algorithm. Pellegrini et al. (2014) formulated an

[2] The event-activity network is a graph, comprised by a set of nodes and directed arcs. Each node represents an arrival event or a departure event of a train, and each arc indicates a waiting, driving, or changing activity.

[3] The delay management problem determines whether trains should wait for a delayed train in order to maintain transfer connections of passengers, or should depart on time.

MILP model to tackle the real-time railway traffic management problem, representing the infrastructure with fine granularity, i.e., the route-lock route-release interlocking system and the route-lock sectional-release system. They studied the problem in the case of simple junctions and more complex areas, and used CPLEX to solve the model. In Pellegrini et al. (2015), a heuristic algorithm, named RECIFE-MILP, was developed based on an extended version of the MILP formulation proposed by Pellegrini et al. (2014). Samà et al. (2016) further investigated how to select the most promising train routes among all possible alternatives, through developing an ant colony optimization meta-heuristic. The most promising subset of train routes was included in the large and complex MILP determined by Pellegrini et al. (2014) and solved with the exact and heuristic approaches presented by Pellegrini et al. (2015).

Table 2.1 summarizes some relevant studies on the real-time traffic management problem, in terms of problem description (i.e., the level of detail, rescheduling measure), mathematical formulation (including model structure, objective, constraints, etc.), and solution algorithm. The studies of the real-time traffic management problem mostly focus on delay recovery only and neglect the equity among trains and train operating companies, the coordination with preventive maintenance, and the integration with train control. Moreover, these studies mostly have a common assumption that a fixed speed profile is used for each train, given a minimum running time and neglecting the dynamic change in speed profile as a consequence of the dispatching actions. Thus, any dynamics-related objectives, such as energy consumption, cannot be considered.

2.2.2 Equitable allocation of capacity in railway timetabling and equitable control of air traffic and road traffic

A substantial amount of studies deal with offline capacity allocation, i.e., equitable allocation of resources among competitors in the train timetabling stage. We next discuss the studies where equity is a concern while allocating capacity offline.

An auction-based allocation mechanism for railway capacity has been considered in many studies, in order to establish fair and non-discriminatory access to a railway network. In this setting, train operating companies compete for the use of a shared railway infrastructure by placing bids for trains that they intend to run. Such a mechanism is desirable from an economic point of view, because it can be argued that it leads to the most efficient use of the capacity. The main motivation and argumentation of that idea can be found in Borndorfer et al. (2006). Harrod (2013) discussed the problem of pricing the train paths for “open access” railway networks in the U.S. market. An approach based on bidding and auctioning for time slot allocation was described, in which equity is related to the possibility of handling all railway traffic in a transparent manner. As stated by the author, “Looking back at the history of the Interstate Commerce Commission in the United States, it would appear that the long and arduous investigations of cost allocation was in essence a pursuit of fairness.” Schlechte (2011) used the same

Table 2.1: Summary of the relevant studies on the real-time traffic management problem

Publications	Level of detail	Rescheduling measure	Model structure	Objective(s)	Solution algorithm
D'Ariano et al. (2007a)	micro	rT, rO	AG-based MILP	minimize the maximum secondary delay for all trains at all visited stations	B&B, H (FCFS, FLFS)
Ginkel and Schöbel (2007)	macro	rT, rO	EA-based IP	minimize the sum of train delays and the weighted sum of all missed connections	H (FSFS, FRFS, FRFS-fix, FSFS-fix)
Rodriguez (2007)	micro	rT, rO, rR	CPM	minimize the total delays of all trains	B&B
Törnquist and Persson (2007)	macro	rT	MILP	minimize the total final delays of all trains; minimize the total cost associated with delays	CS based on four different dispatching strategies
Corman et al. (2010)	micro	rT, rO, rR	AG-based MILP	minimize the maximum consecutive delays in lexicographic order	B&B, H (tabu search)
Schachtebeck and Schöbel (2010)	macro	rT, rO	EA-based IP	minimize the delays and the number of missed connections	H (FSFS, FRFS, FRFS-fix, FSFS-fix)
Corman et al. (2011a)	micro	rT, rO	AG-based MILP	minimize the total delays of all trains along other multiple objectives	B&B, H (priority rule based, FCFS)
Mu and Dessouky (2011)	macro	rT, rO, rR	MILP	minimize the total delays of all trains	GHA, NSA
Corman et al. (2012)	micro	rT, rO	AG-based MILP	minimize the train delays and the number of missed connections	B&B, H (pareto front based)
Dollevoet et al. (2012)	macro	rT, rO	EA-based IP	minimize the average delay of all passengers	CS, a modified Dijkstra's algorithm
Törnquist (2012)	macro	rT, rO	MILP	minimize the total final delays all trains	GHA
Kecman et al. (2013)	macro	rT, rO	AG-based MILP	minimize the maximum consecutive delay	B&B, H (FIFO)
Meng and Zhou (2014)	micro	rT, rO, rR	CF-based IP	minimize the total completion time of all trains	CS, LR, H (priority rule based)
Pellegrini et al. (2014, 2015)	micro	rT, rO, rR	MILP	minimize the maximum or total consecutive delays	CS, H (RECIFE-MILP)
Zhan et al. (2015)	macro	rT, rO	EA-based MILP	minimize the number of cancelled and delayed trains	CS
Samà et al. (2016)	micro	rT, rO, rR	MILP	minimize the total consecutive delays	CS, ACO meta-H
Xu et al. (2017)	micro	rT, rO	AG-based MILP	minimize the total consecutive delays; minimize the sum of the positive consecutive delays	CS

* Symbol descriptions for Table 2.1: re-time (rT); re-order (rO); re-route (rR); Alternative graph (AG); Cumulative flow (CF); Event-activity network (EA); Constraint programming model (CPM); Discrete event model (DEM); Commercial solver (CS); Heuristics (H); Branch-and-bound (B&B); Greedy heuristic algorithm (GHA); Neighborhood search algorithm (NSA); First-Leave-First-Served (FLFS); First-Come-First-Served (FCFS); First-Scheduled-First-Served (FSFS); First-Rescheduled-First-Served (FRFS); FSFS with early connection fixing (FSFS-fix); FRFS with early connection fixing (FRFS-fix); Ant colony optimization (ACO); REcherche sur la Capacité des Infrastructures FErroviaires (RECIFE, in French).

basic assumption that optimization approaches considering all stakeholders provide a more equitable allocation than an incremental or the current human assignment. The idea is that the competing train operating companies can bid for any imaginable use of the infrastructure. Possible conflicts will be resolved in favor of the party with the higher willingness to pay.

Karsu and Morton (2015) reviewed the operational research literature on inequity-averse optimization and focused on the cases where there is a trade-off between efficiency and equity. The operational research approaches that incorporate equity concerns alongside other concerns (mostly efficiency) were discussed in detail, for different problem types. Xu et al. (2014) considered the equity measure as the ratio between the maximum delay encountered by a train and the total planned time without delays. Genetic algorithms were used to solve the resulting problem for a small artificial railway line. In the urban subway traffic, Wu et al. (2015) proposed a timetable synchronization optimization model to equitably optimize passengers' waiting time over all transfer stations, with the aim of improving the worst transfer by adjusting the departure, running, and dwell times for all directions.

The approaches based on auctions and those based on scheduling are two common ways to allocate capacity with some consideration of equity. Those latter appear to be more applicable in case a solution is required in a very short computation time, as it is the case in the real-time train dispatching problem.

Some studies focus on equitable control of air traffic and road traffic. Pellegrini and Rodriguez (2013) analyzed in detail the similarities between railway and air transport modes in the critical battle for improving efficiency. The key issues are the strategic interaction of competitors for capacity allocation and the difficulty of the real-time control. Air traffic controllers are in charge of movement safety on air segments, while railway dispatchers are controlling traffic in a saturated infrastructure. For both situations, safety critical tasks are fulfilled by a safety system or mechanism. Only quality of the traffic control is at stake, and the worst consequence is a large-scale delay propagation. A stream of studies focuses on the current fairness concept in air traffic control. The FCFS (or FIFO, or Ration-by-delay) rule gives relatively equitable decisions, while better operations could be achieved if delays are to be spread equally over as many operations as possible.

For air traffic control problem, Manley and Sherry (2010) introduced a number of metrics concerning the interaction of passenger delay, fuel burn, and equity. Current regulations achieve high equity at the cost of a reduced throughput; equity and delay are in general conflicting objectives. In Vossen et al. (2003), two problems were solved in cascade, first the unconstrained problem of finding an equitable allocation, and then improving its performance with a limited deviation from the equitable allocation determined before. A follow-up work by Glover and Ball (2013) introduced stochasticity in the model, to find solutions that achieve higher levels of equity. Kuhn (2013) addressed performance and equity as once, determining efficiently the Pareto front for those two conflicting objectives. Zhong (2012) defined a bi-criteria optimization model to offload demand from a congested airspace. The Pareto frontier of efficiency and equity was generated to allow decision makers identifying the best trade-off solutions, based on a system view. Equity was considered as a set of additional side constraints. Lagrangian relaxation was further used to relax those latter constraints, yielding a decomposition in a series of single-flight scheduling problems.

Equity in air traffic operations was also considered by De Poza et al. (2009), providing definitions and metrics for equitable air traffic control, combining the geometric and the arithmetic mean of the delay of the different operations. Kim and Hansen (2013) investigated the role of sharing information in achieving equitable and collaborative resource allocation for air traffic flow control. A model was proposed by considering public and private information. Sharing such private information can achieve a clear benefit in terms of efficiency. Hoffman and Davidson (2003) pointed out that equity is a prerequisite for achieving efficient management of disturbances. Equity is achieved when the welfare of each user of the air traffic network is increased to the maximum extent possible, given limited resources, after taking proper account of individual claims and circumstances. They also used a two-stage approach, similar to Vossen et al. (2003), which first determines an equitable allocation and then increases its efficiency.

In air traffic control, there are many optimization models formulating equity from different points of view, e.g., the proportion of the delayed flights, the total delay time/cost, the delay time/cost per passenger, etc., by using different representations, e.g., variance and absolute value. However, they have similar formulations, i.e., keeping the individual values within a small range around the average value.

Furthermore, the issue of equity is also of concern to researchers in road traffic. Some authors dealt with the problem of exploring the impact of existing strategies over equity (Ahmed et al., 2008), evaluating the equity in road resources distribution (Litman, 2002), or designing transportation networks with consideration of equity (Santos et al., 2008). In the context of congestion pricing, optimal pricing models were proposed with social or spatial equity constraints (Yang and Zhang, 2002; Yin and Yang, 2004), and a modeling framework was developed to design a more equitable pricing and tradable credit schemes (Wu et al., 2012), in order to alleviate congestion or improve social benefit on multi-modal networks. Those studies are mostly from the points of view of policy, planning, and design, with an aim of suggesting better ways to incorporate fairness in transportation decisions.

We can conclude that the investigation of equitable traffic control in railway transport system is absent in the literature.

2.2.3 Joint scheduling of trains and preventive maintenance tasks

Preventive maintenance scheduling models in the railway transport field are generally presented to introduce general cost parameters in various categories, with the aim of reducing those costs. In the literature, there are a few studies on the interaction between train scheduling and preventive maintenance planning, and most of them schedule one function by minimizing its impact on the other. In this section, we review those studies that report an explicit interaction between train scheduling and preventive maintenance planning in railway systems.

Budai et al. (2008) gave an overview of the relation between planning of maintenance and production, identifying significant advantages that can be realized by taking into account the impact on production when planning maintenance. The approaches to achieve this goal can be categorized as either production planning subject to maintenance requirements; or taking into account the production impact on maintenance in maintenance planning; or taking into account resource implications (e.g., track and manpower) in maintenance scheduling. Apart from describing the main ideas, approaches and results, a number of applications were provided.

Approaches considering both trains and maintenance possessions in the same model are presented in Peng et al. (2011), Forsgren et al. (2013), and Vansteenwegen et al. (2016). In all cases, a small number of preventive maintenance time slots is introduced into an existing train timetable, allowing different types of adjustments to the trains. The impact of preventive maintenance on train schedules is explored in those papers.

Peng et al. (2011) presented a time-space network model to solve the preventive maintenance scheduling problem. The objective is to minimize the total travel costs of the maintenance teams, as well as the impact of maintenance projects on railroad operations, which were formulated by three types of side constraints: mutually exclusive, time window and precedence constraints. An iterative heuristic solution approach was proposed to solve the resulting large-scale problem, in which the scheduling problem was decomposed into subproblems that were iteratively solved by using local search on the time-space network. Forsgren et al. (2013) treated the tactical timetable revision planning case and handled a network with both single and multi-track segments. A mixed-integer linear programming approach was developed to optimize a timetable in a way that disturbs the traffic flow as little as possible. Trains can be rerouted or canceled considering different running times, depending on their stopping patterns. Vansteenwegen et al. (2016) updated a published timetable in case of the temporary unavailability of some resources, with the aim of minimizing the number of canceled trains. An algorithm was presented to solve maintenance conflicts step by step, in order to obtain a robust schedule in case of planned maintenance interventions (typically blocked tracks). The place and time of the maintenance works were considered as fixed input and only small changes were allowed to the current timetable in order to obtain a feasible and robust train service.

A meta-heuristic approach for scheduling both trains and maintenance possessions was presented by Albrecht et al. (2013c). Problem Space Search was used to generate good quality timetables, in which both train movements and scheduled track maintenance were considered. This work is an extension from the technique originally described by Pudney and Wardrop (2008), where train timetables were constructed by considering the set of trains not yet at their destination and selecting the next train movement based on data such as the earliest possible starting time. Albrecht et al. (2013c) is one of few papers that simultaneously schedule trains and maintenance tasks by a heuristic algorithm based on Problem Space Search. However, integrated optimization approaches that deal with train scheduling and preventive maintenance planning problem are absent in the existing studies.

Lidén and Joborn (2016) considered the minimization of maintenance costs and traffic limitations when dimensioning maintenance windows. However, the planned timetable are not revisable. The authors further addressed the integrated planning problem of railway traffic services and network maintenance in Lidén and Joborn (2017), by means of a mixed-integer programming approach developed based on cumulative flow variables with aggregated network and time. This is one of the few studies that make an attempt to integrate train scheduling and maintenance planning.

2.2.4 Interaction of traffic management and train control

Many studies deal with controlling the train speed, with the aim of minimizing energy consumption. In the literature, the approaches mostly identify train speed profiles using very rough approximation, at least when optimizing the sequence of driving regimes and the switching points between the regimes. A general overview of the studies can be found in the review papers by Albrecht et al. (2011); Wang et al. (2011); Yang et al. (2016), and Scheepmaker et al. (2017).

For operations according to the schedule, there is a large corpus of research available that is able to compute the regimes to be used, and to optimally follow the path of minimal energy consumption, given a running time (see e.g., Howlett and Pudney, 2012; Chevrier et al., 2013; Wang et al., 2013). Some studies focused on maximizing the regenerative energy utilization, (e.g., Rodrigo et al., 2013; Yang et al., 2014). Since little interaction with traffic management is considered in these studies, we do not elaborate on them in this review. We next focus on the studies that address the interaction and integration with traffic management in some way, e.g., in a decomposed, iterative, or non-optimized manner.

A lot of inspiration comes from metro operations, which have a particular structure of very high homogeneity (see e.g., Li and Lo, 2014a,b), basic autonomy from other systems, and limited, predicted interaction along a line. The usage of Automatic Train Operations and Communication-Based Train Control is the most common paradigm to achieve precise control of running traffic (Albrecht et al., 2011). The approach implemented in the Lötschberg tunnel system of Switzerland was described by Montigel (2009), which simulated only a limited number of trains at a time. The approach yields a very good performance, but it is limited to a well-defined small test case with a limited traffic volume. The optimal solution can be found by exhaustive search; however, the scalability and applicability of the approach to different situations (e.g., larger networks and heterogeneous traffic) still need to be assessed. The approach proposed by

Rao et al. (2016) aimed at pushing this concept further. Some heuristic extensions of the previous work (Montigel, 2009) were proposed to address the open issues on the scalability and applicability to general networks and heterogeneous traffic.

In the general case of delayed and rescheduled traffic, the most common approach for integrating these two problems is the sequential adjustment of the speed profile, based on a scheduling solution that approximates or neglects the train control problem, see e.g. D'Ariano et al. (2007b, 2008). In this line of research, Albrecht (2009), and D'Ariano and Albrecht (2010) focused on the energy minimization problem to deliver a continuous speed profile, given a schedule. Albrecht et al. (2013a) used the time windows at stations and relevant points to provide enough room for the rescheduling problem to calculate energy-efficient speed profiles of trains. The result is optimal for energy efficiency, given the solution to the scheduling part, i.e., the passing times of trains at stations and relevant points.

Another stream of approaches includes iterative approaches that feed an optimized speed trajectory back to the scheduling model to improve traffic performance. In general, those approaches offer no guarantee of optimality in either traffic management or train control. Such approaches include the method of Mazzarello and Ottaviani (2007) for the EU project Combine, which involves a double feedback loop architecture to determine both traffic-related and train-related properties by heuristics. A similar approach was later proposed by Lüthi (2009), which allowed the rescheduling of trains in real time and provided dynamic schedule information to drivers, so that they can adjust their speed in order to meet the required schedule. The positive feature of such approaches is that the feedback loops keep the deviations (i.e., train delays from the planned timetable) small. However, having the two models separated means a match between the objectives of the two models has to be found; typically, this may lead to extra delay introduced by speed management. Furthermore, stability, convergence, and system quality under a closed-loop feedback control are even more difficult to quantify than a corresponding sequential one. Quaglietta et al. (2013) and Corman and Quaglietta (2015) investigated and analyzed the outcome for what concerns stability and performance inherently introduced by closing control loops.

In a different research stream, Wang and Goverde (2016) presented a multiple-phase train trajectory optimization method under real-time traffic management, where the train trajectory is re-calculated to track the possibly adjusted timetable. This proposed method was only applied to a case with two successive trains running on a corridor with various delays. In such a case, train control interacts with traffic management by identifying train speed profiles that match the schedule of minimal delays. Wang and Goverde (2017) further proposed a multi-train trajectory optimization method to find optimal meeting locations, arrival and departure times, and speed trajectories of multiple trains within the time and speed windows. Three driving strategies, i.e., delay-recovery, energy-efficient, and on-time driving, are considered in the optimization objective selection. A case with a maximum of four trains on a single-track corridor with four stations was tested for different delay scenarios. Aiming at energy-efficient train timetabling, Wang and Goverde (2019) extended theses methods to optimize trajectories for multiple trains on a railway corridor composed of single and/or double tracks, and implemented the trajectory optimization method adjust the running time allocation of given timetables.

A radically different approach is to invert the hierarchy of the problems, i.e., first solving the problem of generating efficient speed profiles and then using only these in the traffic management part. This has been operationally translated into a choice of speed profiles from a finite set: a single speed profile in the case of Corman et al. (2009), apart from retiming actions; and multiple speed profiles in the case of Caimi et al. (2012), including retiming. Then those profiles were included in the optimization problem. Two conflicting objectives of energy efficiency and delay minimization were considered in Corman et al. (2009), in which the first objective was used as a hard constraint. Two policies were analyzed: 1) waiting in corridors, i.e., trains are allowed to wait in stations and along the line; and 2) green wave, where trains can wait only at stations. The retiming and rerouting decisions were combined through the definition of blocking stairways[4] by Caimi et al. (2012), and a optimization approach was proposed to choose a suitable blocking stairway for each train, out of the given set of alternative blocking stairways.

In Zhou et al. (2017), a unified model was developed based on a space-time-speed grid network to integrate the two problems of macroscopic train timetabling and microscopic train trajectory calculations for high-speed rail lines. Most information regarding traffic properties and train properties was pre-described in the space-time-speed grid network, and the integrated problem was then simplified as a path finding problem. A dynamic programming solution algorithm was proposed to find the train speed profile solutions with dualized train headway and power supply constraints.

In the literature, the available studies try to address their interaction and integration in a decomposed, iterative, or non-optimized manner; however, few authors deal with the integrated problem by employing mathematical optimization methods.

2.2.5 Summary of literature review

As reviewed in Sections 2.2.1 and 2.2.2, previous studies in control of train operations include negative equity approaches, which are actually discriminatory. These include all kind of priority rules that differentiate trains based on their classes, e.g., a freight train should be held at a signal to allow a faster passenger train to go first. Approaches that do not explicitly consider classes, do not lead to such discriminatory situations, including the First-Come-First-Schedule (FCFS) rule and the vast majority

[4] A blocking of a critical railway infrastructure resource, i.e., a switch or a signal, consists of the infrastructure resource and the blocking time interval during which the critical infrastructure resource is blocked. A blocking stairway is then defined by a finite sequence of blockings, and each one combines a route and a speed profile

of the optimization approaches reported in Section 2.2.2. Those approaches are not discriminatory, in the sense that they do not specify a hard ranking of train classes, but all traffic cannot have a systematically guaranteed equity, by using those approaches. In other terms, they achieve equity only in a statistical sense, i.e., averaging over all possible situations of delays and traffic, assuming they all have equal probability, and then the resulting average output will be non-discriminatory. For each and every realization and case, instead, they might actually provide discriminatory solutions that favor a particular train rather than another one. In case of systematic effects (related to the planned operations, the train services, the delays faced, and the demand), which cannot be ruled out so easily, the resulting solution will be discriminatory, as we will quantify in Chapter 3.

The train scheduling problem and the preventive maintenance planning problem are well studied separately in previous studies. As reviewed in Section 2.2.3, most existing studies on train scheduling focus on minimizing the total deviation times from an ideal timetable with pre-defined preventive maintenance plans or without considering maintenance, while studies related to preventive maintenance mostly concern minimizing total preventive maintenance costs and delays of preventive maintenance tasks. The integration of these two problems has been pointed out as a future research, in the conclusions of some review papers, e.g., Budai et al. (2008) and Hadidi et al. (2012). Only a few explicit discussions on the integration of these two problems are seen in the literature, and most of them schedule one function by minimizing its impact on another function. In the general scheduling research field, there are a very limited number of integrated models and algorithms developed for finding the optimal production schedule and maintenance plan. In the railway transport field, only a few integrated optimization models that simultaneously deal with train scheduling and preventive maintenance planning are available, e.g., Forsgren et al. (2013).

In addition, the vast majority of the optimization-based train rescheduling approaches has a common assumption that a fixed speed profile is used for each train, i.e., a predetermined (constant) minimum running time for each train is considered and train dynamics are neglected, as reviewed in Section 2.2.1. As a result, any dynamics-related objectives, such as energy consumption, cannot be directly considered in the optimization. The studies on train control mostly focus on trajectory optimization with a given running time, i.e., determining the driving regimes and the switching points, with the aim of minimizing energy consumption (see the review paper by Yang et al., 2016). As significant correlations exist between these two problems, some studies try to address their interaction and integration in a decomposed, iterative, or non-optimized manner, as discussed in Section 2.2.4. However, few authors deal with the integrated problem by employing mathematical optimization methods. When they do so, they typically either address the energy-efficient management problem for urban transit systems (e.g., Li and Lo, 2014a,b) and the high-speed railway lines (e.g., Zhou et al., 2017) with high homogeneity, classify speed into several levels and managing speed by indicating additional travel time (e.g., Xu et al., 2017), or focus on one of these two problems with

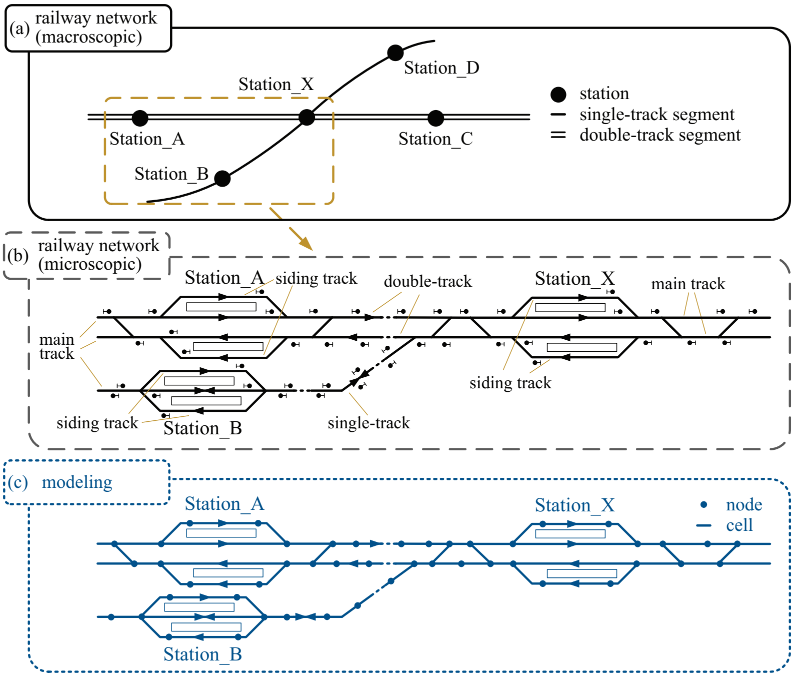


Figure 2.1: A simple railway network represented at macroscopic and microscopic levels, and modeled by nodes and cells

some simplification of the other (e.g., Caimi et al., 2012). Moreover, train operations require safety separation over block sections, in terms of time headway or space headway. The safety headway, either time headway or space headway, between two consecutive trains dynamically depends on their real speed and acceleration/deceleration rate. In real operations, we cannot assume that all traffic runs in free-flow conditions. Therefore, an integrated optimization approach with microscopic details is needed that is able to consider variable running times and safety headways, according to the train speed, accelerating or deceleration features.

2.3 Explanations of relevant terms

This section explains the terms used in this dissertation based on the description given by Pachl (2009). In Figure 2.1(a), we present a simple railway network at a macroscopic level, which consists of five stations and eight segments. Part of Figure 2.1(a) is further microscopically detailed in Figure 2.1(b). The explanations of relevant terms concerning the physical railway network are given as follows:

(i) Tracks are the roadways of a railway system. A track consists of the rails, ties, plates between rails and ties, fasteners, ballast, etc. Main tracks can be used for regular train movements, except for train stopping. Siding tracks are the tracks other than main tracks, which can be used for regular train movements, e.g., train stopping, passing, and overtaking. Figure 2.1(b) gives examples for main tracks and siding tracks.

(ii) A station is a railway facility where trains may stop for boarding and alighting of passengers or loading and unloading of goods. A station has main track(s) and siding track(s) to facilitate passing or overtaking of trains.

(iii) A segment is the track between two stations, which can be divided into block sections for the purpose of safe train separation. A block section (in a fixed block system) is a section of track, which trains cannot enter when it is blocked (reserved or occupied) by other trains. A segment may consist of one track (single-track), two tracks (double-track), or more tracks. Figure 2.1(b) presents single-track segments and double-track segments.

In order to model railway facilities, two concepts are introduced, i.e., nodes and cells, as shown in Figure 2.1(c). The concept of cell is same to that of block proposed by Brännlund et al. (1998) and Harrod (2011), and it was further adopted in many later studies, e.g., Meng and Zhou (2014).

(i) A node represents a beginning or an ending point of a block section. Additionally, it can also be viewed as a relevant point of a railway network, corresponding to a main or a siding track in station, or a point of merging or diverging of tracks.

(ii) A cell is another corresponding concept on which nodes are connected in pairs. A cell is directed from a starting node i to an ending node j, and represents a block section (on a physical network) where only one train is allowed at any time. In fact, the default value of cell capacity should be one at any given time.

2.4 Formulation methods for railway traffic management

Two formulation methods are introduced in this section, i.e., the time-instant formulation method and the flag-variable-based formulation method, which are adopted in the later chapters of this dissertation.

2.4.1 The time-instant formulation method

Given a set F of trains and a set E of cells (i.e, block sections on a physical network), we denote E_f as the set of cells that train $f \in F$ may use. A cell that connects node i and

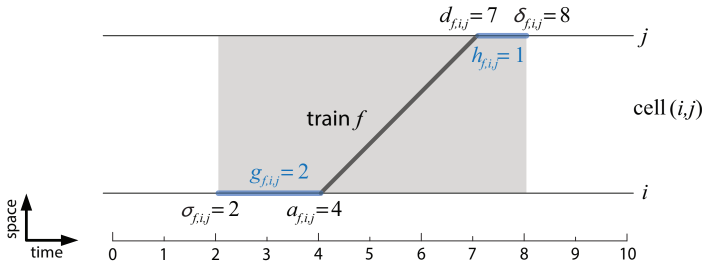


Figure 2.2: The time-instant formulation method

node j is denoted as $(i, j) \in E$. In a time-instant formulation, namely time-continuous formulation in the terminology of Cacchiani et al. (2014), for addressing the train dispatching problem, we use arrival time variables a and departure time variables d to describe train movements on block sections. These arrival and departure time variables are positive real numbers and have subscripts f, i, and j to indicate the train and the cell . More specifically, $a_{f,i,j}$ indicates the arrival time of train $f \in F$ at cell $(i, j) \in E$, and $d_{f,i,j}$ indicates the departure time of train f from cell (i, j). The arrival and departure safety headway time intervals $g_{f,i,j}$ and $h_{f,i,j}$ can be either pre-determined as parameters (e.g., Luan et al., 2017a) or considered as variables. For determining the section blocking time, the occupancy time of cell (i, j) for the arrival of train f is formulated as

$$\sigma_{f,i,j} = a_{f,i,j} - g_{f,i,j}, \quad \forall f \in F, (i, j) \in E_f, \tag{2.1}$$

and the release time of cell (i, j) for the departure of train f is formulated as

$$\delta_{f,i,j} = d_{f,i,j} + h_{f,i,j}, \quad \forall f \in F, (i, j) \in E_f, \tag{2.2}$$

where F is the set of trains, E_f is the set of cells that train f may use, and $\sigma_{f,i,j}$ and $\delta_{f,i,j}$ indicate the occupancy and release time of cell (i, j) for train f.

Figure 2.2 illustrates the movement of train f on cell (i, j) by using arrival and departure time variables. More specifically, train f arrives at time $a_{f,i,j} = 4$ and departs at time $d_{f,i,j} = 7$. As we have the safety headway times $g_{f,i,j} = 2$ and $h_{f,i,j} = 1$, cell (i, j) is blocked for train f from time $\sigma_{f,i,j} = 2$ to time $\delta_{f,i,j} = 8$.

For generating a conflict-free train dispatching solution, the cell capacity constraint is proposed by avoiding the overlap between any pair of trains on the same block section, formulated as follows:

$$\sigma_{f_2,i,j} + \left(1 - \theta_{f_1,f_2,i,j}\right) \cdot M \geq \delta_{f_1,i,j}, \quad \forall f_1 \in F, f_2 \in F, (i, j) \in E_{f_1} \cap E_{f_2}, \tag{2.3}$$

$$\sigma_{f_2,j,i} + \left(1 - \theta_{f_1,f_2,i,j}\right) \cdot M \geq \delta_{f_1,i,j}, \quad \forall f_1 \in F, f_2 \in F, (i, j) \in E_{f_1}, (j,i) \in E_{f_2}. \tag{2.4}$$

where $\theta_{f_1,f_2,i,j}$ is a binary train order variable, with $\theta_{f_1,f_2,i,j} = 1$ if train f_2 arrives at cell (i, j) or cell (j,i) after train f_1, and otherwise $\theta_{f_1,f_2,i,j} = 0$, and M is a sufficiently

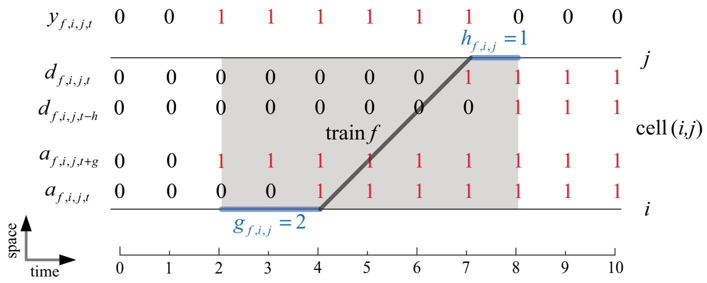

$y_{f,i,j,t}$ 0 0 1 1 1 1 1 1 0 0 0
$h_{f,i,j}=1$
j
$d_{f,i,j,t}$ 0 0 0 0 0 0 0 1 1 1 1
$d_{f,i,j,t-h}$ 0 0 0 0 0 0 0 0 1 1 1
train f
cell (i,j)
$a_{f,i,j,t+g}$ 0 0 1 1 1 1 1 1 1 1 1
$a_{f,i,j,t}$ 0 0 0 0 1 1 1 1 1 1 1
i
$g_{f,i,j}=2$
space
time
0 1 2 3 4 5 6 7 8 9 10

Figure 2.3: Illustration of the flag-variable-based formulation method

large positive number. Note that we indicate bi-directional block section on a single-track segment as (i, j) and (j,i), which refer to one physical block section in opposite direction. Thus, this formulation method can be applied to single-track, double-track, or N-track networks.

2.4.2 Flag-variable-based formulation method

The second formulation method results in a time-dependent model, and it has been applied to the train (re-)scheduling problem by Meng and Zhou (2014). In this method, flag variables are used to describe the arrival and departure of trains on block sections. These flag variables are binary and have subscripts f, i, j, and t to indicate the train, the cell (i.e., block section), and the time instant. More specifically, $a_{f,i,j,t}$ indicates whether train f has arrived at cell (i, j) by time t, and $d_{f,i,j,t}$ indicates whether train f has departed from cell (i, j) by time t. Without loss of generality, the planning horizon is discretized and denoted as integers from time index 1 to T, i.e., $t \in [1,T]$. For each train on each block section, the total number of flag variables strongly depends on the planning horizon T and the time step interval t_n, i.e., $t \in \{1, 1+t_n, 1+2t_n, ..., T\}$. At each time step t, each train has one flag variable to indicate the train arrival and another one to indicate the train departure on each block section. The safety headway time intervals $g_{f,i,j}$ and $h_{f,i,j}$ can be pre-determined as parameters (e.g., Luan et al., 2017b); however, they can also be considered as variables by using this flag-variable-based formulation method. Moreover, a set of shifted flag variables $a_{f,i,j,t+g_{f,i,j}}$ and $d_{f,i,j,t-h_{f,i,j}}$ is further used to indicate whether train f starts and ends occupying cell (i, j) at time $t-g_{f,i,j}$ and $t+h_{f,i,j}$. Therefore, instead of the cell occupancy/release variables $\sigma_{f,i,j}$ and $\delta_{f,i,j}$ used in the time-instant formulation method, binary cell blockage variable $y_{f,i,j,t}$ is used here to indicate whether train f is occupying cell (i, j) at time t, formulated as

$$y_{f,i,j,t} = a_{f,i,j,t+g_{f,i,j}} - d_{f,i,j,t-h_{f,i,j}}, \forall f \in F, (i, j) \in E_f, t \in \{1, 1+t_n, ..., T\}. \quad (2.5)$$

Figure 2.3 illustrates the same train movement as Figure 2.2 by using flag variables. As train f arrives at cell (i, j) at time 4 and departs at time 7, the flag variables of train arrival $a_{f,i,j,t}$ and train departure $d_{f,i,j,t}$ are changed from 0 to 1 at time 4 and time 7 respectively. As we assume that the safety headway time intervals $g_{f,i,j}$ and $h_{f,i,j}$ are 2 and 1 respectively, cell (i,j) is blocked for train f from time 2 to time 8, i.e., $y_{f,i,j,t} = 1$ when $t = 2,...,7$, and otherwise $y_{f,i,j,t} = 0$. The time step interval t_n could also be one second, ten seconds, or one minutes, which has great effect on both the accuracy of the final solution and the computational efficiency (as this formulation method results in a time-dependent optimization problem).

By using flag variables, the cell capacity constraint

$$\sum\nolimits_{f\in F} y_{f,i,j,t} + \sum\nolimits_{f\in F} y_{f,j,i,t} \leq 1, \quad \forall (i,j) \in E_f, t \in \{1, 1+t_n, ..., T\} \tag{2.6}$$

can be easily formulated without consideration of the train orders.

2.5 Summary

In this chapter, we have first reviewed the studies on the real-time traffic management problem, the equitable capacity allocation problem in train timetabling and equitable control of air and road traffic, the joint scheduling problem of trains and preventive maintenance tasks, and the interaction of traffic management and train control. In spite of the rich body of the existing train dispatching studies, there is a significant absence in the literature with regards to the computation of optimal train dispatching solutions with consideration of delay equity, to the integrated optimization of train scheduling and preventive maintenance planning, and to the integration of traffic management and train control. We have then introduced some relevant terms for the railway traffic management, including track, segment, station, node, and cell. Finally, two formulation methods have been briefly introduced, namely the time-instant formulation and the flag-variable-based formulation, which are the basis of the optimization approaches proposed in the later chapters.

Chapter 3

Non-discriminatory traffic control[1]

This chapter focuses on delivering non-discriminatory train dispatching solutions while multiple TOCs are competing in a rail transport market and on investigating impacting factors of the inequity of train dispatching solutions. The optimization problems adopt the time-instant formulation method introduced in Section 2.4.1.

This chapter is organized as follows. Section 3.1 gives a detailed introduction of the non-discriminatory traffic control problem. In Section 3.2, mathematical formulations are proposed, including an MILP problem (P1) that represents equity in the constraints, an MILP problem (P2) that represents equity in the objective function, and an MILP problem (P3) that ignores equity as a benchmark, and an MILP problem (P4) where only consecutive delay equity is considered. Section 3.3 presents a detailed description of the experimental settings, followed by the analysis of the experimental results, which quantify the trade-off between train delays and delay equity and the key determinants of delay equity. Finally, conclusions are given in Section 3.4.

3.1 Introduction

Railways have developed as vertically-integrated (state-owned) organizations, which have been the most common structure for the rail sector in most countries, with responsibility for both the railway infrastructure facilities and train operations (Kurosaki, 2008). Since the 1990s, rail policy regulations in Europe have fostered competition into the rail transport market. This has led to a vertical separation between infrastructure management and train operations, the progressive opening up to the market for new operating companies, and the rules regarding the allocation of slots and the pricing of infrastructure use, administered by an independent regulator (Nash and Rivera-Trujillo, 2004). Directive 91/440/EC (European Commission, 1991) is one of such policies,

[1] With minor updates, this chapter has been published in “Luan, X., Corman, F., Meng, L. (2017). Non-discriminatory train dispatching in a rail transport market with multiple competing and collaborative train operating companies. *Transportation Research Part C: Emerging Technologies*, 80, 148-174.”

which forced separation of concerns in the railway transport field, by specifying the roles of Infrastructure Manager (IM) and Train Operating Companies (TOCs). The former is in charge of making infrastructure available for both tactical train timetabling and operational train dispatching, and the latter have economic interests to strive for increasing ridership. Such policies (e.g., Directive 91/440/EC, European Commission (1991)) consider competition among TOCs as a key element to achieve efficient operations. Nevertheless, situations of quasi-monopoly are common, which may result in discriminatory treatment among different TOCs, in both the tactical train timetabling and the operational train dispatching. Similar situations exist in China, where passenger trains are generally put in a higher priority in using tracks than freight trains. This is a rather standard allocation approach, but it seriously affects the interests of freight TOCs and downgrades the efficiency of the whole system, particularly during perturbations. To protect the legitimate rights and interests of TOCs and to keep an orderly market, providing non-discriminatory access to rail infrastructure for TOCs is of great importance, in both planning and operational control levels.

The competitive interaction, concerning equity among multiple TOCs, has been studied so far mostly from a policy and financing point of view (see Borndorfer et al., 2006; Harrod, 2013). Those are offline issues addressed during design and strategic planning, including for instance the equitable allocation of timetable slots. As requested in Directive 2001/14/EC (European Commission, 2001), the access to the rail infrastructure for all TOCs should be provided in a fair and non-discriminatory manner. This requirement is reflected in the timetable planning process, which follows a sequence of applications of TOCs for infrastructure capacity, scheduling the requested applications, coordination of the conflicting requests, (if conflicts still exist, then) declaring the infrastructure congested, and employing non-discriminatory priority criteria to allocate the congested infrastructure. However, the rules for access and use of the infrastructure during real-time traffic management mainly focus on restoring the normal situation and do not require a special focus on non-discriminatory actions. Additionally, penalties may be charged for the actions that disrupt the normal operation, compensation may be granted for the TOCs that suffer from disruption, and TOCs may be rewarded for better than planned performance. During online operations, the available capacity can be reduced by delays and delay propagation, which may result in infeasibility of the planned train timetables. TOCs only look at maximizing their interests; however, their interests suffer from the negative consequences of delays, in terms of refund, penalties, and passenger dissatisfaction. The problem we have is then how to allocate this (reduced) capacity among competing TOCs without favoring any of them, i.e., how to provide non-discriminatory access to the limited capacity for the competing TOCs.

In fact, few online (i.e., in relation with real operations) approaches are known to address this problem. Most existing studies on the traffic control problem focus on minimizing the negative impacts of perturbations and pay little attention to discrimination (which corresponds to delay inequity) among competing TOCs while generating dispatching solutions (see the reviews in Section 2.2.1). This brings about the motivation of this study, i.e., delivering non-discriminatory train dispatching solutions in order to protect the rights and interests of TOCs during real-time train dispatching, and filling the research gap in the literature.

In this chapter, we focus on generating non-discriminatory train dispatching solutions (or achieving a satisfactory degree of equity while dispatching trains) and on exploring the aspects related to delay equity (i.e., what impacts on the delay equity and how to improve the equity of train dispatching). We address the train dispatching problem in a non-discriminatory way: this means that we use an optimization approach to explicitly consider delay equity among multiple competing TOCs or trains, in addition to minimizing the average (consecutive) train delay time. We consider delay equity as the degree of homogeneity of the delays faced by different trains, or trains of different TOCs. An inequitable (or discriminatory) situation occurs when some trains or some TOCs face much larger delays than other trains or TOCs.

3.2 Mathematical formulation

3.2.1 Notation

Table 3.1 provides the general subscripts, input parameters, and decision variables used in this chapter.

Table 3.1: General subscripts, sets, input parameters, and decision variables

Symbol	Description
	Subscripts and sets
V,E,F,U	sets of nodes, cells, trains, and TOCs respectively
i,j,k	node index, $i,j,k \in V$
e	cell index, generated by two adjacent nodes i and j, $e=(i,j)\in E$
f	train index, $f\in F$, $\lvert F\rvert$ is the total number of trains
u	TOC index, $u\in U$
F_u	set of trains belonging to TOC u, $F_u\subseteq F$, $\lvert F_u\rvert$ is the total number of trains belonging to TOC u
E_f	set of cells train f may use, $E_f\subseteq E$
	Input parameters and sets
$\vartheta_{f,i,j}$	free flow running time of train f to drive through cell (i,j)
ε_f	planned departure time of train f at its origin node
δ_f^{prm}	primary departure delay time[1] of train f at its origin node
σ_f	planned arrival time of train f at its destination node

continued on next page

continued from previous page

Symbol	Description
β_f	delay cost of train f (per unit time)
$w^{\text{min}}_{f,i,j}$	the minimum dwell time for train f on cell (i,j)
$g_{f,i,j}$	setup time of train f on cell (i,j) between start of cell occupancy and train arrival (before a train operation)
$h_{f,i,j}$	clearance time of train f on cell (i,j) between end of train departure and cell release (after a train operation)
o_f	origin node of train f
s_f	destination node of train f
μ_u	the maximum acceptable threshold for delay cost of TOC u
γ_u	threshold for the maximum allowed deviation between the delay of TOC u and the average delay of all the TOCs
θ_f	the maximum tolerable delay time for train f
M	a sufficiently large positive number
	Decision variables
$a_{f,i,j}$	arrival time of train f at cell (i,j)
$d_{f,i,j}$	departure time of train f from cell (i,j)
$x_{f,i,j}$	binary variable, $x_{f,i,j}=1$, if train f occupies cell (i,j) at some time, and otherwise $x_f=0$
$y_{f,f',i,j}$	binary variable, $y_{f,f',i,j}=1$, if train f' arrives at cell (i,j) after train f, and otherwise $y_{f,f',i,j}=0$
$\tau_{f,i,j}$	travel time of train f on cell (i,j)
δ^{dstn}_f	total delay time of train f at its destination
$\delta^{\text{dstnCsc}}_f$	consecutive delay time[2] of train f at its destination

[1] A primary delay is a delay resulting from an incident that directly delays a train.

[2] A consecutive delay is a delay resulting from an incident that indirectly delays a train. Consecutive delay is often seen as a consequence of primary delay.

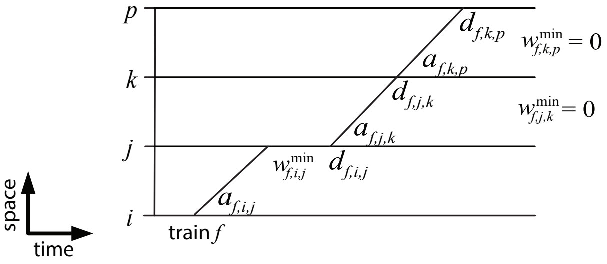


Figure 3.1: An example to illustrate the arrival, dwell, and departure time variables of a train on three consecutive cells

Four types of variables are used to formalize the rerouting and rescheduling decisions: route selection variables x, train order variables y, arrival time variables a, and departure time variables d. Specifically, $x_{f,i,j}$ captures the routing decisions in a rail network, $y_{f,f',i,j}$ describes the detailed train orders, and a pair of variables $a_{f,i,j}$ and $d_{f,i,j}$ are introduced to represent both temporal and spatial resource consumption of trains. The travel time $\tau_{f,i,j}$ is then a consequence of the interaction of all those variables for all trains in the network, as well as the train delay time, which is denoted as δ_f^{dstn}.

Figure 3.1 illustrates the arrival, dwell, and departure time variables of train f on three consecutive cells (i,j), (j,k), and (k,p). Train f is required to only stop on cell (i,j), i.e., $w_{f,i,j}^{\min}$ is set to non-zero and $w_{f,j,k}^{\min}$ and $w_{f,k,p}^{\min}$ are zero. As illustrated in Figure 3.1, the arrival time $a_{f,i,j}$ is actually the time point when train f enters cell (i,j), and the departure time $d_{f,i,j}$ indicates the time point when train f leaves cell (i,j). Moreover, the time point when train f leaves cell (i,j) should equal the time point when train f enter the consecutive cell (j,k), i.e., $d_{f,i,j}{=}a_{f,j,k}$.

3.2.2 Optimization problems

An optimization problem (P1) that represents delay equity of competitors as a set of constraints, is first presented. The objective function

$$\min \quad Z_{(\mathrm{P1})} = \frac{\sum_{f\in F}\left(\beta_f \cdot \delta_f^{\text{dstn}}\right)}{|F|} \tag{3.1}$$

minimizes the average train delay costs of all trains with respect to all operational and safety requirements. For each train, the total train delays are considered, including primary delays at the origin station and consecutive delays encountered by resolving train conflicts.

The following three constraints:

$$\sum_{j:(o_f,j)\in E_f} x_{f,o_f,j} = 1, \qquad \forall f \in F \tag{3.2}$$

$$\sum_{i:(i,j)\in E_f} x_{f,i,j} = \sum_{k:(j,k)\in E_f} x_{f,j,k}, \qquad \forall f \in F, j \in V \backslash \left\{o_f, s_f\right\} \tag{3.3}$$

$$\sum_{j:(i,s_f)\in E_f} x_f\left(i,s_f\right) = 1, \qquad \forall f \in F \tag{3.4}$$

ensure the movement of a train on the network, from its origin node, via the intermediate stops, and to its destination node respectively.

The constraint

$$a_{f,o_f,j} \geq \left(\varepsilon_f{+}\delta_f^{\text{prm}}\right) \cdot x_{f,o_f,j}, \qquad \forall f \in F, \left(o_f, j\right) \in E_f \tag{3.5}$$

guarantees that trains do not leave their origins before the earliest departure time, i.e., the sum of the planned departure time and the primary delay time.

To force the transition of a train within a cell, i.e., the train departure time from a cell is greater (later) than its arrival time at the same cell, the constraint

$$d_{f,i,j} \geq a_{f,i,j}, \qquad \forall f \in F, (i,j) \in E_f \tag{3.6}$$

is proposed.

If two adjacent cells (i,j) and (j,k) are consecutively used by train f, then we should ensure that the departure time of train f from cell (i,j) equals its arrival time at cell (j,k), i.e., $d_{f,i,j} = a_{f,j,k}$, formulated as follows:

$$d_{f,i,j} = a_{f,j,k}, \qquad \forall f \in F, j \in V \backslash \left\{o_f, s_f\right\}, (i,j) \in E_f, (j,k) \in E_f. \tag{3.7}$$

The train travel time constraint

$$\tau_{f,i,j} = d_{f,i,j} - a_{f,i,j}, \qquad \forall f \in F, (i,j) \in E_f \tag{3.8}$$

calculates the travel time of train f on cell (i,j), i.e., the sum of the actual running time and the dwell time of train f on cell (i,j), denoted as $\tau_{f,i,j}$. The train travel time here is actually the time duration for a train staying on a cell.

We use the following constraint

$$\tau_{f,i,j} \geq \left[\vartheta_{f,i,j} + w_{f,i,j}^{\min}\right] \cdot x_{f,i,j}, \qquad \forall f \in F, (i,j) \in E_f \tag{3.9}$$

to enforce the required train free-flow running time, as well as the minimum dwell times at stations. It is worth noting that the minimum dwell time is the time required to complete the processes of passengers boarding and alighting, goods loading and unloading, etc. Note that the minimum dwell times are set to zero for cells where train stops are not required.

The train order variables $y_{f,f',i,j}$ and the cell usage variables $x_{f,i,j}$ are linked by the following two constraints:

$$y_{f,f',i,j} + y_{f',f,i,j} \geq x_{f,i,j} + x_{f',i,j} - 1, \forall f, f' \in F, f \neq f', (i,j) \in E_f \cap E_{f'}, \tag{3.10}$$

$$y_{f,f',i,j} + y_{f',f,i,j} \leq 3 - x_{f,i,j} - x_{f',i,j}, \forall f, f' \in F, f \neq f', (i,j) \in E_f \cap E_{f'}. \tag{3.11}$$

Specifically, constraints (3.10)-(3.11) make sure that if and only if both trains f and f' traverse the cell (i,j), i.e., $x_{f,i,j} = x_{f',i,j} = 1$, then both the inequalities reduce to an equality: $y_{f,f',i,j} + y_{f',f,i,j} = 1$. This equality further indicates that either train f' arrives at cell (i,j) after train f or train f arrives at cell (i,j) after train f'.

For trains f and f' on cell (i,j), the train order variables $y_{f,f',i,j}$ and $y_{f',f,i,j}$ should always be smaller than the cell usage variables $x_{f,i,j}$ and $x_{f',i,j}$, which is formulated as follows:

$$y_{f,f',i,j} \leq x_{f,i,j}, \qquad \forall f \in F, f' \in F, f \neq f', (i,j) \in E_f \cap E_{f'}, \tag{3.12}$$

$$y_{f,f',i,j} \leq x_{f',i,j}, \qquad \forall f \in F, f' \in F, f \neq f', (i,j) \in E_f \cap E_{f'}. \tag{3.13}$$

With the following two constraints,

$$\begin{aligned} a_{f',i,j} - g_{f',i,j} + \left[3 - x_{f,i,j} - x_{f',i,j} - y_{f,f',i,j}\right] \cdot M \geq d_{f,i,j} + h_{f,i,j}, \\ \forall f \in F, f' \in F, f \neq f', (i,j) \in E_f \cap E_{f'} \end{aligned} \tag{3.14}$$

$$a_{f',j,i} - g_{f',j,i} + \left[3 - x_{f,i,j} - x_{f',i,j} - y_{f,f',i,j}\right] \cdot M \geq d_{f,i,j} + h_{f,i,j}, \quad \forall f \in F, f' \in F, f \neq f', (i,j) \in E_f, (j,i) \in E_{f'} \tag{3.15}$$

we ensure that any pair of trains using one cell in the same or different direction respectively are conflict-free. If two trains are running on the same cell, the successive train can only access to the cell after the cell is released for the proceeding train.

The total delay time of each train is calculated according to its planned arrival time σ_f, formulated as follows:

$$\delta_f^{\text{dstn}} = d_{f,i,s_f} - w_{f,i,s_f}^{\min} - \sigma_f, \qquad \forall f \in F, \left(i, s_f\right) \in E_f \tag{3.16}$$

Recall that the total delay time is the sum of the primary delay time at origin station and the consecutive delays encountered by solving train conflicts. The consecutive delay time is later formulated in equation (3.24).

Three parameters μ_u, γ_u, and θ_f are used in the following two constraints,

$$\frac{\sum\limits_{f \in F_u} \left(\beta_f \cdot \delta_f^{\text{dstn}}\right)}{|F_u|} \leq (1+\gamma_u) \times \frac{\sum\limits_{f \in F} \left(\beta_f \cdot \delta_f^{\text{dstn}}\right)}{|F|} + \frac{\mu_u}{|F_u|}, \qquad \forall u \in U \tag{3.17}$$

$$\delta_f^{\text{dstn}} \leq \theta_f, \qquad \forall f \in F \tag{3.18}$$

to ensure that any TOC or train cannot incur too large deviations in delays, compared with other TOCs or trains. The representations of equity in constraints (3.17) and (3.18) are inspired from the typical air traffic control (Rios and Ross, 2007; Zhong, 2012). The parameter γ_u is given as a percentage, e.g., $\gamma_u = 10\%$ means that TOC u can take 10% extra delays at most with respect to other TOCs. Parameter μ_u indicates a typical punctuality threshold, which can be used to solve the problem on how to equitably allocate a very small delay (e.g., only ten seconds) to TOCs. For instance, $\mu_u = 10$ implies that if the delay cost of TOC u is not greater than 10 seconds, TOC u is always satisfied, even if other TOCs have no delay costs. In other words, the delay equity constraints (3.17) will play a role when and only when the delay cost of TOC u is greater than 10 seconds. For the equity of trains, constraint (3.18) simply requires that the delay time of train f cannot be greater than θ_f. It should be noted that the input parameter θ_f should not be too small to avoid infeasibility of the problem.

We next propose an MILP problem (P2), where the delay equity of the competitors is represented in the objective function. Some additional parameters and variables used by the P2 problem are given in Table 3.2.

Three weights λ_a, λ_b, and λ_c are respectively used to balance the importance among the average delay costs, the equity of competing TOCs, and the equity of trains. The equity of a TOC is indicated by the deviation of the delays and measured at the aggregated level of TOCs without considering individual train delays, which can be regarded as "macroscopic" equity. Instead, the equity of a train refers to the single deviation of delay at the level of each train, which can be viewed as "microscopic" equity. The delay cost deviations of TOC u and train f, denoted as ψ_u and ϕ_f respectively, are considered as equity indicators in the P2 problem. The maximum delay cost deviation variables Ψ and Φ are used to measure the quality of the overall system from the equity point of view.

Three objectives are considered in the P2 problem, formulated as follows:

$$\min \quad Z_{(\mathrm{P2})} = \lambda_a \cdot \frac{\sum\limits_{f\in F}\left(\beta_f \cdot \delta_f^{\mathrm{dstn}}\right)}{|F|} + \lambda_b \cdot \Psi + \lambda_c \cdot \Phi \tag{3.19}$$

to minimize the train delay costs in the first term, and to reduce the delay inequity of competitors (i.e., TOCs and trains in the second and third terms respectively), while respecting all operational and safety requirements. Regarding the equity, we consider the equity of a train for a set of dispatching actions as the biggest (positive) difference between the average delay cost and the delay cost of each train. Similarly, the equity of a TOC is measured as the biggest (positive) difference between the average delay cost at the level of the given TOC and the delay cost of each TOC.

The two constraints

$$\psi_u = \frac{\sum\limits_{f\in F_u}\left(\beta_f \cdot \delta_f^{\mathrm{dstn}}\right)}{|F_u|} - \frac{\sum\limits_{f\in F}\left(\beta_f \cdot \delta_f^{\mathrm{dstn}}\right)}{|F|}, \qquad \forall u \in U \tag{3.20}$$

$$\phi_f = \beta_f \cdot \delta_f^{\mathrm{dstn}} - \frac{\sum\limits_{f\in F}\left(\beta_f \cdot \delta_f^{\mathrm{dstn}}\right)}{|F|}, \qquad \forall f \in F \tag{3.21}$$

calculate to the delay cost deviations of TOCs and trains respectively. The deviation per competitor (i.e., TOC and train) is calculated by the difference between the delay cost of each competitor and the average delay cost of all competitors.

Furthermore, the maximum delay cost deviations of TOCs and trains, denoted as Ψ and Φ, are respectively computed by

$$\Psi \geq \psi_u, \qquad \forall u \in U, \tag{3.22}$$

$$\Phi \geq \phi_f, \qquad \forall f \in F, \tag{3.23}$$

and they are then minimized in the objective function (3.19). It should be noted that a same 10 minutes deviation from the original timetable could lead to quite different

Table 3.2: Additional parameters and variables for the P2 problem

Symbol	Description
λ_a, λ_b, λ_c	weights used in the objective function, for average delay cost of trains, equity of TOCs, and equity of trains respectively
ψ_u	deviation between the delay cost of TOC u and the average delay cost of all the TOCs
Ψ	the maximum delay cost deviation of all the TOCs
ϕ_f	deviation between the delay cost of train f and the average delay cost of all the trains
Φ	the maximum delay cost deviation of all the trains

result of equity. Let us consider a case that a train arrives 10 minutes early and another case that a train arrives 10 minutes late. The two cases both result in 10 minutes deviation of the train from the original timetable, but in the latter case, the train has less equity as it faces delays. Thus, we do not consider the expected delay time in the objective function when measuring equity, and so do other optimization problems for keeping consistency.

In order to provide a benchmark, we further consider an MILP problem (P3) in which delay equity is completely neglected. The objective function of the P3 problem is formulated in (3.1), subject to constraints (3.2)-(3.16).

Moreover, the above problems consider both primary delays and consecutive delays along the routes, i.e., the total delays. As the consecutive delay is the only factor that can actually be reduced by optimized dispatching, it would be interesting to explore the impact of minimizing the consecutive delay only on the results. Therefore, we formulate the consecutive delay $\delta_f^{\mathrm{dstnCsc}}$ as follows:

$$\delta_f^{\mathrm{dstnCsc}} = d_f\left(i,s_f\right) - \sigma_f - \delta_f^{\mathrm{prm}} \ \forall f \in F, \left(i,s_f\right) \in E_f \tag{3.24}$$

Additionally, equation (3.16) should be replaced by equation (3.24), and the variable δ_f^{dstn} in the objective function (i.e., (3.1) and (3.19)) and constraints (i.e., (3.17)-(3.18) and (3.20)-(3.21)) should be changed to $\delta_f^{\mathrm{dstnCsc}}$. This results in an MILP problem (P4) that includes the objective function (3.19) and constraints (3.2)-(3.15) and (3.20)-(3.24), by considering consecutive delays only while generating an equitable train dispatching solution.

3.3 Case study

3.3.1 Setup

This section provides the description of the experimental settings based on the Dutch railway network, the generation of train primary delays at their origins, and a scheme to specify the configuration of each experiment corresponding to each subsection in Section 3.3.2.

Description of the realistic dataset based on the Dutch railway network

The realistic dataset under consideration refers to a line of the Dutch railway network, connecting Utrecht (Ut) to Den Bosch (Ht), about 50 kilometers long. The rail network is sketched in Figure 3.2; it is composed of 40 nodes and 42 links, with two main tracks, divided into one long corridor for each traffic direction and 9 stations. The two tracks in different directions are independent, so only one direction is considered, i.e., from Utrecht (Ut) to Den Bosch (Ht). Free-flow running and clear times are computed microscopically based on the typical speed profiles of trains as in Corman

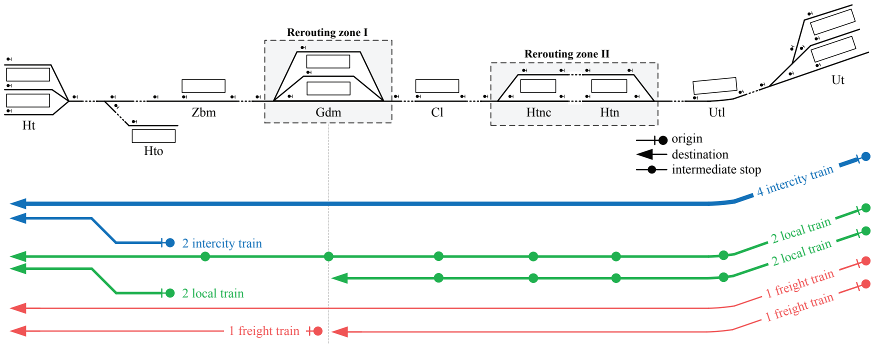


Figure 3.2: A realistic experimental network adapted from the Dutch railway network

et al. (2011b), and rounded to seconds. Three categories of trains are considered: intercity, local, and freight trains, and each of them is associated to a competing TOC: TOC_InterCity, TOC_Local, and TOC_Freight respectively.

Four global routes (i.e., from Ut to Ht, from Ut to Gdm, from Gdm to Ht, and from Hto to Ht) are determined, which merge just before Den Bosch (Ht). The routes of intercity, local, and freight trains are graphically presented in the lower part of Figure 3.2, in terms of origin, intermediate stops, destination, and the number of trains per hour. Trains can be overtaken when multiple tracks are present. We consider one hour of traffic based on a regular-interval timetable, with 15 trains. Local trains stop at all stations; intercity and freight trains stop only at the origin and destination stations. The dotted line boxes in Figure 3.2 show two rerouting zones I and II, where trains can change their usual/planned local routes (tracks).

Primary delays

Each train is given a randomly generated primary delay time δ_f^{prm} at its origin. We consider 50 delay cases of the primary delays following a 3-parameter Weibull distribution. We consider 2 options for the delay distribution: different distribution per train category and same distribution for all trains. If trains have different delay distributions, the parameters below are used:

- for intercity trains, scale=394, shape=2.27, shift=315;
- for local trains, scale=235, shape=3.00, shift=186;
- for freight trains, scale=1099, shape=2.62, shift=885.

If all trains have the same distribution, they follow the one given to the intercity trains. The values come from fitting to real life data, as explained in Corman et al. (2011b).

Experimental schemes

The experiments are denoted with a five-field code <**A,B,C,D,E**>, as described in Table 3.3.

The detailed experimental scheme is illustrated in Table 3.4 and the property/parameters investigated are highlighted. Each experiment corresponds to each subsection of Section 3.3.2.

The proposed optimization problems are all solved by CPLEX optimization studio 12.3, on a computer with an Intel(R) Core(TM) i7 @ 2.00 GHz processor and 16GB RAM. Recall that 50 delay cases of randomly generated primary delays are considered. The total number of experimental cases equals to 6250.

3.3.2 Experimental results

In this section, we analyze the experimental results with different model and parameter settings, in order to identify the key factors that influence equity, followed by a summary that provides a global view of the models' performance and related findings. Possible applications of the proposed optimization approaches in practice are further discussed in Section 3.3.3.

We report the detailed analysis of the experiments as the ***average*** result of 50 delay cases with randomly generated primary delays, in each row of Tables 3.5 to 3.10, as well as in each point value of Figures 3.6 to 3.9. Figures 3.3 to 3.5 and 3.10 give ***all*** results of those 50 delay cases, and then show the distribution of the delay cost deviation for competitors (TOCs or trains), i.e., giving the number of trains with a certain delay cost. TOCs or trains are identified by different colors in some figures, i.e., blue for intercity, green for local, and red for freight.

We describe here the contents of the table, which has the same structure as most of the subsequent tables. The **weights vector** (λ_a,λ_b,λ_c) represents the weights used in the objective function of the P2 and P4 problems, e.g., weights vector (1,2,2) means λ_a=1, λ_b=2, and λ_c=2. Each vector of weights is described by a single **Weighting ratio** κ calculated as the equitable weight (the largest one of λ_b and λ_c) divided by delay cost weight (λ_a). For example, weights vector (10,1,1) results in $\kappa=\frac{1}{10}$=0.1. An increasing weighting ratio reflects an increasing importance of equity. **Computation time** is the required time to find an optimal solution. The **average delay cost** represents the objective value of the P1 and P3 problems, and also gives the value of the first objective portion of the P2 and P4 problems. The delay cost deviation is disaggregated per train and per TOC. This is the difference between the delay cost for each competitor (TOC or train) and the average for all competitors (TOCs or trains). The maximum value of those deviations is considered as equitable objective in the P2 and P4 problems, and shown as **Max deviation of TOCs' delay cost** and **Max deviation of trains' delay cost** in the following tables. A large deviation means less equity. The deviation of

Table 3.3: Description of the five-field code

Code	Code meaning	Code symbol	Symbol description
A	model	P1	the P1 problem representing equity in the constraints
		P2	the P2 problem representing equity in the objective function
		P3	the P3 problem without consideration of equity
		P4	the P4 problem considering consecutive delays, instead of total delays
B	distribution of primary delays	*Diff*	following Weibull distributions with different parameters
		Sam	following a Weibull distribution with the same parameters
C	weights used in the objective function	–	only the average delay cost of trains is considered in the objective function, corresponding to the P1 problem and the P3 problem
		(1,2,2)	$(\lambda_a, \lambda_b, \lambda_c)$=(1,2,2), it means that the three weights λ_a, λ_b, and λ_c are set to be 1, 2, and 2 respectively, other weights vector have the similar meaning, e.g., (1,1,1), (10,1,1), and (1,5,5)
		$(\lambda_a, 0, \lambda_c)$	equity of TOCs is not considered
		$(\lambda_a, \lambda_b, 0)$	equity of trains is not considered
		$(\lambda_a, \lambda_b, \lambda_c)$	combined weights, where $\lambda_a \in \{1,2,3,5,10\}$, λ_b and $\lambda_c \in \{0,1,2,3,5,10\}$
D	rerouting option	*false*	rerouting is not considered
		true	rerouting is allowed
E	delay cost of trains per unit time	(1,1,1)	uniform delay cost per unit time, which indicates that intercity, local, and freight trains are treated with same importance
		(2,1,0.5)	variable delay cost per unit time, which implies that intercity trains a have higher delay cost (importance) and freight trains have a lower delay cost (importance)

Table 3.4: Scheme of the experiments presented in Section 3.3.2, the entries in bold are the property or parameters that we investigate.

Corresponding	Five-field code					Description of experiments
experiments	A	B	C	D	E	
Section 3.3.2(1)	**P2**	*Diff*	(2,3,3)	(1,1,1)	*false*	determining the trade-off between equity and inequity, as well as comparing the solutions obtained by the optimization problems (i.e., P2, P3, and considering two other objectives "Punctuality" and "Min_MaxDelay"[1]) and three scheduling algorithms (i.e., FIFO, FSFS, AMCC scheduling rules)[2]
	P3	*Diff*	–	(1,1,1)	*false*	
Section 3.3.2(2)	**P1**	*Diff*	–	(1,1,1)	*false*	evaluating the performance of the P1 problem and P2 problem, i.e., representing equity in the constraints and in the objective function respectively
	P2	*Diff*	(5,3,3) (2,3,3)	(1,1,1)	*false*	
Section 3.3.2(3)	P2	***Diff***	(1,1,1)	(1,1,1)	*false*	influence of different primary delays, exploring the factors that equity depend on, and estimating whether traffic suffers from heterogeneous delay
	P2	***Sam***	(1,1,1)	(1,1,1)	*false*	
	P3	***Diff***	–	(1,1,1)	*false*	
	P3	***Sam***	–	(1,1,1)	*false*	
Section 3.3.2(4)	P2	*Diff*	**$(\lambda_a,\lambda_b,\lambda_c)$**	(1,1,1)	*false*	exploring the sensitivity of solutions to the weights $(\lambda_a,\lambda_b,\lambda_c)$, $(\lambda_a,0,\lambda_c)$ and $(\lambda_a,\lambda_b,0)$, in the objective function of the P2 problem
	P2	*Diff*	**$(\lambda_a,0,\lambda_c)$**	(1,1,1)	*false*	
	P2	*Diff*	**$(\lambda_a,\lambda_b,0)$**	(1,1,1)	*false*	
Section 3.3.2(5)	P2	*Diff*	(3,2,2)	(1,1,1)	***false***	comparing the solutions with and without rerouting option
	P2	*Diff*	(3,2,2)	(1,1,1)	***true***	
Section 3.3.2(6)	P2	*Diff*	(2,3,3)	**(1,1,1)**	*false*	studying the relative importance of mixed traffic; comparing the equitable solution (i.e., all trains have the same delay cost per unit time) and the balanced solution (i.e., train delay cost per unit time differs among train classes)
	P2	*Diff*	(2,3,3)	**(2,1,0.5)**	*false*	
Section 3.3.2(7)	**P4**	Diff	$(\lambda_a,\lambda_b,\lambda_c)$	(1,1,1)	*false*	studying the solutions obtained by the P4 problem, which considers consecutive delays only

[1] "Punctuality" focuses on train punctuality to maximize the number of punctual trains, and "Min MaxDelay" considers the severity of train delay to minimize the maximum train delays.

[2] A heterogeneous situation means that trains with different speeds or different stop patterns may interfere with each other.

a competitor can be reduced by decreasing its own delay cost or increasing the delay cost of others. The average number of delayed trains is the number of delayed trains on average for the 50 delay cases. Each of the 50 instances considered has 4300 variables and more than 14000 constraints.

(1) Solutions of scheduling algorithms and optimization methods: the P2 problem vs. the P3 problem

This subsection quantifies the trade-off between equity and inequity. On top of the optimization models, we also consider three other scheduling algorithms, i.e., the well-known FIFO (First-In-First-Out) dispatching rule, common in railway practice, which simply gives priority to the train arriving first at the current block section; the FSFS (First-Scheduled-First-Served) rule that follows the train orders of the original timetable; and the AMCC (Avoid Maximum Current $C_{\max}$) rule that consists of forbidding the train orders causing the largest delay at a time, as described in D'Ariano et al. (2007a). We consider the weights vector of the P2 problem with equity to be (2,3,3) in this subsection. Moreover, while optimizing, two other objectives are also used, i.e., "Punctuality" that focuses on the train punctuality to maximize the amount of punctual trains and "Min_MaxDelay" that considers the severity of train delay to minimize the maximum train delays.

The results for different scheduling algorithms are reported in Table 3.5. In column 3 of Table 3.5, the computation time is quite small, less than four seconds for all solutions. Column 4 clearly shows that the average delay cost increases going from inequitable solutions to "Min_MaxDelay" solutions, FIFO solutions, AMCC solutions, "Punctuality" solutions, FSFS solutions, and equitable solutions. Equity also results in increasing delays, which can be viewed as a "price of equity" and quantified as (436.04 - 95.59)×15 = 5106.75 seconds for all traffic, in the case with weights vector (2,3,3).

Figure 3.3 shows the distribution of the delay cost per competitor. The X-axis represents the delay cost of trains and the Y-axis shows the number of trains. The vertical lines indicate the average delay cost for different scheduling algorithms, viewed as the reference for computing deviation (equity). When considering equity, the distribution of the deviation results in a much sharper peak, i.e., all trains have similar delay; while the other scheduling algorithms might result in smaller average values (i.e., the vertical reference line) but with a much larger deviation, which reaches as far as 522 seconds in the FSFS solutions at least and up to 3601 seconds in the "Punctuality" solutions at most, marked as orange symbols in Figure 3.3. Focus on train punctuality results in large spread of delays (less equity), as shown in Figure 3.3(d).

Figure 3.4 reports in a similar manner the distribution of the delay cost per TOC (with intercity, local, and freight trains shown in blue, green, and red respectively), for six solutions, as labelled. It is evident that the equitable model results in much less deviation for all TOCs, while the FIFO and FSFS rules result in a large spread, for all

Table 3.5: Results of different scheduling algorithms and optimization methods

	Weights vector $(\lambda_a, \lambda_b, \lambda_c)$	Computation time (unit: second)	Average delay cost (unit: second)	Max deviation of TOCs' delay cost (unit: second)	Max deviation of trains' delay cost (unit: second)	Average number of delayed trains
FIFO	–	–	118.68	134.56	482.64	11.42
FSFS	–	–	209.90	53.80	257.68	12.58
AMCC	–	–	151.33	79.14	279.89	12.60
Punctuality	–	3.28	186.21	263.91	1115.63	6.05
Min_MaxDelay	–	2.27	99.43	144.92	327.20	12.16
without equity. i.e., solutions of the P3 problem	–	0.39	95.59	112.00	365.10	9.82
with equity, i.e., solutions of the P2 problem	(2,3,3)	0.30	436.04	0.00	0.00	15.00

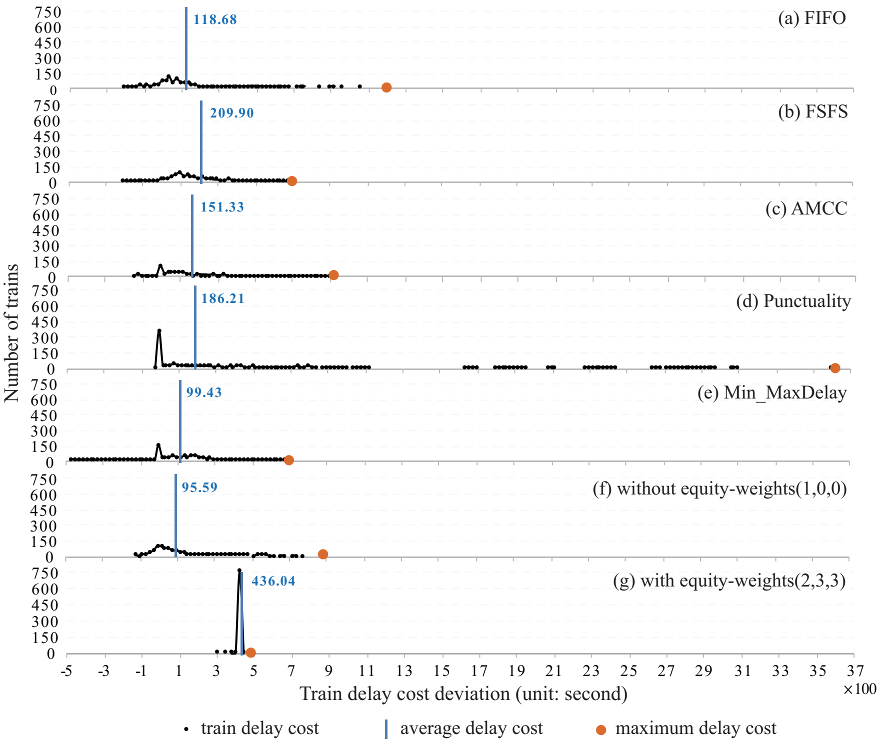


Figure 3.3: Distribution of the train delay cost

TOCs. In fact, the FSFS rule results in a slightly larger equity than the other rules, while a much better (more equitable) solution is found by the equitable model. Moreover, the P2 problem always provides the highest quality of equity, but results in the largest delay cost.

(2) Comparison of considering equity as a hard/soft constraint: the P1 problem vs. the P2 problem

We here study the impact of having equity enforced by hard constraints in the P1 problem, rather than represented as an objective in the P2 problem. A strict requirement of TOCs' equity is used as a hard constraint with the following extra parameters: the acceptable delay cost $\mu_u = 60$, and the maximum allowed deviation $\gamma_u = 1\%$ for all TOCs. Several options of the maximum tolerable delay time θ_f for each train are considered, as listed in the second column of Table 3.6. Unlike other tables, the third column of Table 3.6 shows the number of feasible solutions among the 50 delay cases.

Table 3.6 reports the main results of the P1 and P2 problems. From the viewpoint of equity, the solution quality of the P1 problem is not as good as that of the P2 problem, especially for the trains' equity in the last column. This mostly depends on the loose

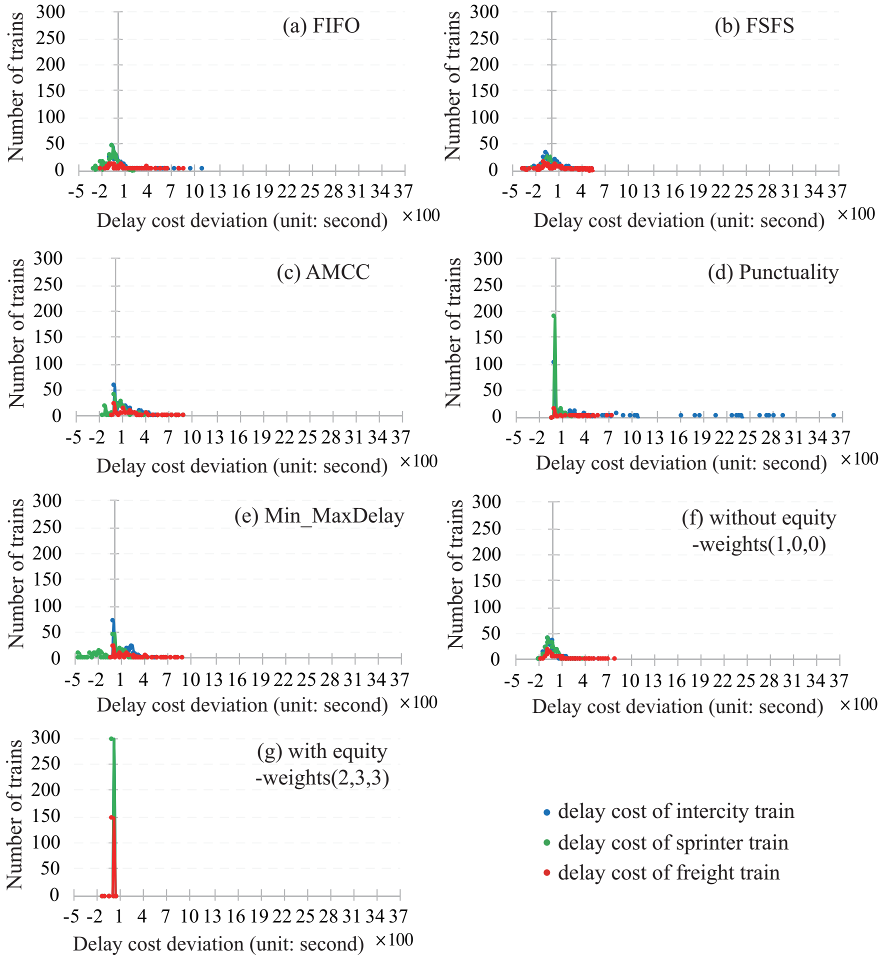
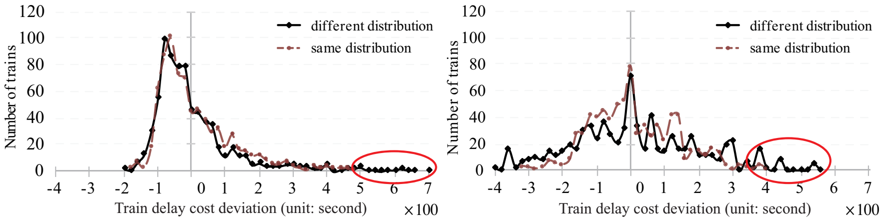


Figure 3.4: Distribution of the delay cost deviations of trains, shown separately for different TOCs

setting of the parameters. Setting a more strict value for the maximum tolerable delay time θ_f reduces this problem, but the feasibility cannot be guaranteed for all instances, as shown in the third column. As shown in column 3, by setting the maximum tolerable delay time θ_f to 600, 38 of the 50 cases are feasible; by setting $\theta_f = 300$, only 16 feasible cases can be obtained.

(3) Impact of distribution of primary delays: same distribution vs. different distribution

This subsection explores the impact of distribution of primary delays. As shown in Table 3.7 and Figure 3.5, when primary delays follow the same distribution for all com-



(a) without equity

(b) with equity-weights(1,1,1)

Figure 3.5: Distribution of the train delay cost, solutions with primary delays following the same and different distributions

petitors, a greater equity can be achieved by optimization, even if equity is neglected. Primary delays following the same distribution decrease the deviation to 36% (for the TOC's equity in the P3 problem) or 84% (for the train's equity in the P2 problem). This shows the significant effect of the primary delays on equitable operations.

Figure 3.5 reports graphically similar findings: the delay cases following different distribution result in a larger maximum delay cost deviation (i.e., less equity), as shown by the red circles. This justifies the intuition that traffic with very heterogeneous delay dynamics leads to more complex situations and makes the equity more difficult to achieve.

(4) Sensitivity of the solutions to the relative weights

Recall that we use three weights (λ_a,λ_b,λ_c) to balance the importance of the train delay cost, the delay equity of competing TOCs, and the delay equity of trains. We here analyze sensitivity of the solutions to those weights. Several combinations of weights used in this subsection are reported in Tables 3.3 and 3.4. We basically consider three cases:

1) considering both TOC's and single train's performance with the same value, which is denoted as weights(,,);

2) considering only TOC's performance and neglecting train equity, which is denoted as weights(,,0);

3) considering equity of single train and neglecting equity of TOC, which is denoted as weights(,0,).

Each vector of weights (λ_a,λ_b,λ_c) is described by a single weighting ratio κ, calculated as $\kappa = \frac{\max(\lambda_b,\lambda_c)}{\lambda_a}$. An increasing weighting ratio reflects an increasing importance of equity. The impact of the weighting ratio κ on the computation time, average delay cost, maximum deviation of TOCs' and trains' delay cost is shown in Figures 3.6 and 3.7.

Table 3.6: Comparison of solutions considering equity as a hard/soft constraint

Experimental configurations	Setting*	Percentage of feasible solutions	Computation time (unit: second)	Average delay cost (unit: second)	Max deviation of TOCs' delay cost (unit: second)	Max deviation of trains' delay cost (unit: second)
Solution of the P1 problem	θ_f=300	32%	0.11	95.35	13.94	159.63
	θ_f=600	76%	0.11	130.06	17.53	307.18
	θ_f=900	100%	0.12	181.84	18.86	411.98
	θ_f=1500	100%	0.19	181.15	19.10	547.52
	θ_f=2100	100%	0.20	181.15	18.78	592.74
Solution of the P2 problem	weights (5,3,3)	100%	0.33	199.46	0.00	237.14
	weights (2,3,3)	100%	0.30	436.04	0.00	0.00

[*] Note that the second column lists the maximum tolerable delay time θ_f for the P1 problem or the weights $(\lambda_a, \lambda_b, \lambda_c)$ for the P2 problem.

Table 3.7: Results with primary delays following same and different Weibull distributions

Experimental configurations		Weights vector $(\lambda_a, \lambda_b, \lambda_c)$	Computation time (unit: second)	Average delay cost (unit: second)	Max deviation of TOCs' delay cost (unit: second)	Max deviation of trains' delay cost (unit: second)
Different distributions of primary delays	without equity	–	0.39	95.59	112.00	365.10
	with equity	(1,1,1)	0.26	222.81	0.00	213.23
Same distributions of primary delays	without equity	–	0.25	89.18	40.76	259.40
	with equity	(1,1,1)	0.19	148.41	0.00	179.05

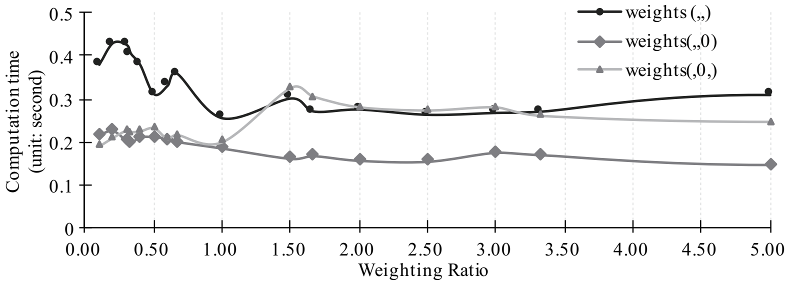
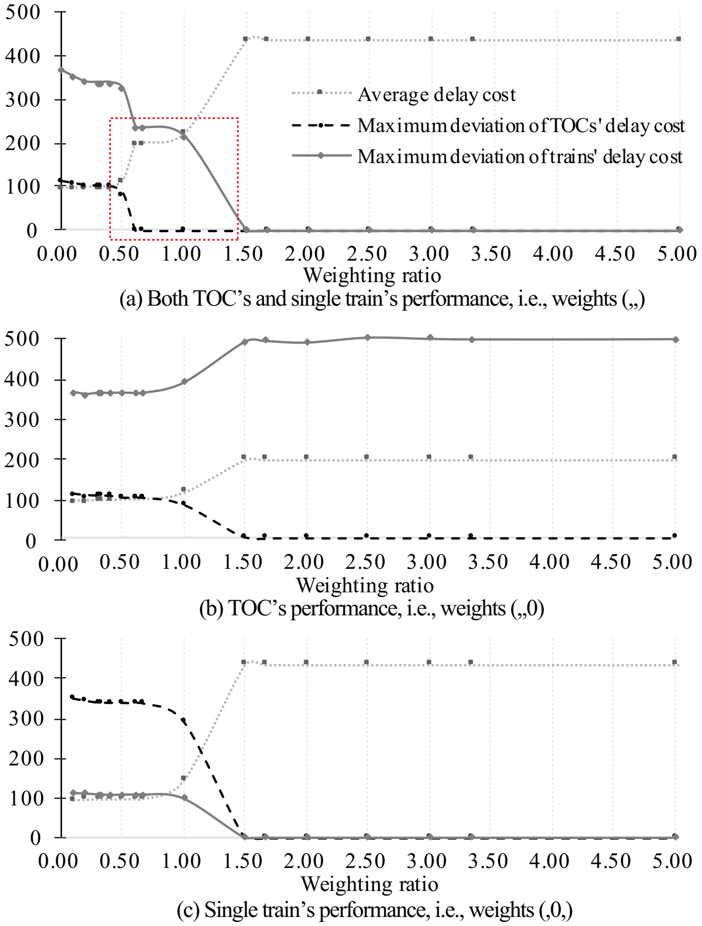


Figure 3.6: Computation time with different weight vectors



(a) Both TOC's and single train's performance, i.e., weights (,,)

(b) TOC's performance, i.e., weights (,,0)

(c) Single train's performance, i.e., weights (,0,)

Figure 3.7: Average delay cost, max delay cost deviation of TOCs and trains with different weights

As shown in Figure 3.6, the computation time is quite small (less than half second for all cases). In comparison, the cases with weights(,,) generally have longer computation time. Looking at the plots of Figure 3.7(a), the delay cost of trains increases with weighting ratio κ, and so does equity. When the weighting ratio κ is greater than 1.5, the requirements of equity can be fully satisfied (i.e., the maximum delay cost deviation is zero). Specifically, when weighting ratio κ changes from 0.50 to 0.60, a great reduction of the maximum deviation of competitors' delay cost, i.e., 27% and 100% for trains' and TOCs' equity respectively, is achieved. Moreover, as shown in the red box of Figure 3.7(a), "macroscopic" equity (TOC's equity) is much easier and faster to obtain than single train's equity. When weighting ratio $0.60 \leq \kappa < 1.50$, the maximum delay cost deviation of TOCs is zero (i.e., equity is fully achieved), and that of trains remains anyway large. Considering only TOC's equity allows to aggregate over an increasingly large set of trains, and gives more freedom in rescheduling trains.

We next discuss the equity for a single train, or considering only TOC's equity. To do so, either weight λ_b or λ_c is set to be zero. As shown in Figure 3.7(b), if only TOC's performance is considered, the maximum delay cost deviation of trains remains anyway large. In other words, there is still inequity among trains, but from a TOC point of view solutions are equitable. When single train's equity is considered, as reported in Figure 3.7(c), both the delay cost deviations of TOCs and trains approach zero, i.e., the equity of TOCs and trains is both kept well. This means that equity at microscopic level (i.e., the level of train) implies equity at macroscopic level (i.e., the aggregated level of TOC).

(5) Comparison of solutions with and without rerouting

On the Dutch railway network that we use, the infrastructure offers a few possibilities of train rerouting. For each train, a set of local rerouting options is considered that can be exploited by the optimizer. Figure 3.8 shows that a longer computation time (still compatible with real-time operations, as the computation time is always less than five seconds) is needed to find optimized rerouting measures.

In general, train delays and delay equity are conflicting objectives. But if rerouting is

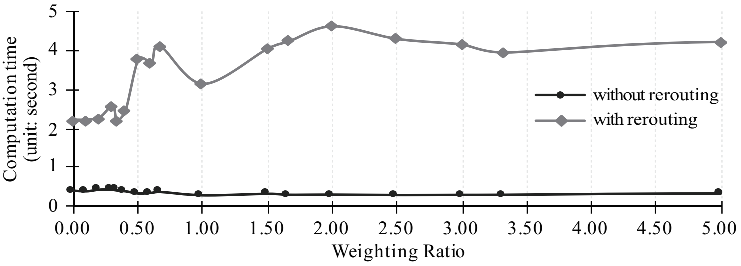


Figure 3.8: Computation time with and without rerouting

considered, the delay cost and the maximum delay cost deviations of competitors can be both reduced at the same time, by a small factor of 7% and 3% for the case with weights (3,2,2), as shown in Table 3.8. This limited improvement probably depends on the simple topology of the network (only two possible rerouting zones, see Figure 3.2).

(6) Impact of train delay cost per unit time

We now study the relative importance of mixed traffic by considering different delay costs per unit time for intercity, local, and freight trains, e.g., equal to 2, 1 and 0.5 respectively. The computation results are shown in Table 3.9. Unlike other tables, columns 3-5 in Table 3.9 show the average delay *time* (in purple) and *cost* (in orange) for each TOC, and columns 6-7 give the maximum deviation of competitors' delay *time* (in purple) and *cost* (in orange) (for TOCs and trains respectively).

Figure 3.9 reports the distribution of deviations of competitors' delay *time* and *cost* respectively (on the X-axis). The Y-axis shows the number of trains. The vertical lines (zero lines) indicate the average delay *time*/*cost*, viewed as the reference of deviation (equity). The intercity, local, and freight are represented in blue, green, and red respectively. Figure 3.9(a) and (c) show the total distribution, while Figure 3.9(b) and (d) show the distribution per TOC (in different colors). Figure 3.9(a) and (b) report the *time* deviation, while Figure 3.9(c) and (d) report the *cost* deviation.

TOC_Freight faces longer delay time, as its delay cost per unit is the lowest. This is evident in Figure 3.9(a) and (b). When considering delay cost (and not delay time), in Figure 3.9(c) and (d), the deviations of trains' delay cost are more concentrated to the Y-axis, and the delay costs of three TOCs are quite similar.

(7) Solutions considering only consecutive delays

This subsection studies the P4 problem, which considers only consecutive delays (i.e., the sum of the delays encountered by solving train conflicts). Table 3.10 shows the results with different scheduling algorithms and different weights. Figure 3.10 reports the distribution of delay cost deviations of trains, considering all delay cases and scheduling algorithms, in an analogous way to Figure 3.3.

Table 3.8: Comparison of solutions with and without rerouting, weights vector (3,2,2)

	Weights vector $(\lambda_a, \lambda_b, \lambda_c)$	Computation time (unit: second)	Average delay cost (unit: second)	Max deviation of TOCs' delay cost (unit: second)	Max deviation of trains' delay cost (unit: second)
without rerouting	(3,2,2)	0.36	200.41	0.00	235.63
with rerouting	(3,2,2)	4.11	187.79	0.00	229.39

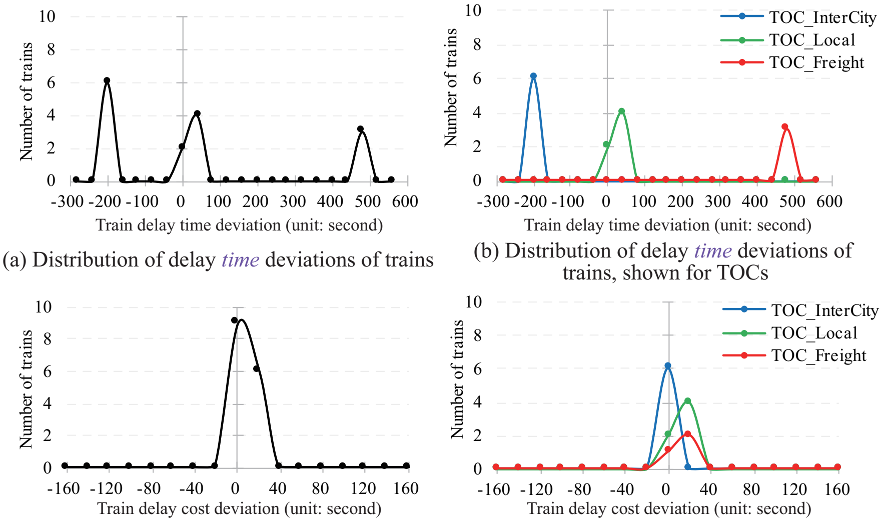
Table 3.9: Results with different delay cost per unit time, considering weights vector (2,3,3)

	Computation time (unit: second)	Average delay *time*/*cost* (unit: second)			Max deviation of TOCs' delay *time*/*cost* (unit: second)	Max deviation of trains' delay *time*/*cost* (unit: second)
		TOC_InterCity	TOC_Local	TOC_Freight		
delay cost (1,1,1)	0.30	436.04/436.04	436.04/436.04	436.04/436.04	0.00/0.00	0.00/0.00
delay cost (2,1,0.5)	0.38	231.14/462.28	462.28/462.28	924.56/462.28	385.23/0.00	462.36/0.13

Table 3.10: Results considering only consecutive delays, without rerouting options

	Weights vector $(\lambda_a, \lambda_b, \lambda_c)$	Weighting ratio κ	Computation time (unit: second)	Average delay cost (unit: second)	Max deviation of TOCs' delay cost (unit: second)	Max deviation of trains' delay cost (unit: second)
FIFO	–	–	–	33.87	60.51	309.75
FSFS	–	–	–	85.10	47.61	293.76
without equity	–	0.00	0.21	10.79	15.30	90.30
with equity	(5,1,1)	0.20	0.23	12.61	13.84	83.77
	(5,2,2)	0.40	0.21	11.72	14.79	81.74
	(1,1,1)	1.00	0.24	27.47	0.00	62.45
	(2,3,3)	1.50	0.28	89.25	0.00	1.21
	(1,3,3)	3.00	0.28	90.90	0.29	0.78
	(1,10,10)	10.00	0.29	187.79	0.49	0.49

Graphically, the solution of the P4 problem has a smaller spread compared to the other scheduling algorithms. The average delay cost approaches the value of the FSFS solution, and is larger than the other scheduling algorithms, but a much smaller deviation is found, i.e., the equity is improved greatly. What is also interesting is that this model results in better performances compared to the P2 problem , see Figure 3.3 and Table 3.5. As marked in orange symbols of Figure 3.10, the maximum delay cost deviations in this case for the FIFO, and FSFS are respectively 670 and 625 seconds; for the P3 problem the maximum delay cost deviation is 528 seconds, while for the P4 problem considering equity it is as low as 16 seconds. This phenomenon can be explained by the fact that consecutive delays are the only factors that can actually be reduced by op-



(a) Distribution of delay *time* deviations of trains

(b) Distribution of delay *time* deviations of trains, shown for TOCs

(c) Distribution of delay *cost* deviations of trains

(d) Distribution of delay *cost* deviations of trains, shown for TOCs

Figure 3.9: Distribution of delay time /cost deviations of trains and TOCs with different delay cost per unit time

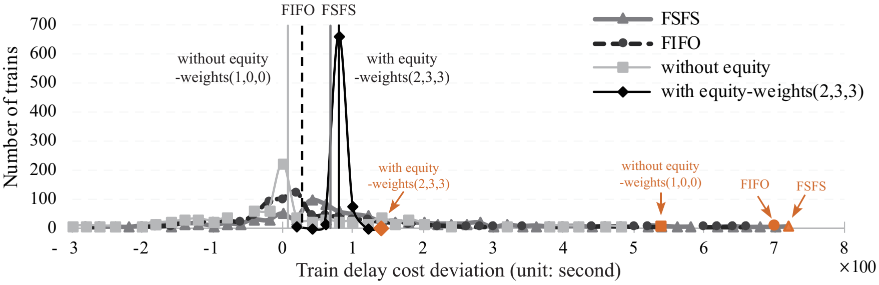


Figure 3.10: Distribution of the delay cost deviations of trains, considering only consecutive delays

timized dispatching. The P2 problem achieves uniform equitable delays for all trains, but this might mean that if a train is delayed a lot, all other traffic will be delayed by a similar amount. Focusing on the consecutive delay only, trains will face different delays (which can be seen as a form of inequity), but the delay incurred in the control process will be spread more uniformly, with overall smaller delays.

3.3.3 Discussion

(1) Summary of experimental results

We here derive the main conclusions, sketched qualitatively in Figure 3.11 from the point of view of delay cost (Y-axis) and inequity among competitors (X-axis). The original timetable (red dot) is assumed to have no delays, and it represents a reference value for what concerns equity. When delays occur, the FSFS solution (purple dot) results in high delay costs and large inequity, and so does the AMCC solution (orange dot). The FIFO solution (associated to equity in air traffic control, pink dot) is relatively bad for both equity and delays.

When using optimization approaches, the objectives of train delays and delay equity are in conflict. Traditional "inequitable" optimized dispatching problem (P3, blue dot) allows to greatly decrease delay costs at the expense of a large inequity (blue dot). Similarly, the "Min_MaxDelay" solution (light blue dot) attains a little higher delay cost and a little improved equity. The "Punctuality" solution (dark green dot), which gives priority to the punctual trains and makes the delayed trains face even more delays, results in the largest inequity. The newly proposed equitable problem (P2) determines and explores the trade-off between train delays and delay equity. This results in the black line spanning the entire plot, depending on the weights used, defining the trade-off between the two performance indicators. Equity of running traffic is improved at

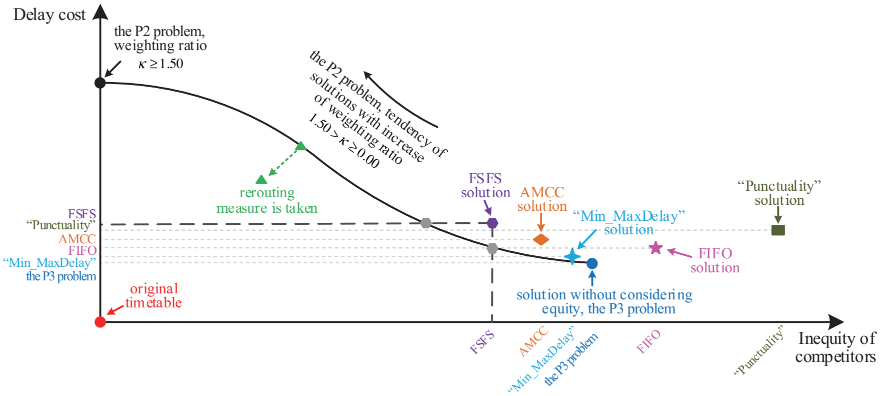


Figure 3.11: Overview of solutions obtained by different approaches and scheduling algorithms

the only expense of larger delay costs, which could be identified a "price of equity". In our cases, a weighting ratio κ larger than 1.50 leads to the full satisfaction of equity, represented as the black dot. The P2 optimization problem performs better than FSFS (purple dot) and AMCC (orange dot) for both delays and equity: given an equity target, less delays are found (the vertical dotted line); given a delay target, higher equity of operations is achieved (the horizontal dotted line). Rerouting (green dot) can improve both performance indicators at once, at the only costs of increased computational complexity.

Equity is also the result of primary delays faced by trains; similar distributions of delays (for instance, TOCs competing in the same market, trains of the same category) result in higher equity. Considering equity at the level of single train performs better than considering equity aggregated at the level of TOC. The former implies the latter in our experimental results. The choice for one or the other setting is an interesting policy issue.

Results are consistently better if equity is considered as an objective in the P2 problem and not as a constraint in the P1 problem. This latter setting depends on extra parameters (i.e., μ_u, γ_u, and θ_f) to keep solutions feasible and whose value has to be further carefully chosen and justified to the stakeholders.

If different delay costs of train categories are considered, equity is associated to limited deviation (and inequity) in costs, while delay times can still vary. The optimization approach exploits trains with lower delay costs and delays them more than other trains. The delay costs can be further adjusted to match the economic value of train punctuality of real operations for different train categories (i.e., the different punctuality targets of passenger and freight trains).

If only consecutive delays are considered, see the P4 problem, the performance concerning equity is similar to that considering total delays. Anyway, the total delay cost is smaller compared to the P2 problem. This is due to the fact that primary delays (over which limited to no control is possible) are not counted in the objective function. When considering consecutive delays only, trains will face different total delays (that can be seen as a form of inequity), but the delay incurred in the control process will be spread more uniformly, and with overall smaller delays.

(2) Possible application of the proposed optimization approaches in practice

In spite of the wish for dispatching trains in a non-discriminatory (equitable) way, only limited steps forward have been made to ensure this behavior for every single delay case and instance. According to the result summary in Section 3.3.3, we now provide some ideas and suggestions for managing railway traffic in a non-discriminatory fashion, from different viewpoints.

Equitable railway traffic management may suffer some extra delay cost (compared to traditional dispatching which neglects equity), which can be viewed as the "price of

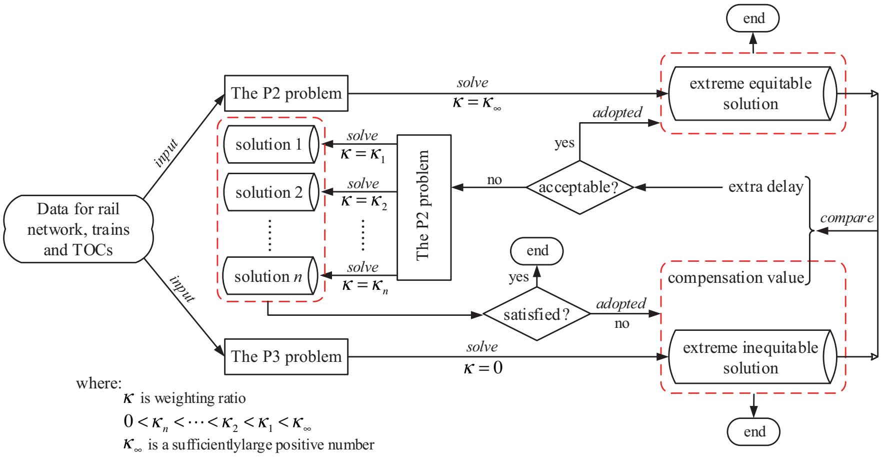


Figure 3.12: Illustration of the steps for applying the proposed optimization approaches in practice

equity”. We found that the “price of equity” depends on the network complexity, the intensity of perturbations, the starting/original timetable and the number of trains in operation. Exploring and determining an acceptable price of equity is a political/regulatory choice.

An example of how to practically deal with the gap between system performance and equity can be seen in Figure 3.12. First, the P2 problem and the P3 problem are simultaneously solved with the same dataset, in order to obtain the two extreme solutions of the equitable case and the inequitable case. If the extra delays associated with equitable dispatching are acceptable, the equitable solution would be chosen right away. If instead the extra delays are unacceptable, the equity can be incorporated to a certain extent only, by using the weighing ratio κ in the optimization; or it can be considered at the level of TOCs only, and not at the level of individual trains. If only the inequitable solution with the minimum delays is to be adopted in practice, a last resort would be to incorporate actions that offset the inequitable traffic. As stated in the Council Directive 2001/14/EC on the Allocation of Railway Infrastructure Capacity and the Levying of Charges for the Use of Railway Infrastructure and Safety Certification (European Commission, 2001), penalty and compensation may be included for actions that disrupt the operation of the network and for TOCs that suffer from disruption respectively. For instance, the TOCs with less delays may provide compensation to others that have more delays. The value of compensation can be measured by the difference of delay cost between the inequitable solution and the absolute equitable solution. This leads to a typical cooperative game theory setting of redistributing welfare.

The trade-off between system optimum (inequitable solution) and the equitable solution might be large, and sometimes even prescribe traffic to be delayed only for achieving equity. This behavior might be acceptable only under the strictest application of a non-discriminatory policy, but degrades performance of the system to a large extent. Defining a threshold for acceptable equity in terms of weighing factors can provide a balance between the two objectives, or typical multi-criteria decision making techniques can be applied. Finally, the resulting equity depends on factors that can be regulated by policy, such as capacity allocation and the original timetable. An open challenge for policy makers is how to define timetables that naturally lead to equitable solutions in practice.

3.4 Conclusions

This chapter has addressed the problem of determining real-time non-discriminatory (enforcing equity of train traffic against all possible traffic conditions and delays) train dispatching solutions, where a set of mathematical formulations and comprehensive experiments have been presented. The non-discriminatory train dispatching problem has been respectively formulated by a set of constraints in the P1 problem and bia the objectives in the P2 problem. The performance of the proposed optimization approaches for non-discriminatory traffic control has been assessed in comparison with the traditional problem (P3),where equity is neglected, the optimization problems considering the objective of train punctuality and severity of train delay respectively, and also with three dispatching rules (i.e., FIFO, FSFS, and AMCC rules), on a case study adapted from the Dutch railway network. Conclusions made from the experimental results indicate how to manage railway traffic in a non-discriminatory fashion, from the policy and practice points of view. We have studied the trade-off between equity and system performance. According to the experimental results, the optimization problem (P2) yields better performance than the FIFO, FSFS, and AMCC scheduling rules in terms of both delays and equity. The minimization of the train delays and the delay inequity are two conflicting objectives; generally, equity of running traffic is improved at the expense of larger delays. Similar distributions of the primary delays result in higher equity. Moreover, considering only the equity of TOCs allows to aggregate over an increasingly large set of trains and gives more freedom in rescheduling trains.

The future research could focus on the following main extensions. First, complex interlocking systems can be incorporated further in the optimization problems, by refining the concept of cells. This would allow to enlarge the set of routes in station areas, as well as including more processes at stations, like turn-around or shunting. A second direction is to study how to best structure the original timetable, with the objective of ensuring equitable traffic control in operations. This would describe the impact of the timetable beyond robustness and resilience against small delays in operations (see Bešinović et al., 2016). The trade-off between equity and heterogeneity of the timetable should also be explored. Finally, a comprehensive framework can be defined where equitable planning (capacity allocation) and equitable control can be considered at once, to reach non-discriminatory operations at system level.

Chapter 4

Integration of traffic control and preventive maintenance planning[1]

This chapter addresses the problem of simultaneously scheduling trains and planning preventive maintenance time slots (PMTSs) on a general railway network. The optimization problem in this chapter is developed based on the flag variables introduced in Section 2.4.1.

This chapter is organized as follows. Section 4.1 gives a detailed introduction of the integrated problem of train scheduling and PMTS planning. Section 4.2 presents a conceptual illustration for interpreting the integration of train scheduling and PMTS planning. A virtual-train-based formulation technique is introduced in Section 4.3, followed by an integrated optimization approach for scheduling trains and planning PMTSs. In Section 4.4, a Lagrangian-relaxation-based solution framework is proposed. Section 4.5 systematically examines the effectiveness and computational efficiency of the proposed optimization approach and algorithms. Conclusions are given in Section 4.6.

4.1 Introduction

Railway transport plays a crucial role in addressing the ever-growing needs for mobility of population and goods. In order to fulfill the growing demand and achieve higher competitiveness in a multimodal transport market, the infrastructure needs to be well-utilized (in terms of a train timetable) to meet passenger and goods transport demand. Meanwhile, railway infrastructure should be in a good condition (i.e., well-maintained by means of preventive maintenance (PM)) for ensuring that tracks are in the appropriate states for running trains. However, performing PM tasks in a time slot

[1] With minor updates, this chapter has been published in "Luan, X., Miao, J., Meng, L., Corman, F., Lodewijks, G. (2017). Integrated optimization on train scheduling and preventive maintenance time slots planning. *Transportation Research Part C: Emerging Technologies*, 80, 329-359."

normally needs a possession of tracks, which implies a complete capacity breakdown of the tracks; as a result, no train is allowed to run on them during the possession. Thus, an effective train schedule with joint consideration of PMTS plans is typically desired, especially for the bottleneck area(s) of a railway network during peak hours.

Train schedules are tactical plans that specify for each train a physical network route and arrival and departure time at passing stations. PMTS plans define work space and work time possession for each PM task. The former aim at delivering railway services to customers, and the latter have the role of supporting railway services by preventing infrastructure failures. In practice, train schedules and PMTS plans are usually designed separately by different departments and planners. However, the interaction between those two is critical, as they take possession of infrastructure (utilizing capacity) competitively. Operating more trains leads to less time slots available for performing maintenance, and vice versa. The tension is especially high when infrastructure capacity is inadequate, which is the case in many bottleneck areas. When generating a train schedule (or a PMTS plan), an unavoidable issue is to coordinate with PMTS plans (or train schedules), by means of simultaneously considering train scheduling and PMTS planning. Inappropriate coordination would result in inefficient use of capacity and even conflicts between those two. Moreover, situations of interchange stations on a railway network would be even more complex. The capacity of an interchange station might be underutilized, due to unsynchronized occupancy of PMTSs for different lines. It is hardly possible to find a timetable with efficiently utilized capacity for trains and maintenance tasks, if the two tasks (i.e., schedule trains and PMTSs) would not be simultaneously considered.

In this chapter, we integrate the train scheduling and PMTS planning processes by means of an optimization approach. With the given demand of trains and PM tasks, we simultaneously optimize the routes, orders, and departure and arrival times of trains at passing stations, as well as the work time of PM tasks (i.e., PMTSs). By applying a flag-variable-based formulation method (introduced in Section 2.4.2), a novel integrated mixed-integer linear programming (MILP) approach is proposed, to deliver a globally optimal or satisfactory schedule for both trains and PMTSs with microscopic feasibility details. This means that PMTSs are also scheduled and no longer pre-determined in the train scheduling process; they are positioned in time so as to have the best impact. To achieve this integration, a modeling technique is especially presented that naturally provides an easy formulation method to describe PMTSs as virtual trains. Complex track capacity is formulated by side constraints and further dualized in a Lagrangian-relaxation-based solution framework, where the original complex integrated problem of train scheduling and PMTS planning is decomposed into several single-train-based subproblems. For each subproblem, a standard label-correcting algorithm is employed for finding the time-dependent least-cost path on a time-space network. The resulting dual solutions can be transformed to feasible solutions by adopting priority rules. Numerical experiments are conducted on a small artificial network and a real-world network adapted from a Chinese railway network, to evaluate the performance (in terms of effectiveness and computational efficiency) of the integrated optimization approach and the Lagrangian-relaxation-based solution framework. The benefits of simultaneously scheduling trains and planning PMTSs are also demonstrated, compared with a commonly used *sequential scheduling method*, which will be described at the end of Section 4.2.1.

4.2 Conceptual illustration

In this section, integrated scheduling and sequential scheduling of trains and PMTSs are conceptually illustrated, followed by a problem statement and notations.

4.2.1 Integrated and sequential scheduling of trains and PMTSs

Recall that the train scheduling problem aims at determining routes, orders, and departure and arrival times for a set of trains such that the resulting train schedule does not violate capacity and satisfies operational safety. The PMTSs planning problem considered in this research involves the allocation of time slots for PM tasks in a timetable, i.e., determine the work time for a set of PM tasks. Note that other maintenance-related problems, e.g., how often should we perform maintenance to match the deterioration of infrastructures, are out of the scope of this research.

These two problems are not independent, and the choices taken to solve one of them heavily influence the other. Less coordination between these two problems would lead to negative consequences and further affect railway services. The following negative consequences are seen in practical planning processes, when the train scheduling and PMTSs planning are separated and not harmonized:

(1) *Conflicts*: situations where infrastructure resources are requested by a train and a PMTS in overlapping time periods. Conflicts often occur between train schedules and PM plans, if there is no interaction between those two scheduling processes. Such conflicts would result in a potential safety risk and affect safety of passengers and maintenance workers. This risk forces the planners to accept a sub-optimal schedule for each individual problem.

(2) *Underutilized capacity*: situations where potential capacity could be exploited for operating more trains. Capacity is potentially available, but practically impossible to be used somewhere else in time or space. This is more critical at interchange stations where two or more railway lines merge. As illustrated in the upper portion of Figure 4.1, Line_1 and Line_2 merge at Station_X. One PMTS is required for each line with a duration of three hours. Figure 4.1(a) presents a schedule of trains and PMTSs for instance, in which the PMTSs' starting times of the two lines are at 3 and 6 respectively. This results in a 6-hour unavailable period for Station_X.

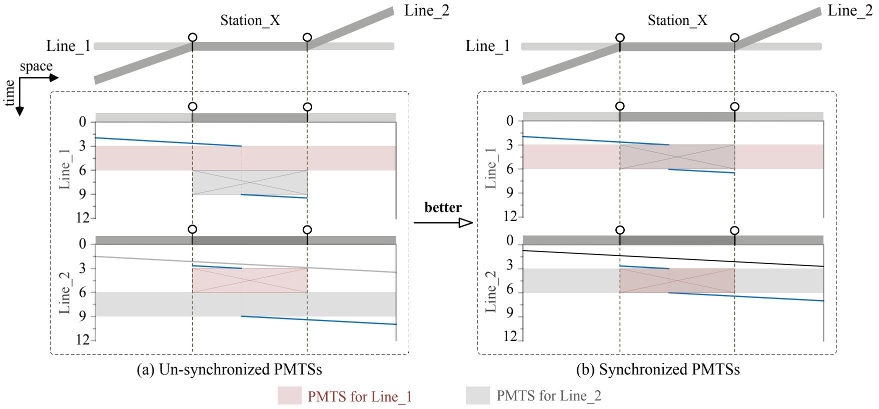


Figure 4.1: Un-synchronized and synchronized PMTSs at an interchange station

In such a case, a solution with synchronized PMTSs is preferred, as shown Figure 4.1(b). Such a synchronization can improve the utilization of capacity, so that more time slots are available for operating trains and further providing more train services to customers. Such an underutilization of capacity can be seen in practice, where it results from the uncoordinated planning. However, this does not imply that all PMTSs at an interchange station should be performed in overlapping time periods; the solution varies from case to case and should be optimized from a global perspective.

(3) *Inefficiency of the overall system*. The global optimal schedule for some objectives, like minimizing the total train travel times, is far from easy to achieve, if trains and PMTSs are sequentially optimized. A schedule with bad quality would further affect the efficiency of a railway system.

(4) *Dissatisfaction of passengers*. Due to a lack of coordination, one train may stop at a station to wait for the end of maintenance works. An increased train travel time may result in an uncomfortable experience for passengers and further affect passengers' satisfaction.

(5) *Reduced benefits of train operating companies*. Maintenance aims at keeping the railway production system into a good condition to perform its function. More capacity used by maintenance implies less capacity available for operating trains. Inefficient use of capacity impacts the number of trains operated, which would further reduce the benefits of train operating companies.

For safety reasons, conflicts must be eliminated completely. To deal with this, a sequential scheduling method is usually adopted, where PMTSs are pre-scheduled and considered as input for scheduling trains (e.g., Caprara et al., 2006). Figure 4.2(a) illustrates the sequential scheduling method with two steps: 1) determine PMTS(s); 2)

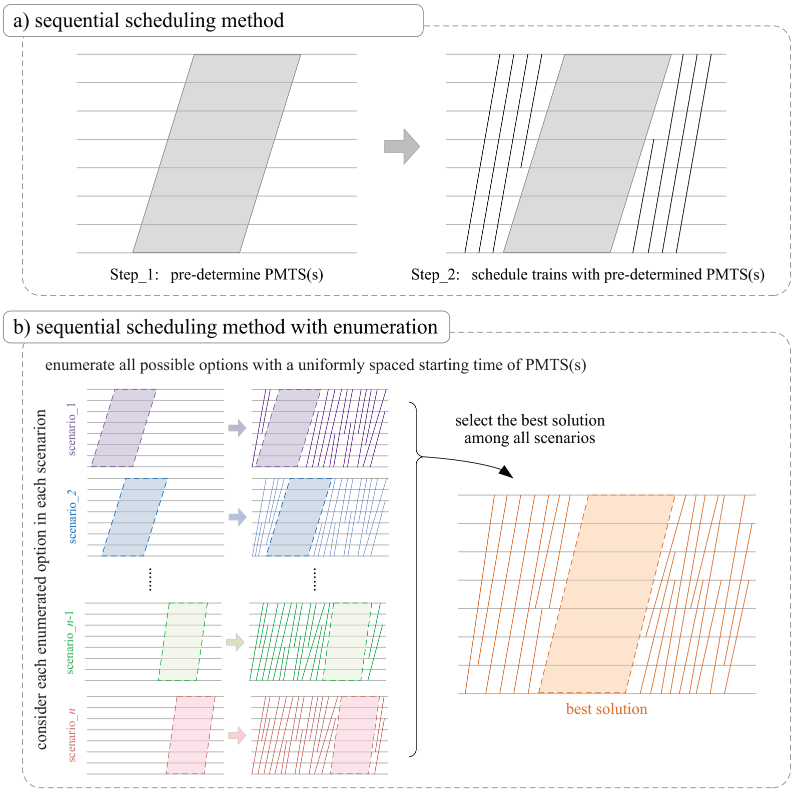

a) sequential scheduling method
Step_1: pre-determine PMTS(s)
Step_2: schedule trains with pre-determined PMTS(s)
b) sequential scheduling method with enumeration
enumerate all possible options with a uniformly spaced starting time of PMTS(s)
consider each enumerated option in each scenarion
scenario_1
scenario_2
scenario_n-1
scenario_n
select the best solution among all scenarios
best solution

Figure 4.2: Illustration on the sequential scheduling method

schedule trains given the pre-determined PMTS(s). One possible drawback associated with such a sequential scheduling method is that the limited options given in the first PMTSs planning stage could dramatically downgrade the performance of the second train scheduling solution. The available capacity for trains might be reduced by inappropriate PMTS schedules in the first stage, so that the planned trains might not be able to be scheduled completely and efficiently.

In reality, the pre-scheduled PMTSs may be adjusted according to the feedbacks of scheduling trains. However, no inspiration is found in the literature to achieve such a feedback loop, i.e., how to update the pre-determined PMTSs in the first stage (an interesting future research topic would be to determine smart ways to do that, e.g., based on meta-heuristics). For obtaining a sequential solution with better quality, an enumerative process can be applied, as illustrated in Figure 4.2(b), rather than randomly pre-determining the PMTSs (as shown in Figure 4.2(a)). Several scenarios are generated by enumerating all possible options with a uniformly spaced starting time of PMTSs. In each scenario, each option of PMTS starting times is considered, and we assume that the PMTSs start at the same time on all relevant block sections. A new train schedule is then generated according to the considered option of PMTSs in each scenario. However, the global optimal schedule of trains and PMTSs is not easy to find, unless all possible options of PMTS plans are enumerated. In fact, it is quite difficult (sometimes even impossible) to identify and explore all possible options of PMTSs, especially on a large-scale rail network over a long-term planning horizon.

To implement the sequential scheduling method, we refer to the optimization problem and algorithm proposed by Meng and Zhou (2014), which is able to simultaneously retime and reroute trains for generating an optimal train schedule. The cell capacity in the corresponding time periods of the pre-determined PMTSs is set to zero for representing the track possessions of PM tasks.

Such a sequential scheduling method is able to avoid conflicts (it generates a feasible schedule of trains and PMTSs) and marginally improves capacity utilization by considering an enumerative process. However, it can hardly achieve an optimal balance among utilization of capacity, system-level efficiency, passengers satisfaction, and benefits of train operating companies.In this research, we do not consider any preprocessing step for PMTS plans nor enumerative processes: trains and PMTSs are scheduled at the same time, in order to systematically search a larger solution space and achieve an optimal performance of the overall system.

4.2.2 Problem statement and notations

Given a railway network with stations and segments, a set of real trains from prespecified origins to destinations and a set of virtual trains implicitly representing PMTSs over a given planning horizon, the integrated train scheduling and PMTSs planning problem consists in finding the best incorporation between trains and PMTSs, simultaneously determining train aspects (i.e., the orders, routes, and departure and arrival times of trains) and maintenance aspects (i.e., the work-space, work-time, and shape[1]). In our integrated optimization problem, the following inputs are considered: 1) a planning horizon T, in which trains and PMTSs are scheduled; 2) a railway network with stations and segments, which are further modeled as a sequence of nodes and cells; 3) a set of virtual trains (representing PMTSs, i.e., a set of PM tasks for railway lines or segments) with their origins and destinations (associated with the work-space of PMTSs), safety headway times (associated with the duration of PMTSs), and the minimum and maximum dwell times (associated with the shape of PMTSs); 4) a set of real trains with their origins, destinations, earliest departure times, preferred arrival times, free-flow running times over cells, and safety headway times. Note that the free-flow running times of trains over the cells are computed based on the planned speed profile, which can in general be different for each train. A fixed speed profile model is used in this research, as common in the train scheduling studies.

[1] There are two common shapes of PMTSs: rectangle or stairway. If the starting times of PM tasks on a sequence of block sections are same, as well as the end times, then the blockage of the PM tasks on a time-space graph will result in a rectangle shape. If there are spaced starting times for the PM tasks on a sequence of block sections, then it will show a stairway shape on a time-space graph.

Table 4.1 gives the general subscripts, sets, input parameters, and decision variables of the proposed optimization approach.

Table 4.1: General subscripts

Symbol	Description
	Subscripts and sets
N	set of nodes
E	set of cells
T	planning horizon
R	set of real trains
V	set of virtual trains (PMTSs), $\lvert V \rvert$ is the total number of virtual trains
F	the total set of real and virtual trains, $F = R \cup V$
i,j,k	node index, $i, j, k \in N$
e	cell index, generated by two adjacent nodes i and j, $e = (i, j) \in E$
t	scheduling time index, $t \in \{1, ..., T\}$
r	real train index, $r \in R$
v	virtual train index, i.e., preventive maintenance time slot (PMTS) index, $v \in V$
f	train index, $f \in F$
E_f	set of cells train f may use, $E_f \subseteq E$
E_S	set of cells corresponding to stations on railway network, $E_S \subseteq E$
	Input parameters and sets
o_f	origin node of train f
s_f	destination node of train f
$\sigma_{f,i,j}$	free-flow (minimum) running time of train f to drive through cell (i, j)
$\delta^{\min}_{f,i,j}$	minimum dwell time for train f on cell (i, j)
$\delta^{\max}_{f,i,j}$	maximum dwell time for train f on cell (i, j)
$C_{i,j,t}$	flow capacity on cell (i, j) at time t, set to be 1 by default
q_r	the ideal arrival time of real train r at its destination node
$g_{r,i,j}$	safety headway time between cell occupancy and arrival of real train r on cell (i, j)
$h_{r,i,j}$	safety headway time between departure of real train r on cell (i, j) and cell release
$w_{v,i,j}$	safety headway time (pre-blockage time interval) of virtual train v on cell (i, j), i.e., duration of PMTS corresponding to virtual train v on cell (i, j)
	Decision variables
$a_{f,i,j,t}$	binary(flag) arrival variable, $a_{f,i,j,t} = 1$ if train f has already arrived at cell (i, j) by time t, and otherwise $a_{f,i,j,t} = 0$
$d_{f,i,j,t}$	binary(flag) departure variable, $d_{f,i,j,t} = 1$ if train f has already departed from cell (i, j) by time t, and otherwise $d_{f,i,j,t} = 0$

continued on next page

continued from previous page

Symbol	Description
$y_{f,i,j,t}$	binary time-space occupancy variable, $y_{f,i,j,t} = 1$, if train f occupies cell (i,j) at time t, and otherwise $y_{f,i,j,t} = 0$
$x_{f,i,j}$	binary train routing variable, $x_{f,i,j} = 1$ if train f selects cell (i,j) on the rail network, and otherwise $x_{f,i,j} = 0$
$\tau_{f,i,j}$	travel time of train f on cell (i,j)

Cells in the set E_f that train f may use must be consecutive and connect origin node o_f to destination node s_f. Cells used to represent stations on the railway network are contained in the set E_S. Each train f has a free-flow (minimum) running time $\vartheta_{f,i,j}$, minimum dwell time $\delta^{\min}_{f,i,j}$, and maximum dwell time $\delta^{\max}_{f,i,j}$ from node i to j. Each cell (i,j) has a flow capacity $C_{i,j,t}$ that indicates how many trains are allowed on cell (i,j) at time step t. The flow capacity $C_{i,j,t}$ normally defaults to one, which means that only one train is allowed on any cell at any time. Each real train is assigned an ideal timetable, and q_r is the ideal arrival time of real train r at its destination node. Train movements are separated by a safety headway time interval, which depends on the train length, speed, and route chosen. The parameters $g_{r,i,j}$ and $h_{r,i,j}$ are introduced to present cell pre-blockage time and post-release times for real trains. The parameter $w_{v,i,j}$ is the safety headway time (pre-blockage time interval) of virtual trains, which corresponds to the durations of PMTSs.

Four types of variables are used to formalize the routing and scheduling decisions: departure time variables d, arrival time variables a, cell occupancy variables y, and route selection variables x. Specifically, $x_{f,i,j}$ captures the routing decisions on a rail network, $y_{f,i,j,t}$ describes a detailed train route through the extended time-space network, and the pair of flag variables $a_{f,i,j,t}$ and $d_{f,i,j,t}$ represent both temporal and spatial resource consumption of trains. The travel time $\tau_{f,i,j}$ is then a consequence of the interaction of all those variables for all trains on the network.

We make the following assumptions: (1) each train is represented by a point (with length 0) ; (2) train acceleration and deceleration processes are not considered; (3) for a double-track railway segment between two stations, each track is modeled as a sequence of directional cells (i.e., directional block sections), and for a single-track railway segment, the only track between two stations is modeled as bi-directional cells (i.e., bi-directional block section); (4) every station is simplified to a number of main and siding track(s), which can be further modeled as a single cell or a set of cells; (5) the granularity of time is one minute; (6) only one real train is permitted on a cell at any given time, and a real train and a virtual train (representing a PMTS) cannot occupy a cell at the same time. However, a cell can be used by more than one virtual train, since two maintenance works can be performed at once.

4.3 Mathematical formulation

In this section, a virtual-train-based formulation technique is first presented to describe cell reservation and occupancy of trains and PMTSs simultaneously, based on the flag-variable-based formulation method introduced in Section 2.4.2 for handling spatial occupancy and safety headway constraints. We then formulate the integrated optimization problem of train scheduling and PMTS planning on a general railway network based on the proposed virtual-train-based formulation.

4.3.1 Virtual-train-based formulation

In a train timetable, the interaction of trains and PMTSs can be considered as an allocation of cell capacity. In other terms, maintenance can be regarded as a kind of activity carried out on cells, as train movement is. Thus, we propose a virtual-train-based formulation technique, where each PMTS is represented by a virtual train with a specifically designed safety headway $w_{v,i,j}$.

For instance, as illustrated in Figure 4.3, the PMTS for cell (i,j) is represented by a virtual train v. In this figure, the cell pre-blockage parameter $w_{v,i,j}$ is 50 minutes, which implies 50 minutes duration of the PMTS on cell (i,j), corresponding to the pink rectangle in Figure 4.3. The running time of a virtual train is set to 0 in any case. The running time has no effect on the duration of PMTS, and only the safety headway time (pre-blockage time interval) of a virtual train has.

Table 4.2 lists some properties of real trains, virtual trains, and PMTSs. The properties of virtual trains reflect the corresponding properties of PMTSs, which should be considered while formulating PMTSs.

As listed, some similarities are found between real trains and virtual trains, namely the origin and destination are pre-specified, and dwell time ranges from the given minimum value to the maximum value. It is worth noting that the minimum dwell time of a real train is the required time to complete the processes of passengers boarding/alighting, goods loading/unloading, etc. In this research, the maximum dwell time

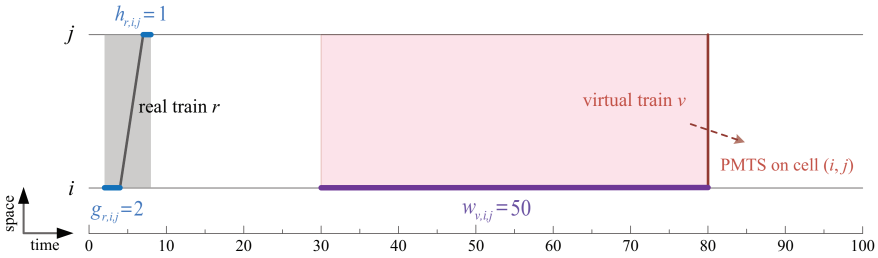


Figure 4.3: Illustration of the virtual-train-based formulation technique

Table 4.2: Properties of real trains and virtual trains and the corresponding properties to PMTSs

Category of property	Real train	Virtual train		PMTS
origin and destination route	pre-specified free	pre-specified fixed	->	work space
running time over cells safety headway	no less than the minimum value pre-determined, a relatively *small* value	zero pre-determined, a relatively *large* value	->	work time window (start and end times, duration)
dwell time at station	ranges from the given minimum value to maximum value	ranges from the given minimum value to maximum value	->	shape
cell occupancy	only one real train is allowed on a cell at any time, and a real train has conflicts with a virtual train	no capacity constraint for virtual trains, but a virtual train has conflicts with a real train	->	a PM work has conflicts with train movements, and there is no conflict between PMTSs

is used to avoid unpermitted stops of trains, e.g., if a train is required to stop only at its origin and destination, then the maximum dwell times at intermediate stations are set to zero. However, the minimum and maximum dwell times of virtual trains are used to shape PMTSs. The differences between real trains and virtual trains are as follows: (1) the routes of real trains are optimized, and routes of virtual trains (i.e., a set of sections to be maintained, but their relative order is free) are fixed input; (2) the running times of real trains over cells cannot be less than the free-flow running times, and those of virtual trains are set to zero; (3) the safety headways of real trains are relative small (e.g., 2 minutes), and those of virtual trains are much larger (e.g., 50 minutes); and (4) regarding the cell occupancy, a cell can be occupied by more than one virtual train at the same time, since PM works can be performed simultaneously at an interchange station connecting two or more lines.

The formulation techniques used in this research have significant advantages to achieve the integration of train scheduling and PMTSs planning. The flag-variable-based representation enables many unique formulation features. First, it can easily capture the complex safety headway constraints on a general rail network at the microscopic level, with or without a pre-determined train route, by reformulating the temporal and spatial resource occupancy of trains. Second, it can flexibly describe the properties of different types of trains (i.e., real train and virtual train) on each cell at any time, and it further provides the possibility of scheduling trains and PMTSs simultaneously through a virtual-train-based formulation technique. Third, it enables an efficient problem decomposition mechanism by trains, while each subproblem is relatively simple to solve on an extended time-space network. Note that the problem decomposition mechanism used by the Lagrangian-relaxation-based solution algorithm will be detailed in Section 4.4. Furthermore, thanks to the virtual-train-based formulation technique, the problem decomposition mechanism by trains has strong applicability for the integrated problem of scheduling trains and planning PMTSs, since PMTSs are viewed as virtual trains, whose properties can be classified into the same categories with that of real trains.

4.3.2 Optimization problem

We assign to each real train an ideal timetable, which would be the most desirable timetable for the real train (e.g., the Periodic Service Intention of Caimi et al., 2011). However, the given timetable may be modified to satisfy the safety or operational requirements. We now propose the integrated optimization problem of train scheduling and PMTS planning, denoted as the P_1 problem. The objective function is formulated as follow:

$$\min \quad Z_{\mathrm{P}_1} = \sum_{r \in R} \sum_{(i,s_r) \in E_S} \left| \sum_{t=1,\ldots,T} t \cdot [d_{r,i,s_r,t} - d_{r,i,s_r,t-1}] - q_r \right| \tag{4.1}$$

to minimize the sum of the absolute arrival time deviations of real trains at destinations between the ideal and actual timetables. For the PMTSs, no desired work time is given; so they are not considered in the objective function.

The following three constraints:

$$\sum_{j:\left(o_f,j\right)\in E_f} x_{f,o_f,j} = 1, \qquad \forall f \in F, \tag{4.2}$$

$$\sum_{i:(i,j)\in E_f} x_{f,i,j} = \sum_{k:(j,k)\in E_f} x_{f,j,k}, \qquad \forall f \in F, j \in N\backslash\left\{o_f, s_f\right\}, \tag{4.3}$$

$$\sum_{j:\left(i,s_f\right)\in E_f} x_{f,i,s_f} = 1, \qquad \forall f \in F, \tag{4.4}$$

are used to ensure the flow balance of train f on the rail network, at origin, intermediate, and destination nodes respectively. Each train f has to choose one and only one route that connects its origin to its destination. Recall that the route is optimized for real trains, but pre-determined for virtual trains (PMTSs).

The transition of train f within cell (i, j) is enforced by

$$d_{f,i,j,t} \leq a_{f,i,j,t}, \qquad \forall f \in F, (i, j) \in E_f, t = 1,...,T, \tag{4.5}$$

i.e., the flag departure variable cannot be larger than the flag arrival variable for train f on cell (i, j) at time t.

If two adjacent cells (i, j) and (j,k) are consecutively used by train f, then the departure time of train f from cell (i, j) should equal its arrival time at cell (j,k), i.e., $d_{f,i,j,t} = a_{f,j,k,t}$, which is forced by

$$\sum_{i:(i,j)\in E_f} d_{f,i,j,t} = \sum_{k:(j,k)\in E_f} a_{f,j,k,t}, \quad \forall f \in F, j \in N\backslash\left\{o_f, s_f\right\}, t = 1,...,T. \tag{4.6}$$

We use the constraint

$$x_{f,i,j} = a_{f,i,j,T}, \qquad \forall f \in F, (i, j) \in E_f \tag{4.7}$$

to link the variable $a_{f,i,j,t}$ of the time-space network with the variable $x_{f,i,j}$ of the physical network, in order to formulate whether cell (i, j) is selected by train f for traversing the network from its origin to destination.

The train travel time constraint

$$\tau_{f,i,j} = \sum_{t=1,...,T} t \cdot \left[d_{f,i,j,t} - d_{f,i,j,t-1}\right] - \sum_{t=1,...,T} t \cdot \left[a_{f,i,j,t} - a_{f,i,j,t-1}\right] \qquad \forall f \in F, (i, j) \in E_f \tag{4.8}$$

computes the actual travel time $\tau_{f,i,j}$ of train f on cell (i, j).

The constraints

$$\tau_{r,i,j} \geq \left[\delta^{\min}_{r,i,j} + \sigma_{r,i,j}\right] \cdot x_{r,i,j}, \qquad \forall r \in R, (i, j) \in E_r \tag{4.9}$$

$$\tau_{r,i,j} \leq \left[\delta^{\max}_{r,i,j} + \sigma_{r,i,j}\right] \cdot x_{r,i,j}, \qquad \forall r \in R, (i, j) \in E_r \tag{4.10}$$

ensure that the train travel time $\tau_{f,i,j}$ satisfies the required free-flow running time, as well as the minimum and maximum dwell times at stations.

The flag arrival variable $a_{f,i,j,t}$ and flag departure variable $d_{f,i,j,t}$ have a non-decreasing behavior as a function of time t. Thus, if train f has arrived at or departed from cell

(i, j) by time t, then the variables will have a value of 1 for all later time periods $t' \geq t$, as formulated by

$$a_{f,i,j,t-1} \leq a_{f,i,j,t}, \qquad \forall f \in F, (i, j) \in E_f, t = 1, ..., T, \tag{4.11}$$

$$d_{f,i,j,t-1} \leq d_{f,i,j,t}, \qquad \forall f \in F, (i, j) \in E_f, t = 1, ..., T. \tag{4.12}$$

In comparison with the optimization problem proposed by Meng and Zhou (2014), the following constraints (4.14)-(4.17) are newly proposed, in order to simultaneously schedule real trains and virtual trains (PMTSs).

The running time of virtual train v on cell (i, j) is set to 0, which is forced by

$$\tau_{v,i,j} = 0, \qquad \forall v \in V, (i, j) \in E_v. \tag{4.13}$$

The free-flow running time for all virtual trains is set to 0; thus, in order to guarantee the duration of PMTSs on their origin cells, a virtual train v is not allowed to depart from its origin earlier than the given time $w_{v,o_v,j}$, which is formulated as follows:

$$\sum_{t=1,...,T} t \cdot \left[a_{v,o_v,j,t} - a_{v,o_v,j,t-1}\right] \geq w_{v,o_v,j}, \qquad \forall v \in V, (o_v, j) \in E_v, \tag{4.14}$$

The constraints

$$y_{r,i,j,t} = \begin{cases} a_{r,i,j,t+g_{r,i,j}}, & \text{if} \quad 1 \leq t < h_{r,i,j} + 1 \\ a_{r,i,j,t+g_{r,i,j}} - d_{r,i,j,t-h_{r,i,j}}, & \text{if} \quad h_{r,i,j} + t = 1, ..., T - g_{r,i,j} \\ 1 - d_{r,i,j,t-h_{r,i,j}}, & \text{if} \quad T - g_{r,i,j} < t \leq T \end{cases} \\ \forall r \in R, (i, j) \in E_r \tag{4.15}$$

$$y_{v,i,j,t} = \begin{cases} a_{v,i,j,t+w_{v,i,j}} - d_{v,i,j,t}, & \text{if} \quad t = 1, ..., T - w_{v,i,j} \\ x_{v,i,j} - d_{v,i,j,t}, & \text{if} \quad T - w_{v,i,j} < t \leq T \end{cases} \\ \forall v \in V, (i, j) \in E_v \tag{4.16}$$

draw the relation between the cell blockage variable $y_{f,i,j,t}$ and the flag arrival and flag departure variables, i.e., $a_{f,i,j,t}$ and $d_{f,i,j,t}$, for real trains $r \in R$ and virtual trains $v \in V$ respectively.

The cell capacity constraint

$$\frac{\sum\limits_{v \in V:(i,j) \in E_v} y_{v,i,j,t}}{|V|} + \sum_{r \in R:(i,j) \in E_r} y_{r,i,j,t} + \sum_{r' \in R:(j,i) \in E_{r'}} y_{r',j,i,t} \leq C_{i,j,t}, \\ \forall (i, j) \in E, t = 1, ..., T \tag{4.17}$$

explicitly ensures that the number of trains occupying cell (i, j) is less than the capacity of cell (i, j), which defaults to 1. It should be noted that the occupancy of the cell from i to j should also be counted into the occupancy of the cell from j to i by train f for bi-directional traffic, and vice versa, as cell (i, j) and (j, i) essentially refer to one physical track circuit. Moreover, a cell is allowed to be occupied by more than one virtual train at any time, because the maintenance tasks for different lines can be implemented simultaneously at an interchange station. The mean cell occupancy rate of virtual trains is used to indicate that there is always no conflict between virtual trains, since the mean occupancy rate cannot be greater than 1 (i.e., the default value of cell capacity $C_{i,j,t}$) in any case and will be greater than 0 if any virtual train is running on the cell. This cell capacity constraint is specially designed for the integration of train scheduling and PMTSs planning, which differs from the capacity constraints used by Meng and Zhou (2014).

The P_1 problem is an MILP problem, including the objective function (4.1) and the constraints (4.2)-(4.17). By dealing with the integration of the train scheduling and the PMTS planning, the complexity of the P_1 problem increases, which makes the problem difficult to solve.

4.4 Lagrangian-relaxation-based solution framework

This section aims to solve the proposed optimization problem (P_1), which incorporates the PMTS planning into the train scheduling, through a Lagrangian-relaxation-based solution framework. Complicating constraints in the P_1 problem are first dualized, which results in a relaxed problem (P_{LR}). The P_{LR} problem is further decomposed into a sequence of single-train-based (for either real trains or virtual trains) subproblems, which are solved by a time-dependent least-cost path algorithm. The Lagrangian-relaxation-based solution framework used in this research can help constructing a lower bound and provide a good base solution for generating feasible solutions with valid upper bounds. The subgradient method is used to update Lagrangian multipliers. We then detail the overall Lagrangian-relaxation-based solution framework and the underlying label correcting algorithm for solving the time-dependent least-cost path problem, as well as the priority-rule-based method for transforming dual solutions to feasible solutions.

4.4.1 Dualizing complicating constraints

The constraints in the P_1 problem can be divided into two categories, i.e, easy constraints and difficult constraints. The former category includes constraints (4.2)-(4.16), because each of them is only directly associated with an individual train. The latter category contains the cell capacity constraint (4.17), which is generally difficult to solve, reflecting the interaction of all trains on the same cell.

We introduce a set of non-negative Lagrangian multipliers $\alpha_{i,j,t}$ for dualizing the cell capacity constraint (4.17). The Lagrangian relaxation problem (P_{LR}) can be described with a penalty term as follows:

$$\begin{aligned} \min \quad Z_{(\mathrm{P}_{\mathrm{LR}})} = & \sum_{r\in R}\sum_{(i,s_r)\in E_S} \left| \sum_{t=1,\ldots,T} t\cdot[d_{r,i,s_r,t} - d_{r,i,s_r,t-1}] - q_r \right| + \\ & \sum_{(i,j)\in E}\sum_{t=1,\ldots,T} \alpha_{i,j,t}\cdot \left[\frac{\sum_{v\in V} y_v(i,j,t)}{|V|} + \sum_{r\in R} y_{r,i,j,t} + \sum_{r'\in R} y_{r',j,i,t} - C_{i,j,t} \right], \end{aligned} \tag{4.18}$$

subject to constraints (4.2)-(4.16).

The multipliers $\alpha_{i,j,t}$ can be interpreted as the cost charged for using cell (i, j) at time t. Essentially, the major goal of the Lagrangian function is to balance the total train deviation from the ideal timetable and the cost for using limited resources (cells) through paying appropriate resource usage prices. It is worth noting that constraints (4.2)-(4.16) can be easily solved in an extended time-space network, which can be derived from the physical network. Specifically, they can be modeled by cell traveling arcs, waiting arcs, and dummy arcs. The method that we use to construct the extended time-space network is same as the one proposed by Meng and Zhou (2014). Interested readers may refer to this reference for more details.

4.4.2 Problem decomposition

By re-grouping the variables in (4.18), we can get the following Lagrangian dual problem:

$$\max_{\alpha_{i,j,t}\geq 0} \quad Z_{\mathrm{LR}} = -\sum_{(i,j)\in E}\sum_{t=1,\dots,T} \alpha_{i,j,t}\cdot C_{i,j,t} + \min \mathrm{LR}_r + \min \frac{\mathrm{LR}_v}{|V|}, \tag{4.19}$$

where

$$\mathrm{LR}_r = \sum_{(i,s_r)\in E_S} \left| \sum_{t=1,\dots,T} t\cdot[d_{r,i,s_r,t} - d_{r,i,s_r,t-1}] - q_r \right| + \sum_{(i,j)\in E}\sum_{t=1,\dots,T} \alpha_{i,j,t}\cdot y_{r,i,j,t}, \tag{4.20}$$

$$\mathrm{LR}_v = \sum_{(i,j)\in E}\sum_{t=1,\dots,T} \alpha_{i,j,t}\cdot y_{v,i,j,t}. \tag{4.21}$$

The original problem is then separated into a sequence of single train (either real train or virtual train) optimization subproblems, and the inner minimization problem is concerned with the sum of LR_r for all real trains and the mean of LR_v for all virtual trains. In the decomposed subproblem LR_r, the deviation time of a real train r (i.e., the first portion of (4.20)) is expressed as the sum of the absolute deviation between the arrival time of real train r at its destination and the corresponding ideal arrival time. The resource price of a real train r or a virtual train v (i.e., the second portion of (4.20) or (4.21)) for traversing the network from its origin to its destination is computed by summing $\alpha_{i,j,t}$ over all selected cells within associated time spans.

With a given set of resource prices, we aim at finding the least-cost path of a train from its origin to its destination. As a result, the single-train-based subproblems are now transformed to a sequence of time-dependent least-cost path problems. Those problems seek to find the best resource utilization scheme for each train, subject to constraints (4.2)-(4.16), which restrict the possible paths in the time-space network.

By dualizing the complicating capacity constraint, the whole problem can be decomposed into several subproblems, and each subproblem corresponds to a real train or a virtual train (PMTS). The nature of the LR subproblems is still MILP problems, and we solve them by using a time-dependent least-cost path algorithm.

4.4.3 Sub-gradient method for updating Lagrangian multipliers

Since the dual cost function (4.19) is not differentiable everywhere, a standard sub-gradient method is used to update the multipliers $\alpha_{i,j,t}$. The resource usage prices are iteratively updated by

$$\alpha^{u+1}_{i,j,t} = \max\left\{0, \alpha^{u}_{i,j,t} + \lambda^{u} \cdot \left[\frac{\sum\limits_{v\in V} y_{v,i,j,t}}{|V|} + \sum_{r\in R} y_{r,i,j,t} + \sum_{r'\in R} y_{r',j,i,t} - C_{i,j,t}\right]\right\} \tag{4.22}$$

where the superscript u is the iteration index used in the dual updating procedure; $\alpha^{u}_{i,j,t}$ and λ^{u} denote the cell multiplier value and step size at iteration u respectively. In the optimum search process, the step size parameter is updated as $\lambda^{u} = \frac{1}{u+1}$, which is commonly-used. If a resource has not been used within several recent iterations, the algorithm automatically resets the price of the unused resource back to zero. With this dynamic generating scheme, the set of Lagrangian multipliers is updated along the iterative process, and the multi-dimensional resource price vector can be relatively easily stabilized.

4.4.4 Priority-rule-based algorithm

For generating a feasible solution at each iteration, a priority-rule-based algorithm is used for transforming a dual solution into a feasible solution. The schedule of trains is computed in a sequential manner according to train priorities, which are dynamically determined by the Lagrangian profits, i.e., the ratio of total free-flow travel time divided by expected total travel time for real trains in the dual solution. Note that the free-flow travel time of virtual trains is 0, so that the way used for real trains to compute the Lagrangian profit would always give an answer of 0. Thus the Lagrangian profit of virtual trains is randomly generated. The algorithm is described by the following three steps:

Step 1. **Train priority ranking**

Rank trains by decreasing values of Lagrangian profits. The Lagrangian profit of real trains is the ratio of the total free-flow travel time divided by the total travel time in the dual solution, and that of virtual trains is randomly generated.

Step 2. **Scheduling trains (including real train and virtual train) one by one**

(1) schedule the train f with the highest priority by applying the time-dependent least-cost path algorithm. If a status of infeasibility occurs, a warning will be given.

(2) fix the routes and departure and arrival times at passing stations for train f, and record the capacity usage of train f on the railway network.

(3) if all trains have been scheduled, move to **Step 3**, otherwise, loop to **Step 2**.

Step 3. **Update and output the upper bound**

(1) compute the objective value of the heuristic solution obtained in **Step 2**.

(2) update the upper bound by using the obtained objective value, if the current global upper bound is greater than the objective value and the given planning horizon is not exceeded.

(3) output the train routes and train departure and arrival times at passing stations, and the updated upper bound in the current iteration.

It should be remarked that using the time-dependent least-cost path algorithm to schedule a real train, a plan (including the train departure time, arrival time, and route) with the minimum deviation time from the ideal timetable for the real train is found according to the given or updated set of resource prices. When scheduling a virtual train (PMTS), the time-dependent least-cost path algorithm is to search the available time slots for performing the PM task with respect to the given or updated set of resource prices. As no desired work time is required for the PMTS, the algorithm takes possession of the available time slot found first for the scheduled PMTS. Moreover, the planning horizon should be carefully chosen and appropriate for the planned trains and PMTSs. If a status of infeasibility occurs, a warning will be given by the algorithm.

4.4.5 Overall Lagrangian-relaxation-based solution framework

The Lagrangian-relaxation-based solution framework is illustrated in Figure 4.4.

Input: identical to the input of the P_1 problem

Output: dual solutions of the P_{LR} problem, feasible solutions of the P_1 problem, and the corresponding optimality gap ε.

Step 1. **Initialization**

Let $u = 1$, initialize the multipliers $\alpha^1_{i,j,t} = 0$, step size $\lambda^1 = 0.5$, local lower bound $z^1_{\text{LB}} = 0$, global lower bound $Z^1_{\text{LB}} = 0$, local upper bound $z^1_{\text{UB}} = 0$, and global upper bound $Z^1_{\text{UB}} = 0$.

Step 2. **Solving the relaxed problem ($\mathbf{P}_{\text{LR}}$)**

(1) solve the subproblems LR_r and LR_v of the P_{LR} problem by a time-dependent least-cost path algorithm, which is equivalent to the one proposed by Meng and Zhou (2014);

(2) compute the local lower bound of the P_{LR} problem for the current iteration u, denoted by z^u_{LB}, then update global lower bound by

$$Z^u_{\text{LB}} = \begin{cases} z^1_{\text{LB}}, & \text{if } u = 1 \\ \max\left\{z^u_{\text{LB}}, Z^{u-1}_{\text{LB}}\right\}, & \text{if } u > 1. \end{cases} \tag{4.23}$$

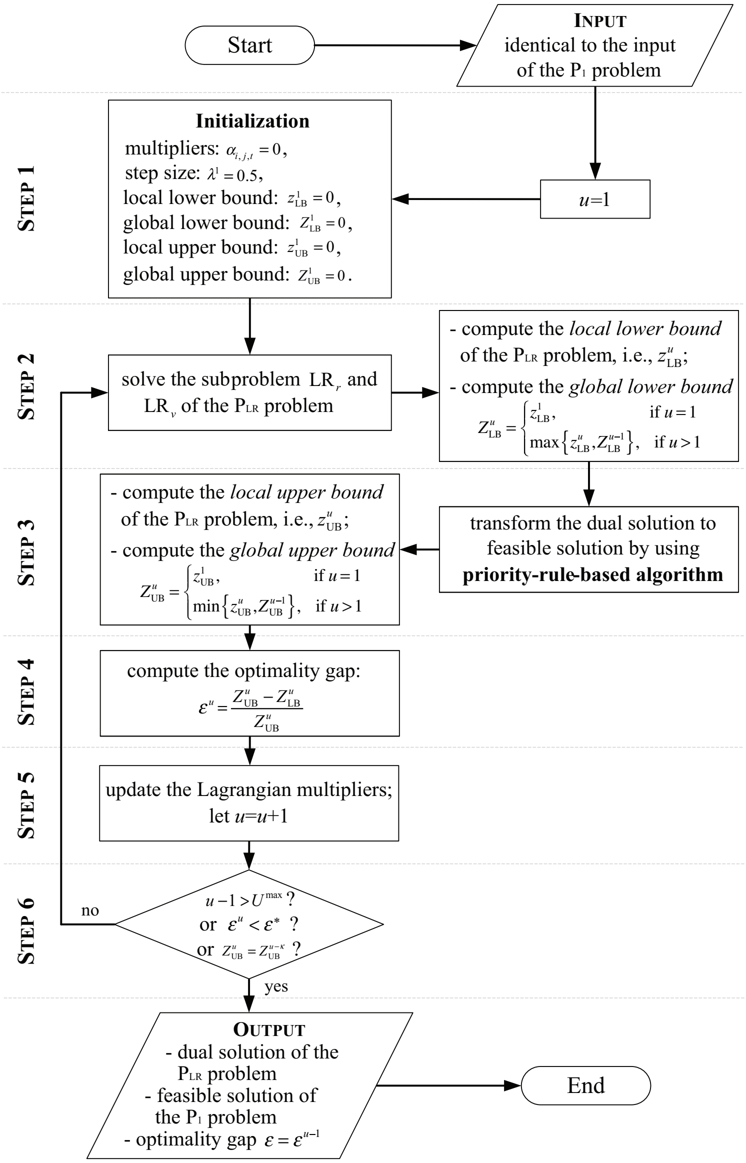

Start
INPUT
identical to the input of the P1 problem
STEP 1
Initialization
multipliers: $\alpha_{i,j,t} = 0$,
step size: $\lambda^1 = 0.5$,
local lower bound: $z_{\text{LB}}^1 = 0$,
global lower bound: $Z_{\text{LB}}^1 = 0$,
local upper bound: $z_{\text{UB}}^1 = 0$,
global upper bound: $Z_{\text{UB}}^1 = 0$.
u=1
STEP 2
solve the subproblem LR_r and LR_v of the P_LR problem
- compute the *local lower bound* of the P_LR problem, i.e., z_{LB}^u;
- compute the *global lower bound*
$Z_{\text{LB}}^u = \begin{cases} z_{\text{LB}}^1, & \text{if } u = 1 \\ \max\{z_{\text{LB}}^u, Z_{\text{LB}}^{u-1}\}, & \text{if } u > 1 \end{cases}$
STEP 3
transform the dual solution to feasible solution by using **priority-rule-based algorithm**
- compute the *local upper bound* of the P_LR problem, i.e., z_{UB}^u;
- compute the *global upper bound*
$Z_{\text{UB}}^u = \begin{cases} z_{\text{UB}}^1, & \text{if } u = 1 \\ \min\{z_{\text{UB}}^u, Z_{\text{UB}}^{u-1}\}, & \text{if } u > 1 \end{cases}$
STEP 4
compute the optimality gap:
$\varepsilon^u = \frac{Z_{\text{UB}}^u - Z_{\text{LB}}^u}{Z_{\text{UB}}^u}$
STEP 5
update the Lagrangian multipliers; let u=u+1
STEP 6
$u - 1 > U^{\max}$?
or $\varepsilon^u < \varepsilon^*$?
or $Z_{\text{UB}}^u = Z_{\text{UB}}^{u-\kappa}$?
no
yes
OUTPUT
- dual solution of the P_LR problem
- feasible solution of the P1 problem
- optimality gap $\varepsilon = \varepsilon^{u-1}$
End

Figure 4.4: Illustration of the Lagrangian-relaxation-based solution framework

Step 3. **Transforming dual solutions to feasible solutions**

Use the priority-rule-based algorithm introduced in Section 4.4.4 to transform the dual solutions into feasible solutions of the P_1 problem, and compute the upper bound of the P_{LR} problem for the current iteration u, denoted by z_{UB}^u, then update global upper bound by

$$Z^u_{\mathrm{UB}} = \begin{cases} z^1_{\mathrm{UB}}, & \text{if } u = 1 \\ \min\left\{z^u_{\mathrm{UB}}, Z^{u-1}_{\mathrm{UB}}\right\}, & \text{if } u > 1 \end{cases} \tag{4.24}$$

Step 4. **Computing optimality gap** ε

Compute the optimality gap ε^u between Z^u_{LB} and Z^u_{UB} for the current iteration u, i.e., $\varepsilon^u = \frac{Z^u_{\mathrm{UB}} - Z^u_{\mathrm{LB}}}{Z^u_{\mathrm{UB}}}$.

Step 5. **Updating Lagrangian multipliers**

Update the Lagrangian multipliers for the next iteration $u+1$ by (4.22) and let $u = u+1$.

Step 6. **Termination condition**

The algorithm will be terminated if one of the following conditions are satisfied:

(1) $u-1 > U^{\max}$, the current iteration $u-1$ is larger than the given maximum iteration $U^{\max}$;

(2) $\varepsilon^u < \varepsilon^*$, the current optimality gap ε^u is smaller than the expected gap ε^*;

(3) $Z^u_{\mathrm{UB}} = Z^{u-\kappa}_{\mathrm{UB}}$, the global upper bound Z^u_{UB} has not improved for a given number of iterations κ.

Otherwise, loop to **Step 2**.

4.5 Case study

This section first presents the description of two experimental networks under consideration, i.e., an artificial network and a real-world network adapted from a Chinese railway network, followed by the computational results of the integrated optimization problem (P_1) and the Lagrangian-relaxation-based solution framework. All the following experiments are performed on a computer with an Intel® Core$^{\mathrm{TM}}$ i7 @ 2.00 GHz processor and 16GB RAM.

4.5.1 Setup

(1) Description of the artificial network

We first adopt an artificial network, as shown in Figure 4.5, with 2 lines and 1 station, which consists of 32 nodes and 36 cells. Double-track Line_1 merges with single-track Line_2 at Station_X, which is represented by nodes (16, 17, 18, 19) and (23, 22, 21, 20) and designed with 1 siding track for each direction, i.e., cell (17, 18) and (22, 21), where trains may stop.

The directionality (up/down) of each cell is marked in Figure 4.5, and the capacity of each cell is set to 1. For all cells, the minimum dwell time and the safety headway

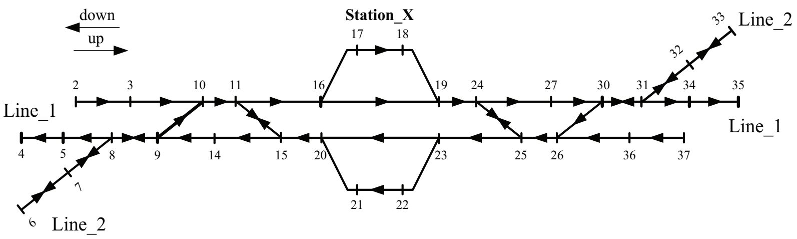


Figure 4.5: An artificial experimental network

Table 4.3: Data for the artificial network

	ID	Origin (O)	Destination (D)	Preferred departure time window (unit: minute)	Ideal arrival time (unit: minute)	Free-flow running time (unit: minute)	Safety headway (unit: minute)
Trains	train_1	2	33	$[0, T]$	15	1	g=1 h=2
	train_2	37	6	$[0, T]$	16		
	train_3	37	4	$[0, T]$	18		
	train_4	33	6	$[0, T]$	18		
	train_5	6	33	$[0, T]$	19		
	train_6	6	35	$[0, T]$	16		
	train_7	2	35	$[0, T]$	65		
	train_8	37	4	$[0, T]$	55		
	train_9	33	4	$[0, T]$	65		
	train_10	6	35	$[0, T]$	62		
	train_11	37	4	$[0, T]$	75		
	train_12	2	33	$[0, T]$	72		
	train_13	37	6	$[0, T]$	83		
	train_14	33	4	$[0, T]$	80		
	train_15	6	33	$[0, T]$	93		
	train_16	37	4	$[0, T]$	95		
	train_17	2	35	$[0, T]$	98		
	train_18	33	6	$[0, T]$	100		
	train_19	37	6	$[0, T]$	110		
	train_20	6	35	$[0, T]$	115		
PMTSs	PMTS_1	2	35	–	–	0	w=29
	PMTS_2	6	33	–	–	0	w=29
Planning horizon T	50, 80, 100, 150, 200		Minimum and maximum dwell time on cell (17, 18) and (22, 21)				0 (min), 40 (max)

times g and h are set to 0, 1 and 2 minutes respectively. We consider 20 trains (namely train_1, ..., train_20) and 2 PMTSs (i.e., virtual trains PMTS_1 and PMTS_2 for the up direction of Line_1 and Line_2 respectively). The data for this artificial network is given in Table 4.3, including the origin, destination, free-flow running time, and safety headway time for each train and PMTS.

(2) Description of a realistic network: a Chinese railway network

The realistic case study used in this chapter refers to part of a Chinese railway network. The complex railway network is sketched and shown in Figure 4.6(a). There are 5 stations, namely station W, N, E, S, and M, with at least 2 platforms and up to 17 platforms. The network is composed of 454 nodes and 513 cells, with 2 main double-track lines, merged at station M.

A total of 21 trains that run in a given planning horizon T (400 minutes, which is

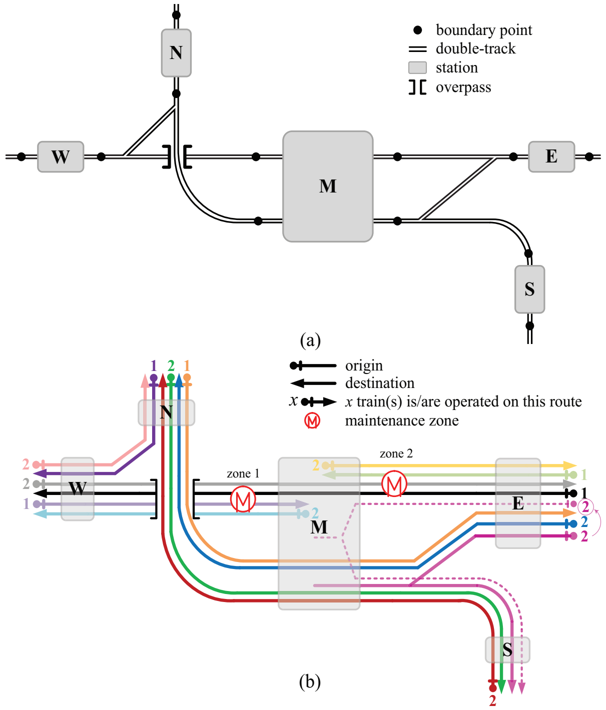


Figure 4.6: A realistic network adapted from a partial Chinese railway network

large enough for the planned trains and PMTSs) are first scheduled over the whole network, meanwhile 2 PMTSs are planned, as shown in Figure 4.6(b). Seven global (bi-)directional O-D pairs (i.e., pairs [M↔W], [M↔E], [W↔N], [W↔E], [N↔S], [N↔E], and [E→S]) are determined. The number of trains operated is labeled at the origin for each O-D pair. Note that Figure 4.6(b) briefly sketches the global O-D pairs and does not give the exact routes in stations, which implies that multiple options of routes are provided for each train. For instance, the train traversing from station E to S has another rerouting option shown as the pink dashed line of Figure 4.6(b). The 21 trains and the 2 PMTSs under consideration are given in Table 4.4, without a mark of †. Moreover, to evaluate the performance of the proposed algorithm on a larger-scale instance, we further consider 10 additional trains and 2 PMTSs, which are detailed and marked by † in Table 4.4.

Regarding the data of the larger amount of trains and PMTSs, 9 global (bi)-directional O-D pairs (i.e., pairs [M ↔ W], [M ↔ E], [M↔S], [M↔N], [W↔N], [W↔E], [N↔S], [N↔E], and [E→S]) are determined, as shown in Figure 4.7. The number of trains operated is labelled at the origin for each O-D pair.

In the remainder of this section, we evaluate the benefits of the integrated optimization problem (P_1) in Section 4.5.2(1), which is solved by CPLEX optimization studio

Table 4.4: Dataset for the realistic network

	ID	Origin (O), i.e., station (node_id)	Destination (D), i.e., station (node_id)	Preferred departure time window (unit: minute)	Ideal arrival time (unit: minute)	Speed multiplier*	Safety headway (unit: minute)
Trains	TRN_1	E(1021)	N(22)	[0,40]	215	0.7	g = 1 h = 2
	TRN_2	E(1020)	N(22)	[10,50]	225	0.7	
	TRN_3	E(1021)	S(515)	[90,140]	390	0.7	
	TRN_4	E(1020)	S(515)	[50,90]	250	1	
	TRN_5	E(1021)	W(1547)	[40,90]	225	0.7	
	TRN_6†	E(1020)	W(1547)	[80,120]	210	1	
	TRN_7	N(21)	E(1022)	[10,50]	226	0.7	
	TRN_8†	N(21)	E(1022)	[40,80]	256	0.7	
	TRN_9	N(21)	S(515)	[20,60]	250	0.7	
	TRN_10†	N(21)	S(515)	[40,80]	197	1	
	TRN_11	N(21)	S(515)	[60,120]	217	1	
	TRN_12	N(21)	W(1547)	[30,80]	108	0.7	
	TRN_13†	N(21)	W(1547)	[30,70]	84	1	
	TRN_14	S(514)	N(22)	[10,50]	234	0.7	
	TRN_15†	S(514)	N(22)	[10,50]	162	1	
	TRN_16	S(514)	N(22)	[40,80]	264	0.7	
	TRN_17	W(1524)	E(1022)	[10,50]	196	0.7	
	TRN_18†	W(1525)	E(1022)	[20,60]	149	1	
	TRN_19	W(1526)	E(1022)	[30,70]	159	1	
	TRN_20	W(1527)	N(22)	[0,40]	80	0.7	
	TRN_21	W(1528)	N(22)	[30,70]	84	1	
	TRN_22	W(1582)	M(2206)	[0,20]	74	0.7	
	TRN_23	M(2204)	E(1022)	[0,20]	55	1	
	TRN_24	E(1020)	M(2180)	[20,40]	91	0.7	
	TRN_25	M(2182)	W(1547)	[0,15]	87	1	
	TRN_26†	N(21)	M(2077)	[60,90]	167	1	
	TRN_27†	M(2079)	S(515)	[8,28]	115	0.7	
	TRN_28†	S(514)	M(2071)	[35,55]	108	1	
	TRN_29†	M(2068)	N(22)	[5,25]	125	0.7	
	TRN_30	M(2179)	W(1547)	[5,18]	98	1	
	TRN_31	M(2206)	E(1022)	[15,35]	89	0.7	
PMTSs	PMTS_zone1	2249	2162	–	–	–	w = 25
	PMTS_zone2	2161	2248	–	–	–	w = 30
	PMTS_zone3†	2035	2114	–	–	–	w = 35
	PMTS_zone4†	2111	2036	–	–	–	w = 40
Planning horizon T		400		Minimum and maximum dwell time at stations			0 (min), 120 (max)

[*] Note that speed multiplier is used to obtain the actual free-flow running time by multiplying the maximum free-flow running time of the cell and the speed multiplier of the train.

[†] The 21 trains and 2 PMTSs without the † mark are used in the first two parts of Section 4.5.2(2.b), and all 31 trains and 4 PMTSs are considered for the larger-scale experiment in the third part of Section 4.5.2(2.b).

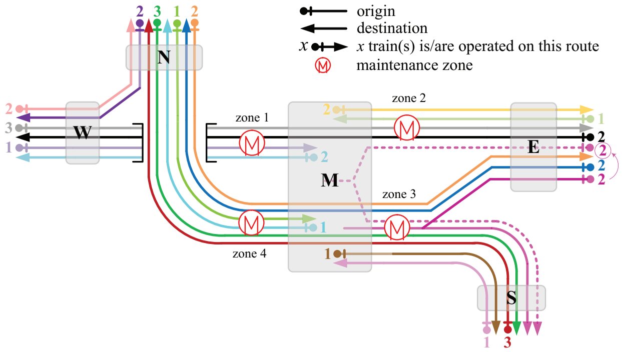


Figure 4.7: Illustration of the O-D pairs of the 31 trains and the workspace of the 4 PMTSs

12.6. We further present an assessment of the Lagrangian-relaxation-based algorithm in Section 4.5.2(2), which is implemented by Visual C++ 2013. Due to the limited applicability of the P_1 problem to large-scale instances, the experiments related to the P_1 problem are only done based on the artificial case study. However, the proposed Lagrangian-relaxation-based algorithm is able to solve large-scale instances, so that both artificial and realistic case studies are used to examine its performance, from the point of view of effectiveness and efficiency.

4.5.2 Experimental results and discussion

(1) Performance of the P_1 problem: experiments based on the artificial network

This section reports the experimental analysis of the integrated optimization problem (P_1), based on the artificial case study introduced in Section 4.5.1(1). The solutions obtained by the sequential scheduling method in an enumerative manner are also provided as benchmarks, in order to demonstrate the benefits of the integrated optimization on train scheduling and PMTS planning.

(1.a) Complexity analysis of the P_1 problem

We first analyze the complexity of the P_1 problem, in terms of the number of variables and constraints. An increasing number of trains (up to 20) and planning horizon (up to 300 minutes) are considered, based on the artificial network of Figure 4.5. As illustrated in Figure 4.8, the number of variables and constraints increase with an increasing number of trains, and planning horizon as well. When considering 20 trains and a 300 minutes planning horizon, the numbers of variables and constraints are 291795 and

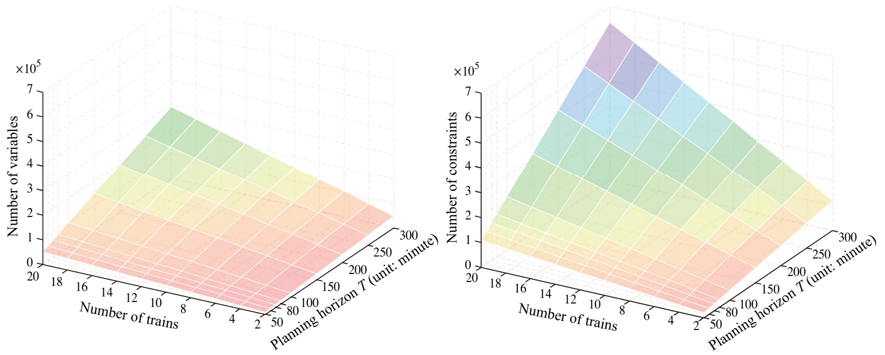


Figure 4.8: Number of variables and constraints of the P_1 problem, corresponding to different planning horizons T and numbers of trains (based on the artificial network of Figure 4.5)

633460 respectively. The number of trains and time horizon affect the complexity of the P_1 problem in a linear manner. This implies that one can improve the computational efficiency by choosing an appropriate planning horizon that only covers the entire paths of all trains and PMTSs involved.

(1.b) Benefits of the integrated optimization on train scheduling and PMTS planning

Due to the limited applicability of the P_1 problem to large-scale instances, we only report the results considering the first 2, 4, 6, and 8 trains listed in Table 4.3 respectively, for different planning horizons (i.e., 50, 60, 70, 80, 100, 150, and 200 minutes).

Several scenarios with different pre-determined PMTSs are considered for the sequential scheduling method, i.e., 22, 16, 21, 26, 24, 31, and 35 scenarios for 50, 60, 70, 80, 100, 150, and 200 minutes planning horizon respectively, as listed in the second row of Table 4.5. By using the model proposed by Meng and Zhou (2014), these scenarios are solved one by one in an ascending order of PMTS starting times, which results in an equivalent iterative process. Moreover, these scenarios have uniformly-spaced starting times for the PMTSs, namely 1, 2, 2, 2, 3, 4, and 5 minutes for the cases with 50, 60, 70, 80, 100, 150, and 200 minutes planning horizon respectively. In each scenario, the PMTSs are assumed to start at the same time on all relevant block sections. For instance, in the case with 50 minutes planning horizon, the starting times of the PMTSs are 0, 1, 2, ..., 21 minutes respectively, and in the case with 200 minutes planning horizon, the starting times are 0, 5, 10, ..., 170 minutes. In the case with 200 minutes planning horizon, 1-minute-spaced starting times of PMTSs leads to a large amount of scenarios (172 scenarios) and a much longer computation time. To reduce the computation time and quickly scan the whole time horizon, a 5-minute interval is instead considered.

The percentages of feasible solutions obtained by the sequential scheduling method are reported in Table 4.5. When considering a small planning horizon (i.e., 50 minutes, which can be viewed as a kind of capacity saturation), not all scenarios are feasible. In fact, only 9.09% of scenarios (2 out of 22) are feasible for the case with 8 trains and a 50 minutes planning horizon. This reflects the drawback of the sequential method: a limited option given in the first PMTSs planning stage could dramatically downgrade

Table 4.5: Feasibility analysis of the sequential scheduling method

Percentage of feasible solutions (unit: %)		Planning horizon (unit: minute)						
		50	60	70	80	100	150	200
Total number of scenarios		22	16	21	26	24	31	35
Number of trains	2	100.00	100.00	100.00	100.00	100.00	100.00	100.00
	4	100.00	100.00	100.00	100.00	100.00	100.00	100.00
	6	27.27	100.00	100.00	100.00	100.00	100.00	100.00
	8	9.09	100.00	100.00	100.00	100.00	100.00	100.00

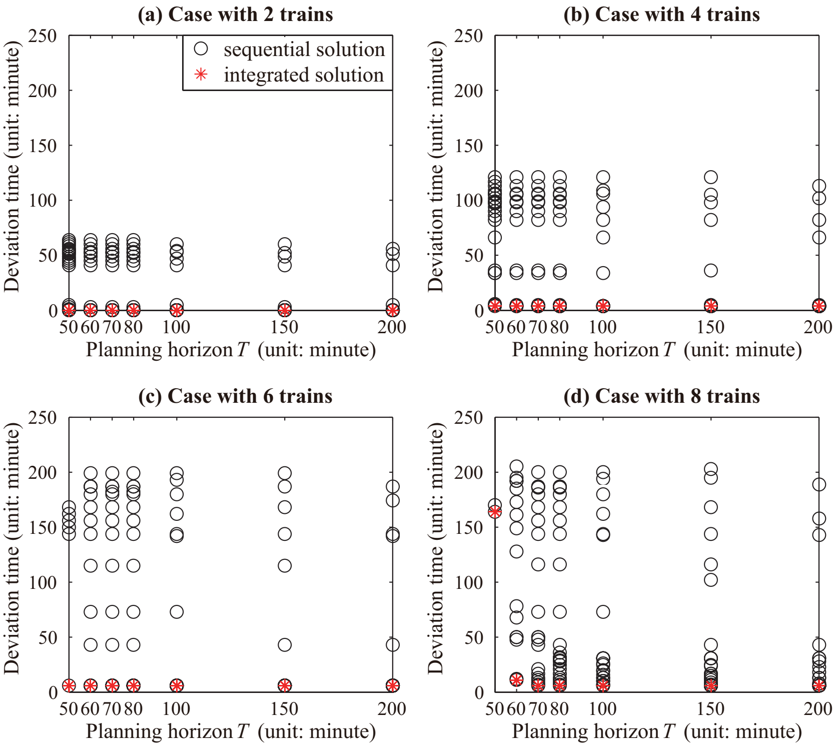


Figure 4.9: Results of the integrated and sequential scheduling methods in different cases

the performance of the second train scheduling solution. Moreover, the available capacity for trains might be reduced by inappropriate PMTS plans in the first stage, so that the planned trains might not be able to be completely scheduled.

Figure 4.9 illustrates the total deviation time of the integrated and sequential scheduling methods for the cases with 2, 4, 6, and 8 trains respectively. Recall that several scenarios with different pre-determined PMTSs are considered for the sequential scheduling method. Each black circle represents the total deviation time of all real trains in a scenario.

As shown in Figure 4.9, in each case, the integrated solutions always achieve the best quality (i.e., the lowest deviation time) comparing with the sequential solutions: *the integrated scheduling method is at least as good as the sequential one*. Moreover, the solution quality can be improved by an increasing planning horizon, due to the adequate capacity. See for instance the case with 8 trains: an increase of the planning horizon from 50 to 60 minutes results in a 153-minute reduction of the total deviation time (reducing it from 164 to 11 minutes). Although a better solution can be found by increasing the planning horizon, a longer planning horizon cannot be always available in real operations. For instance, if the possible planning horizon for the case with 8

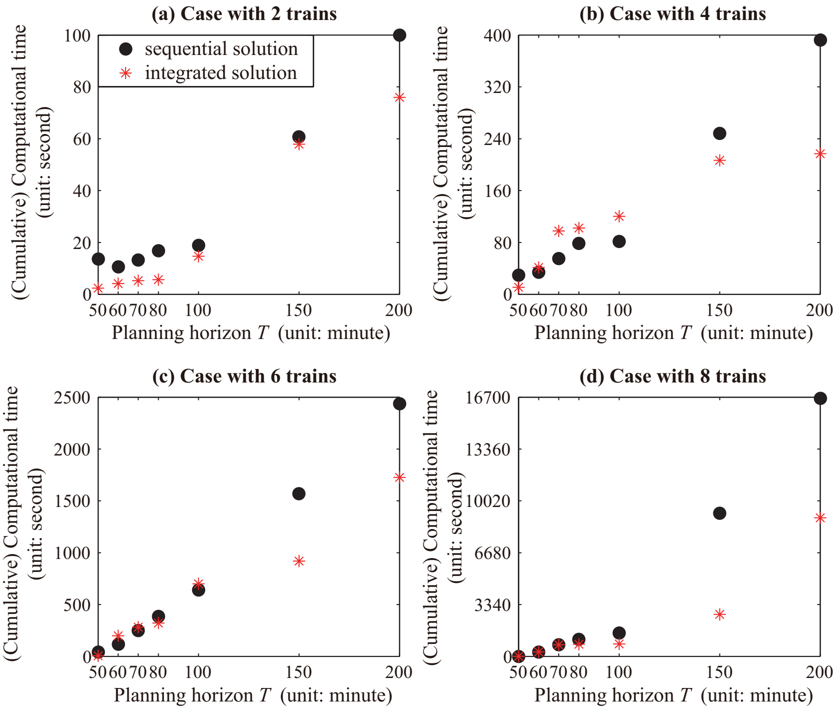


Figure 4.10: (Cumulative) Computation time of the integrated and sequential scheduling methods in different cases

trains is only 50 minutes, the schedule with a 164-minute deviation time of trains is the best one we can apply; if it is possible to increase the planning horizon to 60 minutes, a better schedule with an 11-minute deviation time is available. In real operations, the planning horizon generally increases with an increasing amount of trains. In this chapter we do not focus on determine an appropriate planning horizon; we focus on generate a schedule of trains and PMTSs under a certain condition, namely simultaneously scheduling a certain amount of trains and PMTSs in a certain planning horizon.

Figure 4.10 illustrates the computation time of the integrated solution, and the cumulative computation time of the sequential solutions, corresponding to each case in Figure 4.9. While generating the sequential solutions, an ascending order is considered for the starting times of the pre-determined PMTSs. Moreover, in Figure 4.10, we only cumulate the computation times of the sequential solutions until a solution that has the same deviation time as the integrated solution is found. This helps constructing a readable figure, as the total computation time of all sequential scenarios is too large (up to 36859.27 seconds, 10 hours). Even if counting in such a way, the sequential scheduling method still needs a longer computation time for finding the best solution (the optimal integrated solution) in most cases, among which the largest difference is 7826.36 seconds. However, in some other cases, the computation time of the sequential scheduling method can be 60.62 seconds shorter (at most) than that of the integrated scheduling method. This demonstrates the good performance of the integrated scheduling method on computational efficiency. The best solution cannot be known before solving all the scenarios of the sequential scheduling method and cannot be definitely obtained even when solving all the scenarios. Furthermore, as shown in Figure 4.10, the computation time increases with an increasing planning horizon and an increasing number of trains.

The benefits of the integrated optimization approach are given as follows:

1). A solution with better quality can be obtained efficiently, as the solution space of the integrated optimization problem contains the solution space of the sequential method as a subset;

2). The solving process is simplified and efficient: the global optimal solution can be obtained directly without any enumerative or iterative processes;

3). It is unnecessary to pre-determine PMTS(s), which has a great impact on the feasibility and solution quality.

While the performance of the integrated optimization problem (P_1) is good on small-scale networks, the model complexity limits its scalability and its applicability to large-scale instances: no feasible solution can be obtained by the P_1 problem within a computation time limit, when considering a larger amount of trains (like 20 trains) and a longer time horizon (like 300 minutes). However, the proposed Lagrangian-relaxation-based solution framework can solve such a large-scale problem, as will be shown next.

(2) Performance of the Lagrangian-relaxation-based solution algorithm

This section demonstrates the effectiveness and efficiency of the proposed Lagrangian-relaxation-based solution framework, which includes the time-dependent least-cost path algorithm and the priority-rule-based algorithm. We are mainly interested in solution quality and computation time for the following three types of solution methods.

i). IP-CPLEX, integer programming (IP) implementation of the P1 problem, solved by CPLEX;

ii). LP-CPLEX, linear programming (LP) relaxation of the P1 problem, solved by CPLEX;

iii). LR-C++, IP implementation of the P1 problem, solved by the customized C++ package with the built-in time-dependent least-cost path algorithm in the Lagrangian-relaxation-based solution framework.

We first use the artificial network described in Section 4.5.1(1) as the test bed, to compare the results of the above three types of solution methods. After that, the performance of the proposed Lagrangian-relaxation-based solution algorithm is evaluated on the realistic network described in Section 4.5.1(2). Due to the increasing number of trains used in this section, which leads to a longer computation time, a computation time limit of 10800 seconds (3 hours) is considered for terminating the CPLEX solving process.

(2.a) Evaluation based on the artificial network

- *Effectiveness analysis: lower bound (obtained by LP-CPLEX and LR-C++) and upper bound (obtained by IP-CPLEX and LR-C++)*

In this section, the network in Figure 4.5 with different numbers of trains (range from 2 to 20) is considered, and the planning horizon T is set to 150 minutes. Table 4.6 reports the results (upper bounds) obtained by CPLEX and C++.

The lower bounds of LP-CPLEX and LR-C++ are zero for all cases, so we do not report them in Table 4.6. This phenomenon results from the scale of the experimental case, the pre-defined ideal timetable, and the consideration of a non-negative objective function. According to our experimental experiences, solving a case with a larger number of trains on a large-scale network normally yields better lower bounds. In fact, in the experiments based on the realistic network, we obtain better lower bounds (see below).

Regarding the upper bounds in Table 4.6, we observe that optimal solutions can only be found by IP-CPLEX for the small-scale cases, with no more than 12 trains. When the number of trains is larger than 12, IP-CPLEX cannot obtain any solution within 3 hours

Table 4.6: Upper bound (total deviation time) of IP-CPLEX and LR-C++

Number of trains	Upper bound (unit: minute)		
	IP-CPLEX (optimal)	LR-C++ (feasible)	Optimality gap (%)
2	0	0	0.00
4	4	4	0.00
6	6	6	0.00
8	6	6	0.00
10	6	8	33.33
12	6	8	33.33
14	–	11	–
16	–	16	–
18	–	20	–
20	–	21	–

* Note that optimality gap = (LR-C++ – IP-CPLEX)/IP-CPLEX (%); "–" means that no optimal solution is obtained by IP-CPLEX within 10800 seconds (3 hours), and as a result the corresponding optimality gap measure is not available.

(even no feasible solution is found). The limited applicability of IP-CPLEX on finding solutions for a relatively large network with more trains is evident. In comparison, LR-C++ can find the optimal solutions for the small-scale cases (the cases with no more than 8 trains), and feasible solutions (with an optimality gap of 33.33% for the cases with 10 and 12 trains) within 1 minute (the corresponding computation times are shown in Table 4.7). The applicability of LR-C++ on finding feasible solutions is further confirmed by the realistic network adapted from a Chinese railway network (see below).

- *Efficiency analysis: computation time of LP-CPLEX, IP-CPLEX, and LR-C++*

Table 4.7 shows the computation time of the lower and upper bounds reported in Table 4.6. The computational efficiency of LR-C++ is much better than that of LP-CPLEX and IP-CPLEX (for computing lower bounds and upper bounds respectively), as a longer computation time is required for the CPLEX solving process (up to 822 seconds for the lower bounds and 9232 seconds for the upper bounds). The computation times of LP-CPLEX and IP-CPLEX increase with an increasing number of trains. The upper bound (optimal solution) needs a longer computation time than the lower bound, for all cases (if the optimal solution is available). Note that due to the non-negative property of the objective function, the lower bound can be obtained quickly by LR-C++ (less than 1 second). LR-C++ can obtain an upper bound (a feasible solution) for all cases within 1 minute, at the only cost of a relatively bad solution quality (the optimality gap is given in Table 4.6). For instance, in the case with 10 trains, the optimality gap between IP-CPLEX and LR-C++ is 33.33% (2-minute difference), with a 5691-second reduction of the computation time (from 5739 to 48 seconds).

- *Exploration of the iterative solving process of LR-C++*

Table 4.7: Computation time of the lower bound and upper bound (unit: second)

Number of trains	Computation time of lower bound		Computation time of upper bound	
	LP-CPLEX	LR-C++	IP-CPLEX (optimal)	LR-C++ (feasible)
2	19.51	<1	57.91	2
4	40.85	<1	206.61	8
6	107.91	<1	919.85	9
8	157.32	<1	2703.58	10
10	233.27	<1	5739.08	48
12	345.31	<1	9232.33	13
14	378.45	<1	–	54
16	564.15	<1	–	18
18	671.96	<1	–	19
20	821.80	<1	–	26

* Note that "-" means no optimal solution obtained by IP-CPLEX within 10800 seconds (3 hours).

In this section, we present how real trains and virtual trains (PMTSs) are updated during the iterative solving process of LR-C++. For simplicity, only a small-scale instance is given, i.e., the case with 4 trains, for which the optimal solution can be found by LR-C++ within a displayable amount of iterations. For each iteration, the scheduling sequence, indicators (the arrival time for each real train and the working time window for each PMTS), deviation time of each real train from the ideal timetable, and the local/global upper bounds are provided in Figure 4.11. Recall that, in each iteration, the scheduling sequence is changed by Lagrangian profits, and a feasible solution (upper bound) is generated by applying the time-dependent least-cost path algorithm.

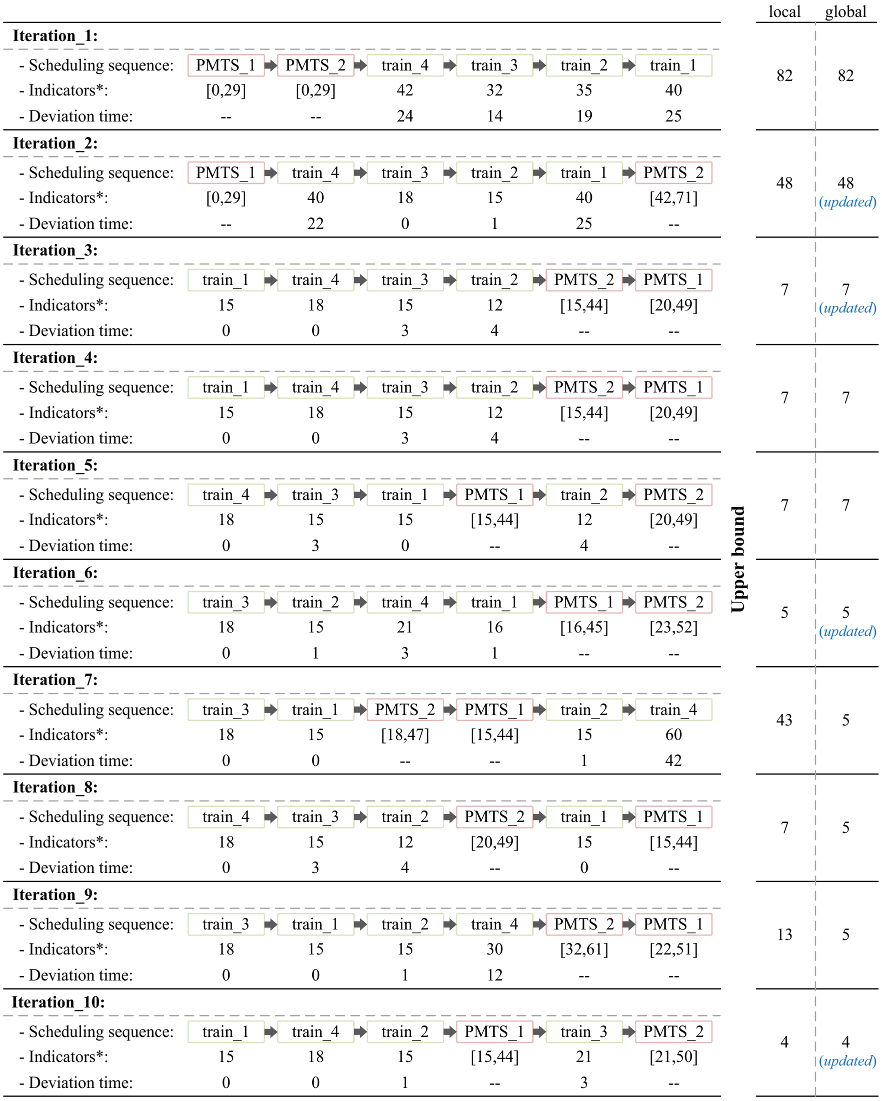
As illustrated, in Iteration_1, two PMTSs are first scheduled, followed by four real trains, which leads to a feasible solution with a total deviation time of 82 seconds. In Iteration_2, a solution with a total deviation time of 48 seconds is found, which is the current best solution; so the global upper bound is updated. This also occurs in Iteration_3, Iteration_6, and Iteration_10, in which the global upper bound is updated to 7, 5, and 4 minutes respectively. Note that due to the competition between PMTS_1 and train_4 for the same infrastructure resources (cells), the deviation time of train_4 is quite large in Iteration_2, even if it is the first scheduled real train. The solution obtained in Iteration_10 is same as the optimal solution found by the P_1 problem.

(2.b) Evaluation based on the realistic network

In this section, we use the large-scale realistic network described in Figure 4.6 to test the performance of LR-C++. The benefits of the integrated scheduling method are first estimated based on the dataset considering 21 trains and 2 PMTSs. A larger amount of trains and PMTSs (31 trains and 4 PMTSs) is further considered to evaluate the performance of LR-C++ on larger scale instances.

- *Benefits of the integrated scheduling method reported by LR-C++*

A comprehensive set of sequential solutions that consider different pre-determined PMTSs are provided as benchmarks, in order to demonstrate the benefits of the integration of train scheduling and PMTSs planning and verify the effectiveness of the proposed algorithm. These sequential solutions are obtained by applying a previous version of the Lagrangian-relaxation-based solution algorithm (proposed by Meng and Zhou, 2014), in which PMTSs are fixed and result in track possessions in a certain time period. Regarding the pre-determined PMTSs, we still follow a similar structure (as that of the previous test cases) that considers a uniformly spaced starting time of the PMTSs. These pre-determined starting times result in several scenarios, and in each scenario the PMTSs are assumed to start at the same time on all relevant block sections. For the realistic dataset, we consider a 10-minute spaced starting time of the PMTSs in a 400-minute planning horizon, which leads to 38 scenarios (the starting time of the PMTSs is 0, 10, ..., 360, 370 minutes respectively in each scenario, as the



							Upper bound	
							local	global
Iteration_1:								
- Scheduling sequence:	PMTS_1	PMTS_2	train_4	train_3	train_2	train_1	82	82
- Indicators*:	[0,29]	[0,29]	42	32	35	40		
- Deviation time:	--	--	24	14	19	25		
Iteration_2:								
- Scheduling sequence:	PMTS_1	train_4	train_3	train_2	train_1	PMTS_2	48	48 *(updated)*
- Indicators*:	[0,29]	40	18	15	40	[42,71]		
- Deviation time:	--	22	0	1	25	--		
Iteration_3:								
- Scheduling sequence:	train_1	train_4	train_3	train_2	PMTS_2	PMTS_1	7	7 *(updated)*
- Indicators*:	15	18	15	12	[15,44]	[20,49]		
- Deviation time:	0	0	3	4	--	--		
Iteration_4:								
- Scheduling sequence:	train_1	train_4	train_3	train_2	PMTS_2	PMTS_1	7	7
- Indicators*:	15	18	15	12	[15,44]	[20,49]		
- Deviation time:	0	0	3	4	--	--		
Iteration_5:								
- Scheduling sequence:	train_4	train_3	train_1	PMTS_1	train_2	PMTS_2	7	7
- Indicators*:	18	15	15	[15,44]	12	[20,49]		
- Deviation time:	0	3	0	--	4	--		
Iteration_6:								
- Scheduling sequence:	train_3	train_2	train_4	train_1	PMTS_1	PMTS_2	5	5 *(updated)*
- Indicators*:	18	15	21	16	[16,45]	[23,52]		
- Deviation time:	0	1	3	1	--	--		
Iteration_7:								
- Scheduling sequence:	train_3	train_1	PMTS_2	PMTS_1	train_2	train_4	43	5
- Indicators*:	18	15	[18,47]	[15,44]	15	60		
- Deviation time:	0	0	--	--	1	42		
Iteration_8:								
- Scheduling sequence:	train_4	train_3	train_2	PMTS_2	train_1	PMTS_1	7	5
- Indicators*:	18	15	12	[20,49]	15	[15,44]		
- Deviation time:	0	3	4	--	0	--		
Iteration_9:								
- Scheduling sequence:	train_3	train_1	train_2	train_4	PMTS_2	PMTS_1	13	5
- Indicators*:	18	15	15	30	[32,61]	[22,51]		
- Deviation time:	0	0	1	12	--	--		
Iteration_10:								
- Scheduling sequence:	train_1	train_4	train_2	PMTS_1	train_3	PMTS_2	4	4 *(updated)*
- Indicators*:	15	18	15	[15,44]	21	[21,50]		
- Deviation time:	0	0	1	--	3	--		

Figure 4.11: Illustration on the iterative solving process of LR-C++ for the case with 4 trains

longest duration of the PMTSs is 30 minutes). However, we only report ten of these possible scenarios (denoted as Scenario_1, ..., Scenario_10). Note that the number of the possible scenarios increases with the increasing instance scale. It is hardly possible to explore all possibilities for a large-scale instance, which would lead to a much longer computation time.

Figure 4.12 provides a summary of the integrated solution and sequential solutions

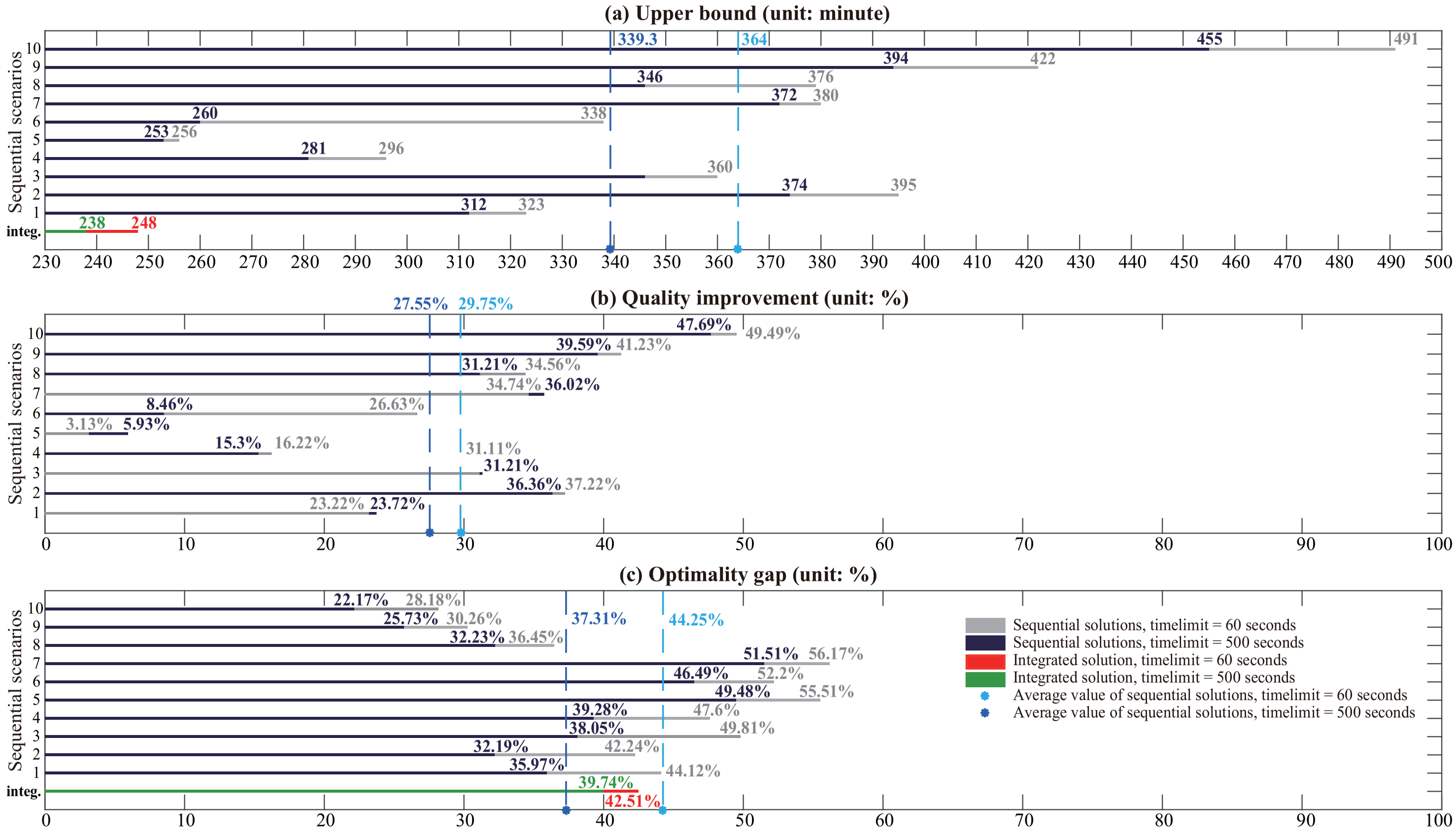


Figure 4.12: Summary of the upper bound, quality improvement, and optimality gap

obtained at 60 and 500 seconds of computation time, reporting the upper bound, quality improvement, and optimality gap respectively. The quality improvement in Figure 4.12(b) indicates how much the integrated solution improves while comparing with the sequential solution in each scenario. The upper bound in Figure 4.12(a) and the optimality gap in Figure 4.12(c) are desired to be as low as possible, while the quality improvement in Figure 4.12(b) is better to be higher. The sequential solution for each scenario obtained at 60 seconds of computation time is presented as a gray bar, and that obtained at 500 seconds is shown as a black bar. The average value of the ten scenarios is given in light blue and dark blue for 60 and 500 seconds respectively. A red/green bar indicates the integrated solution at 60/500 seconds of computation time.

The upper bound of the integrated scheduling method is always better, which achieves 29.75% and 27.55% improvement in average with respect to the sequential solutions at 60 and 500 seconds respectively, as shown in Figure 4.12(b). This demonstrates that the algorithm is able to effectively exploit the larger solution space associated with the integration of maintenance and train operations. The gap of improvement in the results demonstrates the value of the integration of train scheduling and PMTS planning.

The upper bound of the integrated solution found at 60 seconds (248 minutes) is even better than the best upper bound of the ten sequential solutions obtained at 500 seconds (253 minutes in Scenario_5). This implies that the proposed algorithm is efficient, as a solution with good/satisfactory quality (248 minutes) is obtained already at 60 seconds. This solution can be improved further by 10 minutes (from 248 to 238 minutes), if the computation time is extended to 500 seconds.

Moreover, the optimality gap of the integrated solution is reduced from 42.51% to 39.74% with the extra 440-second computation time (from 60 to 500 seconds). The optimality gap of the sequential solutions ranges from 28.18% to 56.17% at 60 seconds of the computation time and 22.17% to 51.51% at 500 seconds. When considering a shorter computation time, i.e., 60 seconds, the optimality gap of the integrated solution (42.51%) is smaller than the average optimality gap of the sequential solutions (44.25%). However, due to the relatively large reduction of the upper bounds of the sequential solutions, the average optimality gap has a larger change of 6.94% (from 44.25% to 37.31%) at 500 seconds. A smaller average gap (37.31%) can be seen in the sequential solutions, compared with that of the integrated solution (39.74%). The smaller average gap of the sequential solutions results from the tight lower bounds obtained, rather than the upper bounds (the obtained feasible solutions). The integrated approach still yields better performance with respect to solution quality.

- *LR-C++ performing on a larger-scale instance*

We next use a larger amount of trains and PMTSs (31 trains and 4 PMTSs in total) on the realistic network to further examine the performance of LR-C++. Figure 4.13 illustrates the upper bound, lower bound, and optimality gap of LR-C++ along the computation time.

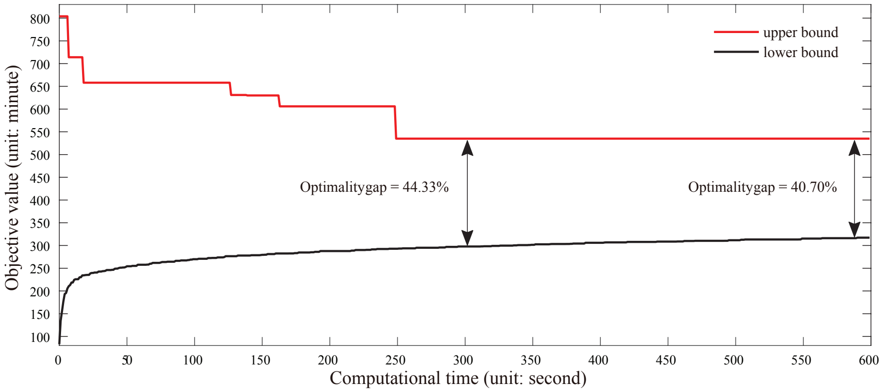


Figure 4.13: Upper bound, lower bound, and optimality gap for the larger-scale instance

As shown in Figure 4.13, for a larger number of trains, LR-C++ can find an upper bound (feasible solution with 535-minute total deviation time) at 300 seconds with an optimality gap 44.33%. By extending the computation time to 600 seconds, the optimality gap is reduced to 40.70%, and the upper bound is still 535 minutes. We can also see that the lower bound and upper bound tend to become better along the computation time. However, the upper bound becomes stable within 300 seconds.

4.6 Conclusions

This chapter has addressed the integrated optimization problem of train scheduling and preventive maintenance time slot planning, by using an innovative virtual-train-based formulation technique and applying a flag-variables-based formulation technique proposed by Meng and Zhou (2014). An integrated optimization problem (P_1) has been developed to deliver a global optimal or satisfactory schedule for both trains and PMTSs with microscopic feasibility details. To solve large-scale problems, a Lagrangian-relaxation-based solution framework has been further proposed, in which the difficult track capacity constraints related to safety operations are dualized to decompose the original complex problem into a sequence of single-train-based subproblems. Each subproblem is solved by a standard label-correcting algorithm for finding the time-dependent least-cost path on a time-space network. A priority-rule-based algorithm has been introduced to transform dual solutions into feasible solutions. The performance of the integrated optimization problem (P_1) and the Lagrangian-relaxation-based solution framework has been assessed on a simple artificial network and a real-world network adapted from a Chinese railway network, from the point of view of effectiveness (more than 25% improvement can be achieved by the integrated scheduling method, as reported in Section 4.5.2(2.b)) and efficiency (a solution with a satisfactory quality can be obtained quickly, at about 60 seconds, see Section 4.5.2(2.b)). The experiments demonstrate that the integrated scheduling method is at least as good as the sequential one and the proposed algorithm is able to exploit the large solution space effectively.

Our future research focuses on the following extensions. First, different optimization or reformulation methods might be developed, which can increase the lower bound and decrease the upper bound of the solutions, in order to reduce the optimality gap and further improve the solution quality. Finally, the relation between maintenance plans and reliability of train services can be defined, by considering the risk associated with delaying maintenance, and be able to (re-)schedule traffic in a closed-loop perspective (Corman and Quaglietta, 2015) or within a robust optimization framework (Meng et al., 2016), in order to further reduce the system cost and achieve the largest economic benefits.

Chapter 5

Integration of traffic control and train control-Part 1: Optimization problems and solution approaches[1]

In this chapter, we study the integration of real-time traffic management and train control by using mixed-integer nonlinear programming (MINLP) and mixed-integer linear programming (MILP) approaches. The optimization approaches in this chapter are developed based on the time-instant formulation method described in Section 2.4.1.

This chapter is organized as follows. In Section 5.1, a detailed introduction of the integrated problem of real-time traffic management and train control is given. Section 5.2 introduces blocking time theory, followed by a problem statement and formulation assumptions in Section 5.3. In Section 5.4, three optimization problems formulating the integration of traffic management and train control are presented. Section 5.5 introduces two solution approaches, i.e., a two-level approach for solving an MINLP problem ($\mathrm{P_{NLP}}$), and a custom-designed two-step method for improving the computational efficiency of the MILP problem ($\mathrm{P_{TSPO}}$). Experimental results based on a real-world railway network are given in Section 5.6 for evaluating the performance of the proposed approaches and investigating the benefits of the integration. Finally, Section 5.7 ends the chapter with conclusions.

5.1 Introduction

Railway transport systems are of crucial importance for the competitiveness of national or regional economy as well as for the mobility of people and goods. To improve reliability of train services and increase satisfaction of customers, many railway

[1] With minor updates, this chapter has been published in "Luan, X., Wang, Y., De Schutter, B., Meng, L., Lodewijks, G., Corman, F. (2018). Integration of real-time traffic management and train control for rail networks-Part 1: Optimization problems and solution approaches. *Transportation Research Part B: Methodological*, 115, 41-71."

infrastructure managers (e.g., Network Rail in United Kingdom and Banedanmark in Denmark) and train operating companies (e.g., V/Line in Australia) have set their own targets for train punctuality, in terms of punctuality rates. Moreover, there have been many projects over the years that have aimed at improving the punctuality of trains, such as the On-Time project (Quaglietta et al., 2016). Policy makers and researchers have been seeking approaches for attaining the punctuality goals.

In real operations, unavoidable perturbations (caused by bad weather, infrastructure failure, extra passenger flow, etc.) often happen and result in delays to the original train timetable, which makes it difficulties to achieve the punctuality goals. When trains are delayed from the normal operation, train dispatchers are in charge of adjusting the impacted train timetables from perturbations (by means of taking proper dispatching measures, e.g., re-timing, re-ordering, and re-routing), so as to reduce potential negative consequences (train delays); train drivers are responsible for controlling the delayed trains (by means of taking proper driving actions, i.e., accelerating, cruising, coasting, and braking) to reach the stations at the times specified by train dispatchers, with the aim of minimizing energy consumption. The problem faced by train dispatchers is well-known as the real-time traffic management problem, and the problem encountered by train drivers is the so-called train control problem. In fact, significant interconnections exist between these two problems, as the traffic-related properties have impact on the train-related properties, and vice versa. Solving the two problems in a sequential way hides the potential improvements in performance of train operations. Better train operations can be potentially achieved by jointly considering the two problems, i.e., (re-)constructing a train timetable in a way that applies different diving actions. However, such a joint consideration leads to a very complex and difficult optimization problem, because not only the timetable should be well-defined for synchronizing the accelerating and braking actions of trains in the same block section, but also the driving actions should be controlled under the speed limits, travel time, and distance constraints (Tuyttens et al., 2013). This is even more critical and difficult for real-time operations. Moreover, the safety headway between two consecutive trains dynamically depends on their real speed and acceleration/deceleration rate. As a result, a prompt and reliable decision-making support tool for both dispatchers and drivers is desired, which requires the integration of a rescheduling optimization with microscopic details and highly accurate real-time train speed trajectory optimization at once.

We therefore address the integration of real-time traffic management and train control by using optimization methods, identifying both traffic-related properties (i.e., a set of times, orders, routes to be followed by trains) and train-related properties (i.e., speed trajectories) at once. To formulate the integrated problem, a mixed-integer non-linear programming (MINLP) problem ($\mathrm{P_{NLP}}$) is first proposed and solved by a two-level approach. An approximation based on piecewise affine functions is applied to the non-linear terms in the $\mathrm{P_{NLP}}$ problem, which results in a mixed-integer linear programming (MILP) problem ($\mathrm{P_{PWA}}$). Furthermore, a preprocessing method for generating the possible train speed profile options (TSPOs) for each train on each block section is considered to reduce the complexity of the problem and to restrict the search only to a subset that allows better energy performance. An MILP problem (P_{TSPO}) is subsequently developed to determine the optimal option with minimum train delays. The two MILP problems are both solved by using an MILP solver, but a custom-designed two-step method is particularly used for the P_{TSPO} problem to speed up the solving procedure. In our optimization problems, the blocking time of a train on a block section dynamically depends on its real speed. We consider the minimization of the total train delay times as the objective. According to the experimental results, the proposed approach can obtain feasible solutions (with good quality) of the integrated traffic management and train control problem for a single direction along a 50 km corridor with 9 stations and 15 trains each hour within 3 minutes of computation time, while achieving the goal of reducing train delays by managing the train speed. In Chapter 6, we will further discuss energy-related extensions based on the proposed optimization approaches, i.e., evaluating energy consumption and computing regenerative energy utilization. With the inclusion of the energy-related aspects, we aim at both delay recovery and energy efficiency, in order to achieve energy-efficient train operation.

5.2 Blocking time theory

The safety headway time is the time interval between two following trains and the minimum headway depends on the so-called "blocking time" (Pachl, 2009). The blocking time is the duration of the time interval in which a section of track (usually a block section) is exclusively allocated to a train and therefore blocked for other trains. Thus, the blocking time runs from the moment of issuing a train movement authorization (e.g., by clearing a signal) to the moment that it becomes possible to issue a movement authorization to another train to enter that same section. The blocking time of a block section is usually much longer than the time that the train occupies the block section. Figure 5.1(a) and Figure 5.1(b) illustrate the blocking time of a block section for a train without and with a scheduled stop respectively.

Pachl (2009) defined the components of the blocking time illustrated in Figure 5.1 as follows: 1) the *setup time* is the time duration for clearing the signal before the arrival of a train; 2) the *sight and reaction time* is the time duration for the driver to view the signal; 3) the *approach time* is the time duration for train running over the preceding block section (from the approach signal to the block signal); 4) the *running time* is the time duration for a train to run on the block section; 5) the *clearing time* is the time duration to clear the block section and the overlap with the full length of the train (if required) after the departure of a train; 6) the *release time* is used to unlock the safety block system. Note that the six components of the blocking time are all time durations, the former three terms are used for pre-blocking a block section, and the latter two terms are for post-releasing a block section. Based on the explanations, the approach time and the clearing time strongly depend on the train characteristics (e.g., train speed

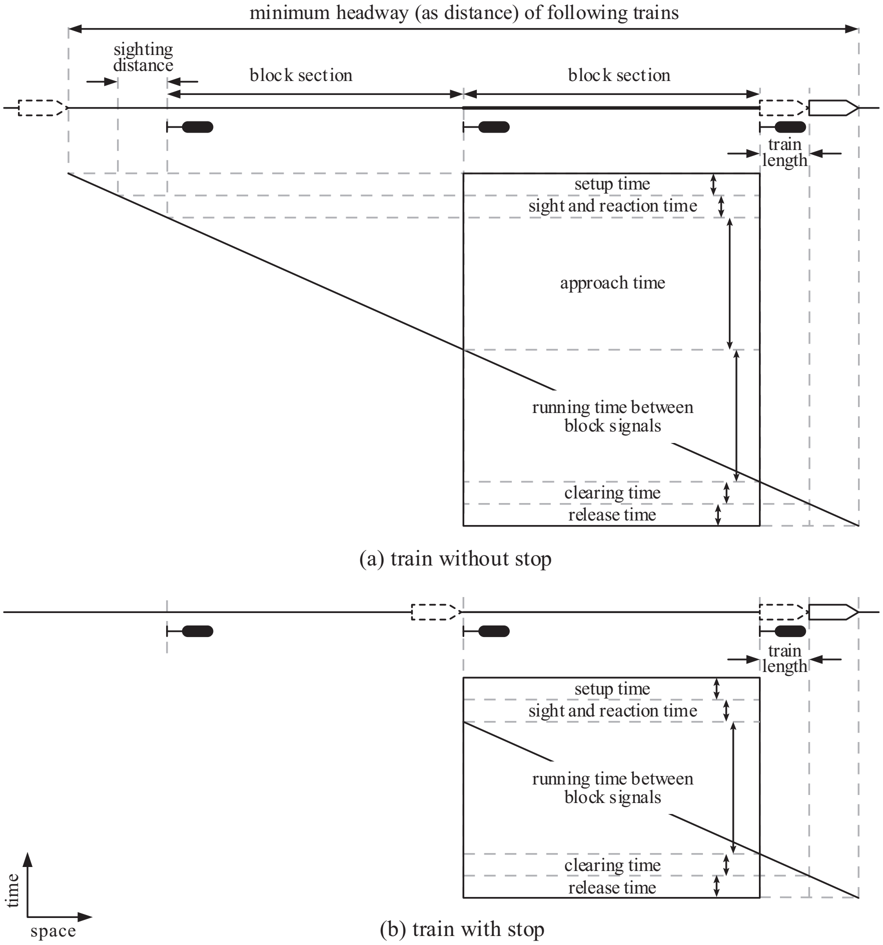


Figure 5.1: The blocking time of a block section for a train without/with a scheduled stop

and train length) and the rail network conditions (e.g., the length of the block section); therefore, they are considered as decision variables and the others (e.g., the setup time) are regarded as constant in this research.

5.3 Problem statement and formulation assumptions

Given a railway network with the technical and operational requirements of stations and segments (e.g., lengths of block sections, speed limitations, and allowed/forbidden dwelling events), a set of trains from pre-specified origins to pre-specified destinations and with pre-specified train characteristics (e.g., length, speed limitation, acceleration, and deceleration), the statement of the integrated traffic management and train control problem is to determine the routes, orders, arrival times, and departure times of the trains at passing stations by finding the optimal train speed profiles, in order to reduce the train delay, and at the same time to reduce the energy for accelerating and re-accelerating caused by unnecessary braking.

We focus on the investigation of the traffic operations. Thus, when constructing the formulations, we emphasize in detail the operational aspect of the traffic and consider the train control aspect with relatively less accuracy in computing the energy consumption (at least, compared with the studies only focusing on train trajectory optimization). In fact, what we target is not to take decisions to change the cruising speed of trains (as it may result in lots of delays due to the high dependence among trains), or to exploit running time buffers to save energy (which can be done by focusing on a single train at a time only, computing ahead of time), but mostly by avoiding unnecessary acceleration and deceleration due to interaction of traffic. We construct and reschedule the train timetable by optimizing the train accelerating and braking actions. Therefore, in our optimization problems, we make the following assumptions: 1) train acceleration is considered as a piecewise constant function of speed by giving a fixed switching point (breakpoint) of speed (e.g., 60 km/h) for each train category; 2) train deceleration is constant for a certain train category and differs among train categories; 3) the speed limit is considered as constant for a certain train category on a certain block section, i.e., the minimum value of the designed train speed and the designed block section (track) speed, but may differ among train categories and block sections; 4) the beginning/end point of a block section or of a main/siding track in a station, or a point of merging/diverging of tracks on a segment, is represented by a node; 5) a block section is described as a cell, which connects two nodes in a pair; 6) a station is simplified to a number of main/siding track(s), which can be further modeled as a single cell or a set of cells; 7) station platforms are placed at the end of cells for trains to stop; 8) for a double-track railway segment between two stations, each track is modeled as a sequence of directional cells (i.e., directional block sections), and for a single-track railway segment, the only track between two stations is modeled as bi-directional cells (i.e., bi-directional block section); 9) the speed of a train on a cell is divided into three phases, i.e., incoming, cruising, and outgoing phases, and train coasting is neglected (however, a coasting phase can be introduced by assuming a piecewise constant deceleration function of the cruising speed, as discussed in Chapter 6); 10) the resistances caused by air, roll, track grade, curves, and tunnels are not considered in this part, but they are included in Chapter 6 while evaluating energy consumption, i.e., the energy consumed for overcoming resistance in accelerating, cruising, and decelerating is computed in Chapter 6; 11) only one train is allowed to access a cell at any time; 12) the granularity of time is one second. Note that the maximum acceleration and deceleration depend on the traction and braking force. In the literature, researchers either consider tractive force as a precise function of speed and control (Howlett, 2000), or assume a constant power (then tractive force is a function of speed, e.g., Howlett, 2000), or assume a constant acceleration (Wang et al., 2016).

5.4 Mathematical formulation

In this section, three optimization approaches are proposed to address the integration of traffic management and train control, i.e., an MINLP approach ($\mathrm{P_{NLP}}$) presented in Section 5.4.1, an MILP approach ($\mathrm{P_{PWA}}$) obtained by approximating the nonlinear terms with PWA functions in Section 5.4.2, and another MILP approach ($\mathrm{P_{TSPO}}$) considering multiple TSPOs generated in a preprocessing step (Section 5.4.3).

5.4.1 Formulation of the $\mathrm{P_{NLP}}$ problem

Table 5.1 lists the sets, subscripts, input parameters, and decision variables used by the $\mathrm{P_{NLP}}$ problem.

Table 5.1: Sets, subscripts, input parameters, and decision variables

Symbol	Description
	Subscripts and sets
F	set of trains, $\lvert F\rvert$ is the number of trains
V	set of nodes, $\lvert V\rvert$ is the number of nodes
E	set of cells, i.e., block sections, $E \subseteq V \times V$, $\lvert E\rvert$ is the number of cells
f	train index, $f \in F$
p,i,j,k	node index, $p,i,j,k \in V$
e	cell index, denoted by (i,j), $e \in E$
E_f	set of cells (or sections) that train f may use, $E_f \subseteq E$
E_f^{stop}	set of cells in which train f should stop, $E_f^{\mathrm{stop}} \subseteq E_f$, $\lvert E_f^{\mathrm{stop}}\rvert$ is the number of stops of train f
	Input parameters
o_f	origin node of train f
s_f	destination node of train f
L_f^{train}	length of train f
c_f^{pri}	primary delay time of train f at its origin node
c_f	planned departure time of train f at its origin node
ρ_f	direction of train f
v_f^{turn}	the train speed at the switching point of acceleration for train f
v^{mincru}	the minimum cruising speed for each train on each cell
v_i^{nlim}	train speed limitation at node i
$v_{i,j}^{\mathrm{clim}}$	train speed limitation on cell (i,j)
$L_{i,j}^{\mathrm{cell}}$	length of cell (i,j)
$A_{f,i,j}$	planned arrival time of train f on cell (i,j), $(i,j) \in E_f^{\mathrm{stop}}$

continued on next page

continued from previous page

Symbol	Description
$w_{f,i,j}^{\min}$	minimum dwell time of train f on cell (i,j)
$w_{f,i,j}^{\max}$	maximum dwell time of train f on cell (i,j)
$\alpha_{1,f,i,j}$	maximum acceleration of train f on cell (i,j), when the train speed is not larger than v_f^{turn}
$\alpha_{2,f,i,j}$	maximum acceleration of train f on cell (i,j), when the train speed is larger than v_f^{turn}
$\beta_{f,i,j}$	the maximum deceleration of train f on cell (i,j)
$\tau_{f,i,j}^{\text{setup}}$	setup time for clearing and setting cell (i,j) when train$f$ is approaching
$\tau_{f,i,j}^{\text{sight}}$	sight time, i.e., running time over a sight distance when train f is approaching cell (p,i), where cell (p,i) is the preceding cell of cell (i,j)
$\tau_{f,i,j}^{\text{reaction}}$	reaction time of train f's driver while approaching cell (i,j)
$\tau_{f,i,j}^{\text{release}}$	release time for releasing cell (i,j) after the clearance of train f
M/ε	a sufficiently large/small positive number
	Decision variables
$a_{f,i,j}$	arrival time of train f at cell (i,j)
$d_{f,i,j}$	departure time of train f at cell (i,j)
$a_{f,i,j}^{\text{turn}}$	time point that train f reaches the switching speed v_f^{turn} in the incoming phase on cell (i,j)
$d_{f,i,j}^{\text{turn}}$	time point that train f reaches the switching speed v_f^{turn} in the outgoing phase on cell (i,j)
$a_{f,i,j}^{\text{cru}}$	time point that train f starts cruising, i.e., the starting time of cruising phase on cell (i,j)
$d_{f,i,j}^{\text{cru}}$	time point that train f ends cruising, i.e., the end time of cruising phase on cell (i,j)
$v_{f,i,j}^{\text{in}}$	incoming speed of train f on cell (i,j)
$v_{f,i,j}^{\text{cru}}$	cruising speed of train f on cell (i,j)
$v_{f,i,j}^{\text{out}}$	outgoing speed of train f on cell (i,j)
$\theta_{f,f',i,j}$	binary train ordering variables, $\theta_{f,f',i,j} = 1$ if train f' arrives at cell (i,j) after train f, and otherwise $\theta_{f,f',i,j} = 0$
$w_{f,i,j}$	dwell time of train f on cell (i,j)
$\tau_{f,i,j}^{\text{approach}}$	approach time of train f on cell (i,j), i.e., running time of train f on the preceding cell (p,i)
$\tau_{f,i,j}^{\text{clear}}$	clearing time for clearing cell (i,j) with the length of train f
$g_{f,i,j}$	safety time interval between occupancy of cell (i,j) and arrival of train f
$h_{f,i,j}$	safety time interval between departure of train f and release of cell (i,j)
$\sigma_{f,i,j}$	occupancy time of cell (i,j) for train f

continued on next page

continued from previous page

Symbol	Description
$\delta_{f,i,j}$	release time of cell (i,j) for train f
$\Theta^{\text{in}}_{f,i,j}$	energy consumption of train f caused by traction force, represented by the difference of the squared speeds in the incoming phase on cell (i,j)
$\Theta^{\text{out}}_{f,i,j}$	energy consumption of train f caused by traction force, represented by the difference of the squared speeds in the outgoing phase on cell (i,j)
$\zeta_{1,f,i,j}$,..., $\zeta_{6,f,i,j}$	logical variables to indicate the relation of the incoming, cruising, outgoing speed, and switching speed v^{turn}_{f}, for train f on cell (i,j), as explained in Table 5.2

Three types of variables are used to formalize the traffic and train related decisions: time variables a and d, speed variables v, and train order variables θ. The other variables are a consequence of the interactions among these variables for all trains in the network, with respect to the formulas of the uniformly accelerating and decelerating motions, definition of the blocking time, and safety requirements.

Figure 5.2 illustrates the relevant variables of train f on two adjacent cells, namely cell (i,j) and cell (j,k). The trajectory of train f on each cell is divided into three phases: incoming, cruising, and outgoing phases. As illustrated in Figure 5.2, train f enters cell (i,j) at time $a_{f,i,j}$ with a speed $v^{\text{in}}_{f,i,j}$, and then a sequence of the following actions is taken on cell (i,j):

1) in the time interval $[a_{f,i,j}, a^{\text{turn}}_{f,i,j}]$, the train accelerates from speed $v^{\text{in}}_{f,i,j}$ to speed v^{turn}_{f} at a steady acceleration $\alpha_{1,f,i,j}$;

2) in the time interval $[a^{\text{turn}}_{f,i,j}, a^{\text{cru}}_{f,i,j}]$, the train accelerates from speed v^{turn}_{f} to speed $v^{\text{cru}}_{f,i,j}$ at a steady acceleration $\alpha_{2,f,i,j}$;

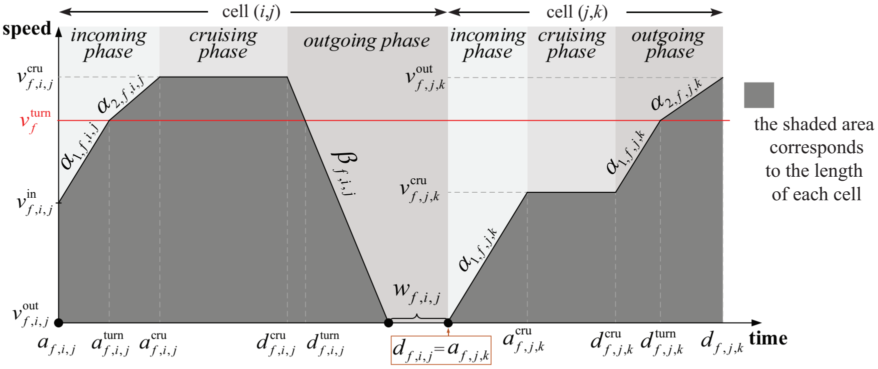


Figure 5.2: Speed-time graph of train f on cell (i,j) and cell (j,k) to illustrate the relevant decision variables

3) in the time interval $[a^{\text{cru}}_{f,i,j}, d^{\text{cru}}_{f,i,j}]$, the train keeps a constant speed $v^{\text{cru}}_{f,i,j}$;

4) in the time interval $[d^{\text{cru}}_{f,i,j}, d_{f,i,j} - w_{f,i,j}]$, the train decelerates from speed $v^{\text{cru}}_{f,i,j}$ to speed $v^{\text{out}}_{f,i,j}$ (taken as 0 km/h in Figure 5.2) at a steady deceleration $-\beta_{f,i,j}$;

5) in the time interval $[d_{f,i,j} - w_{f,i,j}, d_{f,i,j}]$, the train dwells in cell (i, j).

Then, train f departs from cell (i, j) at time $d_{f,i,j}$. Meanwhile, train f arrives at cell (j,k) at time $a_{f,j,k}$, and starts accelerating. As train f does not reach the switching speed v^{turn}_{f} in the incoming phase of cell (j,k), only one acceleration $\alpha_{1,f,i,j}$ is used. Note that the action(s) taken by a train on a cell does not follow a pre-specified sequence (like the one described above); in fact, it is determined by optimizing the time variables (a/d) and speed variables (v). For instance, a train may take a sequence of actions to first accelerate and then decelerate on a cell (i.e., $v^{\text{in}}_{f,i,j} < v^{\text{cru}}_{f,i,j}$ and $v^{\text{out}}_{f,i,j} < v^{\text{cru}}_{f,i,j}$), and it may also take only one action to keep a constant speed traversing the cell (i.e., $v^{\text{in}}_{f,i,j} = v^{\text{cru}}_{f,i,j} = v^{\text{out}}_{f,i,j}$). All possible train trajectories in the incoming and outgoing phases are intuitively provided and explained in Table A.1 of Appendix A.1.1.

We next formulate the integrated traffic management and train control problem. As commonly used in train dispatching optimization problems, each train is assigned a planned arrival time at each planned stop. In the objective function, we minimize the sum over all trains of the mean absolute delay time at all visited stations, i.e., we minimize the deviation from the planned train timetable:

$$\min Z = \sum_{f \in F} \sum_{(i,j) \in E^{\text{stop}}_{f}} \frac{\left| d_{f,i,j} - w_{f,i,j} - A_{f,i,j} \right|}{\left| E^{\text{stop}}_{f} \right|}, \tag{5.1}$$

The train speed consistency constraint

$$v^{\text{out}}_{f,i,j} = v^{\text{in}}_{f,j,k}, \quad \forall f \in F, j \neq o_f, (i,j) \in E_f, (j,k) \in E_f \tag{5.2}$$

ensures the consistency of the train speed between two adjacent cells, i.e., the incoming speed of train f on cell (j,k) equals its outgoing speed on the preceding cell (i, j).

A set of train speed limitation constraints is presented, in which

$$v^{\text{in}}_{f,o_f,j} = 0, \quad \forall f \in F, \left(o_f, j\right) \in E_f, \tag{5.3}$$

$$v^{\text{out}}_{f,j,s_f} = 0, \quad \forall f \in F, \left(j, s_f\right) \in E_f \tag{5.4}$$

guarantee that trains stop at their origins and destinations respectively, i.e., the incoming speed of the origin cell (o_f, j) and the outgoing speed of the destination cell (j, s_f) is zero, and

$$0 \leq v^{\text{in}}_{f,i,j} \leq v^{\text{nlim}}_{i}, \quad \forall f \in F, (i, j) \in E_f, \tag{5.5}$$

$$0 \leq v^{\text{out}}_{f,i,j} \leq v^{\text{nlim}}_{j}, \quad \forall f \in F, (i, j) \in E_f, \tag{5.6}$$

$$v^{\text{mincru}} \leq v^{\text{cru}}_{f,i,j} \leq v^{\text{clim}}_{i,j}, \quad \forall f \in F, (i, j) \in E_f \tag{5.7}$$

ensure that train speed cannot exceed the given speed limitation at each node and on each cell.

The constraint

$$a_{f,i,j} \leq a^{\text{turn}}_{f,i,j} \leq a^{\text{cru}}_{f,i,j} \leq d^{\text{cru}}_{f,i,j} \leq d^{\text{turn}}_{f,i,j} \leq d_{f,i,j} - w_{f,i,j}, \quad \forall f \in F, (i,j) \in E_f \tag{5.8}$$

ensures a proper sequence of the multiple events of train f on cell (i,j), e.g., the train arrival, cruising, and departure occur in sequence.

The cell-to-cell transition constraint

$$d_{f,i,j} = a_{f,j,k}, \quad \forall f \in F, (i,j) \in E_f, (j,k) \in E_f \tag{5.9}$$

enforces the transition time between two adjacent cells, i.e., the departure time of train f on the preceding cell (i,j) equals the arrival time of train f on the successive cell (j,k), if two adjacent cells (i,j) and (j,k) are used consecutively by train f.

The earliest departure time constraint

$$a_{f,o_f,j} \geq c_f + c^{\text{pri}}_f, \quad \forall f \in F, (o_f, j) \in E_f \tag{5.10}$$

ensures that trains do not leave their origins before the earliest departure time, i.e., the sum of the planned departure time and the primary delay time.

A set of train dwell time constraints is considered, in which

$$w^{\text{min}}_{f,i,j} \leq w_{f,i,j} \leq w^{\text{max}}_{f,i,j}, \quad \forall f \in F, (i,j) \in E_f \tag{5.11}$$

guarantees the required minimum and maximum dwell times at stations, and

$$\begin{cases} v^{\text{out}}_{f,i,j} = 0, & \text{if } w_{f,i,j} > 0 \\ v^{\text{out}}_{f,i,j} > 0, & \text{if } w_{f,i,j} = 0 \end{cases}, \quad \forall f \in F, (i,j) \in E_f \tag{5.12}$$

links the outgoing speed variables $v^{\text{out}}_{f,i,j}$ and the dwell time variables $w_{f,i,j}$. The maximum dwell time is used to avoid forbidden dwell events of trains. If a train is allowed to stop on a block section (in a general case), then the corresponding maximum dwell time is set to be sufficiently large; if a train is required to not stop on some particular block sections, then the maximum dwell times on these particular block sections are set to be zero. In (5.12), if train f stops on cell (i,j), i.e., the dwell time $w_{f,i,j}$ is larger than zero, then the corresponding outgoing speed $v^{\text{out}}_{f,i,j}$ equals zero; otherwise, $v^{\text{out}}_{f,i,j}$ should be larger than zero. Note that constraint (5.12) is an "if-then" constraint, which can be rewritten as mixed-integer linear constraints by applying the transformation properties in Williams (2013), which will be introduced in Section 5.4.2. We assume that station platforms for trains to stop are always placed at the end of block sections.

The cell length constraints can be written as

$$L^{\text{cell}}_{i,j} = L^{\text{in}}_{f,i,j} + L^{\text{cru}}_{f,i,j} + L^{\text{out}}_{f,i,j}, \quad \forall f \in F, (i,j) \in E_f, \tag{5.13}$$

where $L^{\text{in}}_{f,i,j}$, $L^{\text{cru}}_{f,i,j}$, and $L^{\text{out}}_{f,i,j}$ indicate the distance that train f runs through on cell (i,j) in the incoming, cruising, and outgoing phases respectively; these distances are given by the following equations:

$$L^{\text{in}}_{f,i,j} = \begin{cases} \frac{1}{2}\left(v^{\text{in}}_{f,i,j} + v^{\text{turn}}_f\right)\left(a^{\text{turn}}_{f,i,j} - a_{f,i,j}\right) \\ \quad +\frac{1}{2}\left(v^{\text{turn}}_f + v^{\text{cru}}_{f,i,j}\right)\left(a^{\text{cru}}_{f,i,j} - a^{\text{turn}}_{f,i,j}\right), \text{ if } v^{\text{in}}_{f,i,j} \leq v^{\text{turn}}_f \leq v^{\text{cru}}_{f,i,j} \\ \frac{1}{2}\left(v^{\text{in}}_{f,i,j} + v^{\text{cru}}_{f,i,j}\right)\left(a^{\text{cru}}_{f,i,j} - a_{f,i,j}\right), \text{ otherwise} \end{cases} \tag{5.14a}$$

$$L^{\text{cru}}_{f,i,j} = v^{\text{cru}}_{f,i,j} \cdot \left(d^{\text{cru}}_{f,i,j} - a^{\text{cru}}_{f,i,j}\right), \tag{5.14b}$$

$$L^{\text{out}}_{f,i,j} = \begin{cases} \frac{1}{2}\left(v^{\text{turn}}_{f} + v^{\text{out}}_{f,i,j}\right)\left(d_{f,i,j} - w_{f,i,j} - d^{\text{turn}}_{f,i,j}\right) & \\ \quad +\frac{1}{2}\left(v^{\text{cru}}_{f,i,j} + v^{\text{turn}}_{f}\right)\left(d^{\text{turn}}_{f,i,j} - d^{\text{cru}}_{f,i,j}\right), \text{if } v^{\text{cru}}_{f,i,j} \leq v^{\text{turn}}_{f} \leq v^{\text{out}}_{f,i,j} & \\ \frac{1}{2}\left(v^{\text{cru}}_{f,i,j} + v^{\text{out}}_{f,i,j}\right)\left(d_{f,i,j} - w_{f,i,j} - d^{\text{cru}}_{f,i,j}\right), \text{ otherwise} & \end{cases} \tag{5.14c}$$

These equations derive from the basic formulas of uniformly accelerating or decelerating motions, i.e., for such a motion with an initial speed v_o, a final speed v_t and an elapsed time Δt, the distance traveled is $L = \frac{v_0+v_t}{2} \cdot \Delta t$. Note that the distance $L^{\text{cell}}_{i,j}$ that train f runs over on cell (i, j) equals the length of cell (i, j) and corresponds to the shaded area in Figure 5.2. Constraints (5.14a)-(5.14c) are nonlinear, due to the nonlinear dynamics of time, speed, and distance.

The approach time and clearing time constraints can be written as

$$\tau^{\text{approach}}_{f,j,k} = \begin{cases} 0, & \text{if } w_{f,i,j} > 0 \\ d_{f,i,j} - a_{f,i,j}, & \text{if } w_{f,i,j} = 0 \end{cases}, \ \forall f \in F, (i, j) \in E_f, (j,k) \in E_f, \tag{5.15}$$

$$\tau^{\text{clear}}_{f,p,i} = 2 \cdot L^{\text{train}}_{f} \Big/ (v^{\text{out}}_{f,p,i} + v^{\text{cru}}_{f,i,j}), \quad \forall f \in F, (p,i) \in E_f, (i, j) \in E_f. \tag{5.16}$$

These two constraints are also nonlinear. In (5.15), if train f does not stop on the preceding cell (i, j), the approach time of train f on cell (j,k) equals its running time on the preceding cell (i, j); otherwise, the approach time of train f on cell (j,k) equals zero. The clearing time of train f on cell (p,i) is determined in (5.16) according to its incoming and cruising speed on the successive cell (i, j). However, constraint (5.16) may cause an error if the actual train speed (when the train tail leaves the preceding block section) is much smaller than the cruising speed $v^{\text{cru}}_{f,i,j}$ of the train on the successive cell (i, j). To solve this issue, we can formulate the clearing time $\tau^{\text{clear}}_{f,p,i}$ as a piecewise constant function of the outgoing speed $v^{\text{out}}_{f,p,i}$, as follows:

$$\tau^{\text{clear}}_{f,p,i} = \begin{cases} C_{f,1} \cdot v^{\text{out}}_{f,p,i}, & \text{if } v^{\text{out\_bk}}_{1,f} \leq v^{\text{out}}_{f,p,i} \leq v^{\text{out\_bk}}_{2,f} \\ C_{f,2} \cdot v^{\text{out}}_{f,p,i}, & \text{if } v^{\text{out\_bk}}_{2,f} \leq v^{\text{out}}_{f,p,i} \leq v^{\text{out\_bk}}_{3,f} \\ C_{f,3} \cdot v^{\text{out}}_{f,p,i}, & \text{if } v^{\text{out\_bk}}_{3,f} \leq v^{\text{out}}_{f,p,i} \leq v^{\text{out\_bk}}_{4,f} \end{cases}, \tag{5.17}$$

where $C_{f,1}$, $C_{f,2}$, and $C_{f,3}$ are pre-defined clearing times for train f and $v^{\text{out\_bk}}_{1,f}$, ..., $v^{\text{out\_bk}}_{4,f}$ are relevant coefficients regarding the piecewise constant line segments. Constraints (5.15) and (5.17) are "if-then" constraints, which can be rewritten as mixed-integer linear constraints by applying the transformation properties in Williams (2013), which will be introduced in Section 5.4.2.

A set of equations is now proposed for determining the safety time interval illustrated in Figure 5.1:

$$g_{f,i,j} = \tau^{\text{setup}}_{f,i,j} + \tau^{\text{sight}}_{f,i,j} + \tau^{\text{reaction}}_{f,i,j} + \tau^{\text{approach}}_{f,i,j}, \quad \forall f \in F, (i, j) \in E_f \tag{5.18}$$

defines the safety time interval between cell occupancy and train arrival, including the setup time $\tau^{\text{setup}}_{f,i,j}$, the sight time $\tau^{\text{sight}}_{f,i,j}$, the reaction time $\tau^{\text{reaction}}_{f,i,j}$, and the approach time $\tau^{\text{approach}}_{f,i,j}$, and

$$h_{f,i,j} = \tau^{\text{release}}_{f,i,j} + \tau^{\text{clear}}_{f,i,j}, \quad \forall f \in F, (i, j) \in E_f \tag{5.19}$$

Table 5.2: Explanation of the speed indicators $\zeta_{1,f,i,j},\ldots,\zeta_{6,f,i,j}$ for train f on cell (i,j)

	Incoming phase			Outgoing phase		
Speed conditions	$v^{\text{in}}_{f,i,j} \le v^{\text{cru}}_{f,i,j}$	$v^{\text{turn}}_{f} \le v^{\text{in}}_{f,i,j}$	$v^{\text{cru}}_{f,i,j} \le v^{\text{turn}}_{f}$	$v^{\text{cru}}_{f,i,j} \le v^{\text{out}}_{f,i,j}$	$v^{\text{turn}}_{f} \le v^{\text{cru}}_{f,i,j}$	$v^{\text{out}}_{f,i,j} \le v^{\text{turn}}_{f}$
	$\Updownarrow$	$\Updownarrow$	$\Updownarrow$	$\Updownarrow$	$\Updownarrow$	$\Updownarrow$
Speed indicators	$\zeta_{1,f,i,j}=1$	$\zeta_{3,f,i,j}=1$	$\zeta_{4,f,i,j}=1$	$\zeta_{2,f,i,j}=1$	$\zeta_{5,f,i,j}=1$	$\zeta_{6,f,i,j}=1$

calculates the safety time interval between train departure and cell release, including the release time $\tau^{\text{release}}_{f,i,j}$ and the clearing time $\tau^{\text{clearing}}_{f,i,j}$.

Then, the cell occupancy and cell release times, i.e., the blocking time for train f traversing cell (i,j), can be respectively written as

$$\sigma_{f,i,j} = a_{f,i,j} - g_{f,i,j}, \quad \forall f \in F, (i,j) \in E_f, \tag{5.20}$$

$$\delta_{f,i,j} = d_{f,i,j} + h_{f,i,j}, \quad \forall f \in F, (i,j) \in E_f. \tag{5.21}$$

The constraint

$$\theta_{f,f',i,j} + \theta_{f',f,i,j} = 1, \quad \forall f \in F, f' \in F, (i,j) \in E_f, (i,j) \in E_{f'} \tag{5.22}$$

indicates that either train f' arrives at cell (i,j) after train f or train f arrives at cell (i,j) after train f'.

Recall that as cells can be bi-directional, trains can use the same cell in different directions, i.e., it is possible to use cell (i,j) and (j,i). Based on the restriction of the train orders in (5.22), the cell capacity constraints can be written as

$$\begin{aligned} &\sigma_{f',i,j} + \left(1-\theta_{f,f',i,j}\right)\cdot M \ge \delta_{f,i,j}, \\ &\qquad \forall f \in F, f' \in F, f \neq f', \rho_f = \rho_{f'}, (i,j) \in E_f, (i,j) \in E_{f'}, \end{aligned} \tag{5.23}$$

$$\begin{aligned} &\sigma_{f',j,i} + \left(1-\theta_{f,f',i,j}\right)\cdot M \ge \delta_{f,i,j}, \\ &\qquad \forall f \in F, f' \in F, f \neq f', \rho_f \neq \rho_{f'}, (i,j) \in E_f, (j,i) \in E_{f'}. \end{aligned} \tag{5.24}$$

Constraints (5.23) and (5.24) ensure that any pair of trains using one cell in the same or different direction respectively are conflict-free, by avoiding the overlap between the cell release time for a preceding train and the cell occupancy time for a successive train. Specifically, for both train f and f' traversing cell (i,j) (i.e., with the same running direction $\rho_f = \rho_{f'}$), if train f' arrives at cell (i,j) after train f, i.e., $\theta_{f,f',i,j} = 1$, constraint (5.24) is non-active and (5.23) reduces to $\sigma_{f',i,j} \ge \delta_{f,i,j}$, which implies that the occupancy time of cell (i,j) for train f' should be later than the release time of cell (i,j) for train f.

To formulate the uniformly accelerating and decelerating motions, six logical speed indicators $\zeta_{1,f,i,j}$, ..., $\zeta_{6,f,i,j}$ are used to indicate the train speed. Table 5.2 gives an overview of the link between the speed conditions and the speed indicators, and Appendix A.1.1 provides the detailed explanation of these indicators. By adapting the transformation properties in Williams (2013) (see also Section 5.4.2), these if-then constraints can be further represented by a set of linear inequalities. For instance,

$\zeta_{1,f,i,j} = 1$, if and only if $v^{\text{in}}_{f,i,j} \leq v^{\text{cru}}_{f,i,j}$ can be represented by the following inequalities:

$$v^{\text{in}}_{f,i,j} - v^{\text{cru}}_{f,i,j} \leq v^{\text{nlim}}_{i} \cdot (1 - \zeta_{1,f,i,j}), \tag{5.25a}$$
$$v^{\text{in}}_{f,i,j} - v^{\text{cru}}_{f,i,j} \geq \varepsilon + (-v^{\text{clim}}_{i,j} - \varepsilon) \cdot \zeta_{1,f,i,j}, \tag{5.25b}$$

where v^{nlim}_{i} is the upper bound of $(v^{\text{in}}_{f,i,j} - v^{\text{cru}}_{f,i,j})$ and $-v^{\text{clim}}_{i,j}$ is the lower bound of $(v^{\text{in}}_{f,i,j} - v^{\text{cru}}_{f,i,j})$.

Thanks to the logical speed indicators $\zeta_{1,f,i,j},\ldots,\zeta_{6,f,i,j}$, we can formulate the uniformly accelerating and decelerating motion in a linear manner and consider multiple scenarios (where different values of acceleration and deceleration are required) at once. The following set of constraints is presented for the incoming phase, in which

$$-\frac{v^{\text{cru}}_{f,i,j} - v^{\text{in}}_{f,i,j}}{\beta_{f,i,j}} - M \cdot \zeta_{1,f,i,j} \leq a^{\text{cru}}_{f,i,j} - a_{f,i,j} \leq -\frac{v^{\text{cru}}_{f,i,j} - v^{\text{in}}_{f,i,j}}{\beta_{f,i,j}} + M \cdot \zeta_{1,f,i,j} \tag{5.26a}$$

indicates the uniformly decelerating motion at a steady deceleration $-\beta_{f,i,j}$,

$$\begin{aligned}\frac{v^{\text{cru}}_{f,i,j} - v^{\text{in}}_{f,i,j}}{\alpha_{2,f,i,j}} - M \cdot \left(2 - \zeta_{1,f,i,j} - \zeta_{3,f,i,j}\right) &\leq a^{\text{cru}}_{f,i,j} - a_{f,i,j} \\ &\leq \frac{v^{\text{cru}}_{f,i,j} - v^{\text{in}}_{f,i,j}}{\alpha_{2,f,i,j}} + M \cdot \left(2 - \zeta_{1,f,i,j} - \zeta_{3,f,i,j}\right)\end{aligned} \tag{5.26b}$$

indicates the uniformly accelerating motion at a steady acceleration $\alpha_{2,f,i,j}$, when the train speed is always larger than the switching speed v^{turn}_{f},

$$\begin{aligned}\frac{v^{\text{cru}}_{f,i,j} - v^{\text{in}}_{f,i,j}}{\alpha_{1,f,i,j}} - M \cdot \left(2 - \zeta_{1,f,i,j} - \zeta_{4,f,i,j}\right) &\leq a^{\text{cru}}_{f,i,j} - a_{f,i,j} \\ &\leq \frac{v^{\text{cru}}_{f,i,j} - v^{\text{in}}_{f,i,j}}{\alpha_{1,f,i,j}} + M \cdot \left(2 - \zeta_{1,f,i,j} - \zeta_{4,f,i,j}\right)\end{aligned} \tag{5.26c}$$

indicates the uniformly accelerating motion at a steady acceleration $\alpha_{1,f,i,j}$, when the train speed is always less than the switching speed v^{turn}_{f}, and

$$\begin{aligned}\frac{v^{\text{turn}}_{f} - v^{\text{in}}_{f,i,j}}{\alpha_{1,f,i,j}} - M \cdot \left(1 - \zeta_{1,f,i,j} + 2 \cdot \zeta_{3,f,i,j} + 2 \cdot \zeta_{4,f,i,j}\right) &\leq a^{\text{turn}}_{f,i,j} - a_{f,i,j} \\ &\leq \frac{v^{\text{turn}}_{f} - v^{\text{in}}_{f,i,j}}{\alpha_{1,f,i,j}} + M \cdot \left(1 - \zeta_{1,f,i,j} + 2 \cdot \zeta_{3,f,i,j} + 2 \cdot \zeta_{4,f,i,j}\right)\end{aligned} \tag{5.26d}$$

$$\begin{aligned}\frac{v^{\text{cru}}_{f,i,j} - v^{\text{turn}}_{f}}{\alpha_{2,f,i,j}} - M \cdot \left(1 - \zeta_{1,f,i,j} + 2 \cdot \zeta_{3,f,i,j} + 2 \cdot \zeta_{4,f,i,j}\right) &\leq a^{\text{cru}}_{f,i,j} - a^{\text{turn}}_{f,i,j} \\ &\leq \frac{v^{\text{cru}}_{f,i,j} - v^{\text{turn}}_{f}}{\alpha_{2,f,i,j}} + M \cdot \left(1 - \zeta_{1,f,i,j} + 2 \cdot \zeta_{3,f,i,j} + 2 \cdot \zeta_{4,f,i,j}\right)\end{aligned} \tag{5.26e}$$

indicate a two-stage uniformly accelerating motion, i.e., the train first accelerates at a steady acceleration $\alpha_{1,f,i,j}$ and then accelerates at a steady acceleration $\alpha_{2,f,i,j}$. The detailed explanation of (5.26) is provided in Appendix A.1.2.

To compute the time points $a^{\text{turn}}_{f,i,j}$ and $d^{\text{turn}}_{f,i,j}$ under some special scenarios, e.g., a train does not reach the switching speed v^{turn}_{f} on a cell, the following set of constraints is proposed for the incoming phase:

$$a^{\text{turn}}_{f,i,j} \leq a_{f,i,j} + M \cdot \left|\zeta_{1,f,i,j} - \zeta_{3,f,i,j}\right|, \tag{5.27a}$$
$$a^{\text{turn}}_{f,i,j} \geq a^{\text{cru}}_{f,i,j} - M \cdot \left|\zeta_{1,f,i,j} - \zeta_{4,f,i,j}\right|. \tag{5.27b}$$

Specifically, when $\zeta_{1,f,i,j} = \zeta_{3,f,i,j}$, i.e., $v^{\text{turn}}_{f} \leq v^{\text{in}}_{f,i,j} \leq v^{\text{cru}}_{f,i,j}$ or $v^{\text{cru}}_{f,i,j} < v^{\text{in}}_{f,i,j} < v^{\text{turn}}_{f}$, constraint (5.27a) reduces to $a^{\text{turn}}_{f,i,j} \leq a_{f,i,j}$. Since $a_{f,i,j} \leq a^{\text{turn}}_{f,i,j}$ is required in (5.8), we can further obtain $a^{\text{turn}}_{f,i,j} = a_{f,i,j}$, i.e., let the time point that train f reaches the speed v^{turn}_{f} on cell (i, j) equals the arrival time of the train. Formulations similar to (5.26) and (5.27) can also be constructed for the outgoing phase.

The optimization problem including the objective function (5.1) and constraints (5.2)-(5.27), is called the P_{NLP} problem, among which there are if-then constraints, i.e., (5.12) and (5.15), and nonlinear constraints, i.e., (5.14) and (5.16).

5.4.2 Formulation of the $\mathrm{P}_{\mathrm{PWA}}$ problem: the $\mathrm{P}_{\mathrm{NLP}}$ problem approximated by using PWA functions

This section proposes the MILP problem ($\mathrm{P}_{\mathrm{PWA}}$) by reformulating and approximating the nonlinear terms in the $\mathrm{P}_{\mathrm{NLP}}$ problem, i.e., (5.12), (5.14), (5.15), and (5.16). A PWA function is adopted for the approximation, as well as three transformation properties proposed in Williams (2013), which are briefly introduced below. Interested readers may refer to this reference for more details.

Let us consider the statement $\tilde{f}(\tilde{x}) \leq 0$, where $\tilde{f}: \mathbb{R}^n \to \mathbb{R}$ is affine, $\tilde{x} \in \chi$ with $\chi \subset \mathbb{R}^n$ and let $\tilde{Q} = \max\limits_{\tilde{x}\in\chi} \tilde{f}(\tilde{x})$, $\tilde{q} = \min\limits_{\tilde{x}\in\chi} \tilde{f}(\tilde{x})$.

- ***Transformation property I***: If we introduce a logical variable $l \in \{0,1\}$, then the following equivalence holds: $\left[\tilde{f}(\tilde{x}) \leq 0\right] \Leftrightarrow [l = 1]$ is true iff $\tilde{f}(\tilde{x}) \leq \tilde{Q} \cdot (1-l)$ and $\tilde{f}(\tilde{x}) \geq \varepsilon + (\tilde{q} - \varepsilon) \cdot l$.

- ***Transformation property II***: The product of two logical variables l_1 and l_2 can be replaced by an auxiliary logical variable $l_3 = l_1 \cdot l_2$, i.e., $[l_3 = 1] \Leftrightarrow [l_1 = l_2 = 1]$, which is equivalent to three linear inequalities: $-l_1 + l_3 \leq 0$, $-l_2 + l_3 \leq 0$ and $l_1 + l_2 - l_3 \leq 1$.

- ***Transformation property III***: The product $l \cdot \tilde{f}(\tilde{x})$ can be replaced by the auxiliary real variable $r = l \cdot \tilde{f}(\tilde{x})$, which satisfies $[l = 0] \Rightarrow [r = 0]$ and $[l = 1] \Rightarrow \left[r = \tilde{f}(\tilde{x})\right]$. Then $r = l \cdot \tilde{f}(\tilde{x})$ is equivalent to four inequalities: $r \leq \tilde{Q} \cdot l$, $r \geq \tilde{q} \cdot l$, $r \leq \tilde{f}(\tilde{x}) - \tilde{q} \cdot (1-l)$ and $r \geq \tilde{f}(\tilde{x}) - \tilde{Q} \cdot (1-l)$.

Note that *Transformation property I* has been used to formulate (5.25) for the speed indicators in Table 5.2 of Section 5.4.1. Moreover, the if-then constraints (5.12) and (5.15) can be reformulated as linear constraints by using *Transformation property I* (for the sake of compactness, we do not present the details here).

To approximate the nonlinear terms, constraint (5.14a) for calculating $L^{\mathrm{in}}_{f,i,j}$ is first reformulated as the following set of linear constraints by using the logical speed indicators $\zeta_{1,f,i,j}$, $\zeta_{3,f,i,j}$, and $\zeta_{4,f,i,j}$:

$$-\frac{(v^{\mathrm{cru}}_{f,i,j})^2-(v^{\mathrm{in}}_{f,i,j})^2}{2\cdot\beta_{f,i,j}} - M\cdot\zeta_{1,f,i,j} \leq L^{\mathrm{in}}_{f,i,j} \leq -\frac{(v^{\mathrm{cru}}_{f,i,j})^2-(v^{\mathrm{in}}_{f,i,j})^2}{2\cdot\beta_{f,i,j}} + M\cdot\zeta_{1,f,i,j}, \tag{5.28a}$$

$$\begin{aligned}\frac{(v^{\mathrm{cru}}_{f,i,j})^2-(v^{\mathrm{in}}_{f,i,j})^2}{2\cdot\alpha_{2,f,i,j}} - M\cdot\left(2-\zeta_{1,f,i,j}-\zeta_{3,f,i,j}\right) &\leq L^{\mathrm{in}}_{f,i,j} \\ &\leq \frac{(v^{\mathrm{cru}}_{f,i,j})^2-(v^{\mathrm{in}}_{f,i,j})^2}{2\cdot\alpha_{2,f,i,j}} + M\cdot\left(2-\zeta_{1,f,i,j}-\zeta_{3,f,i,j}\right),\end{aligned} \tag{5.28b}$$

$$\begin{aligned}\frac{(v^{\mathrm{cru}}_{f,i,j})^2-(v^{\mathrm{in}}_{f,i,j})^2}{2\cdot\alpha_{1,f,i,j}} - M\cdot\left(2-\zeta_{1,f,i,j}-\zeta_{4,f,i,j}\right) &\leq L^{\mathrm{in}}_{f,i,j} \\ &\leq \frac{(v^{\mathrm{cru}}_{f,i,j})^2-(v^{\mathrm{in}}_{f,i,j})^2}{2\cdot\alpha_{1,f,i,j}} + M\cdot\left(2-\zeta_{1,f,i,j}-\zeta_{4,f,i,j}\right),\end{aligned} \tag{5.28c}$$

$$\begin{aligned}\frac{(v_f^{\text{turn}})^2-(v_{f,i,j}^{\text{in}})^2}{2\cdot\alpha_{1,f,i,j}}+\frac{(v_{f,i,j}^{\text{cru}})^2-(v_f^{\text{turn}})^2}{2\cdot\alpha_{2,f,i,j}}-M\cdot\big(1-\zeta_{1,f,i,j}+2\cdot\zeta_{3,f,i,j}\\ +2\cdot\zeta_{4,f,i,j}\big)\leq L_{f,i,j}^{\text{in}}\leq\frac{(v_f^{\text{turn}})^2-(v_{f,i,j}^{\text{in}})^2}{2\cdot\alpha_{1,f}}+\frac{(v_{f,i,j}^{\text{cru}})^2-(v_f^{\text{turn}})^2}{2\cdot\alpha_{2,f,i,j}}\\ +M\cdot\big(1-\zeta_{1,f,i,j}+2\cdot\zeta_{3,f,i,j}+2\cdot\zeta_{4,f,i,j}\big)\,.\end{aligned}\tag{5.28d}$$

These constraints satisfy the uniformly accelerating and decelerating motions, and the detailed explanation of (5.28) is provided in Appendix A.1.3. Constraints similar to (5.28) can also be constructed for reformulating (5.14c) and for further calculating $L_{f,i,j}^{\text{out}}$, but for the sake of compactness, we do not report those details here. Let $\varpi_{f,i,j}^{\text{in}}$, $\varpi_{f,i,j}^{\text{cru}}$, and $\varpi_{f,i,j}^{\text{out}}$ be the square of $v_{f,i,j}^{\text{in}}$, $v_{f,i,j}^{\text{cru}}$, and $v_{f,i,j}^{\text{out}}$ respectively, as formulated in (5.29):

$$\varpi_{f,i,j}^{\text{in}}=\left(v_{f,i,j}^{\text{in}}\right)^2,\quad\forall f\in F,(i,j)\in E_f,\tag{5.29a}$$

$$\varpi_{f,i,j}^{\text{cru}}=\left(v_{f,i,j}^{\text{cru}}\right)^2,\quad\forall f\in F,(i,j)\in E_f,\tag{5.29b}$$

$$\varpi_{f,i,j}^{\text{out}}=\left(v_{f,i,j}^{\text{out}}\right)^2,\quad\forall f\in F,(i,j)\in E_f.\tag{5.29c}$$

As a result, (5.28a)-(5.28d) become linear, and instead (5.29a)-(5.29c) are nonlinear and should be approximated. The reason that we first reformulate (5.14a) and (5.14c) as above is to reduce the number of nonlinear terms that need to be approximated, i.e., by introducing (5.29), (5.28), and the constraints obtained when reformulating (5.14c) become linear. Regarding (5.14b), which calculates $L_{f,i,j}^{\text{cru}}$ for the cruising phase, an additional step is needed to reformulate the nonlinear term $x\cdot y$ as $\frac{(x+y)^2-(x-y)^2}{4}$, i.e., reformulating (5.14b) as follows:

$$L_{i,j}^{\text{cru}}=\frac{1}{4}\cdot\left[\left(v_{f,i,j}^{\text{cru}}+d_{f,i,j}^{\text{cru}}-a_{f,i,j}^{\text{cru}}\right)^2-\left(v_{f,i,j}^{\text{cru}}-d_{f,i,j}^{\text{cru}}+a_{f,i,j}^{\text{cru}}\right)^2\right].\tag{5.30}$$

Then, by defining

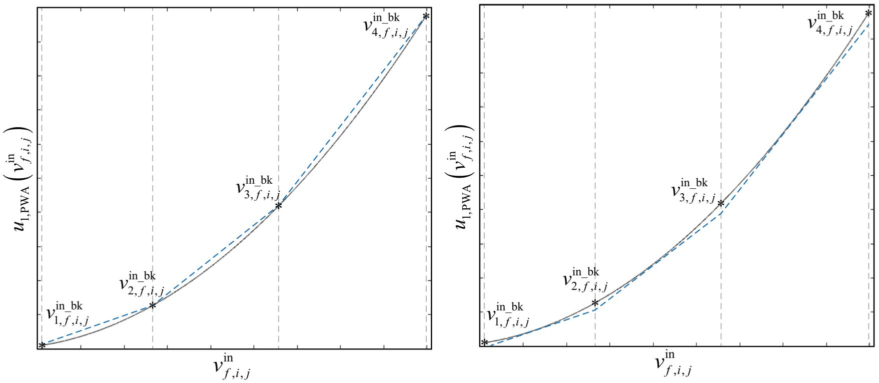
$$m_{f,i,j}=\left(v_{f,i,j}^{\text{cru}}+d_{f,i,j}^{\text{cru}}-a_{f,i,j}^{\text{cru}}\right)^2,\tag{5.31a}$$

$$n_{f,i,j}=\left(v_{f,i,j}^{\text{cru}}-d_{f,i,j}^{\text{cru}}+a_{f,i,j}^{\text{cru}}\right)^2,\tag{5.31b}$$

equation (5.30) becomes linear, and instead (5.31a)-(5.31b) need to be approximated by using PWA functions, as will be explained next.

Based on the above reformulation, the nonlinear constraints (5.16), (5.29), and (5.31) need to be further approximated by using PWA functions. For simplicity, we only describe the approximating process of (5.29a) here; a similar process can be followed for approximating the other nonlinear constraints.

We adopt an approximation using three affine sub-functions as illustrated in Figure 5.3. Note that more affine sub-functions can be selected if needed; the approach then stays similar in such a case. We consider two kinds of line fitting methods, namely the upper/lower line fitting method, where the values of the approximated line segments are no less/greater than the original curve, as shown in Figure 5.3(a)-Figure 5.3(b) respectively. The relevant coefficients regarding the three line segments (e.g., $v_{2,f,i,j}^{\text{in\_bk}}$ and



3(a) Upper line fitting method

3(b) Lower line fitting method

Figure 5.3: The PWA approximation of the non-linear function (5.29a)

$v^{\text{in\_bk}}_{3,f,i,j}$) are determined through minimizing the approximation errors between the original curve (indicated in black) and three line segments (indicated in blue). It is worth noting that the reason of using these two methods is to keep the approximated constraints feasible. For instance, constraint (5.29a) should be approximated by using the lower line fitting method in Figure 5.3(b), in order to guarantee that the approximated value of the train speed is not greater than its actual value and the corresponding speed limitation as well. Additionally, the approximated value of the time, the distance, and the square of the train speed should not be negative, so we keep all approximated values non-negative.

The PWA approximation of the nonlinear function (5.29a) over the interval $\left[\min\left(v^{\text{in}}_{f,i,j}\right),\right.$ $\left.\max\left(v^{\text{in}}_{f,i,j}\right)\right]$, i.e., $\left[v^{\text{in\_bk}}_{1,f,i,j}, v^{\text{in\_bk}}_{4,f,i,j}\right]$, can be written as

$$u_{1,\text{PWA}}\left(v^{\text{in}}_{f,i,j}\right) = \varpi^{\text{in}}_{f,i,j} = \begin{cases} \mu_{1,f,i,j}\cdot v^{\text{in}}_{f,i,j} + \eta_{1,f,i,j}, \text{ if } v^{\text{in\_bk}}_{1,f,i,j} \le v^{\text{in}}_{f,i,j} \le v^{\text{in\_bk}}_{2,f,i,j} \\ \mu_{2,f,i,j}\cdot v^{\text{in}}_{f,i,j} + \eta_{2,f,i,j}, \text{ if } v^{\text{in\_bk}}_{2,f,i,j} \le v^{\text{in}}_{f,i,j} \le v^{\text{in\_bk}}_{3,f,i,j} \\ \mu_{3,f,i,j}\cdot v^{\text{in}}_{f,i,j} + \eta_{3,f,i,j}, \text{ if } v^{\text{in\_bk}}_{3,f,i,j} \le v^{\text{in}}_{f,i,j} \le v^{\text{in\_bk}}_{4,f,i,j} \end{cases} \tag{5.32}$$

where $\mu_{x,f,i,j}$ and $\eta_{x,f,i,j}$, $x = 1,\ldots,3$, are coefficients, .

Let us consider the logical variables $\lambda_{1,f,i,j}$ and $\lambda_{2,f,i,j}$ to satisfy the conditions $\left[v^{\text{in}}_{f,i,j} - v^{\text{in\_bk}}_{2,f,i,j} \le 0\right] \Leftrightarrow \left[\lambda_{1,f,i,j} = 1\right]$ and $\left[v^{\text{in}}_{f,i,j} - v^{\text{in\_bk}}_{3,f,i,j} \le 0\right] \Leftrightarrow \left[\lambda_{2,f,i,j} = 1\right]$, which can be represented as a set of linear inequalities by using Transformation property I (Williams, 2013). Then, the function (5.32) can be rewritten as

$$\begin{aligned} u_{1,\text{PWA}}\left(v^{\text{in}}_{f,i,j}\right) = \varpi^{\text{in}}_{f,i,j} = {} & \lambda_{1,f,i,j}\cdot\lambda_{2,f,i,j}\cdot\left(\mu_{1,f,i,j}\cdot v^{\text{in}}_{f,i,j} + \eta_{1,f,i,j}\right) \\ & + \left(1-\lambda_{1,f,i,j}\right)\cdot\lambda_{2,f,i,j}\cdot\left(\mu_{2,f,i,j}\cdot v^{\text{in}}_{f,i,j} + \eta_{2,f,i,j}\right) \\ & + \left(1-\lambda_{1,f,i,j}\right)\cdot\left(1-\lambda_{2,f,i,j}\right)\cdot\left(\mu_{3,f,i,j}\cdot v^{\text{in}}_{f,i,j} + \eta_{3,f,i,j}\right) \end{aligned} \tag{5.33}$$

We introduce the auxiliary logical variable $\lambda_{3,f,i,j}$ to replace the product $\lambda_{1,f,i,j}\cdot\lambda_{2,f,i,j}$. According to Transformation property II, the condition $\lambda_{3,f,i,j} = \lambda_{1,f,i,j}\cdot\lambda_{2,f,i,j}$ can also be rewritten as a system of linear inequalities. Moreover, by defining new auxiliary variables $z_{x,f,i,j} = \lambda_{x,f,i,j} \cdot v^{\text{in}}_{f,i,j}$, $x = 1,...,3$, which can be expressed as a set of linear inequalities by adapting Transformation property III, the function (5.33) can be further rewritten as

$$\begin{aligned} u_{1,\text{PWA}}\left(v^{\text{in}}_{f,i,j}\right) = \varpi^{\text{in}}_{f,i,j} &= z_{3,f,i,j} \cdot \left(\mu_{1,f,i,j} - \mu_{2,f,i,j} + \mu_{3,f,i,j}\right) \\ &+ z_{2,f,i,j} \cdot \left(\mu_{2,f,i,j} - \mu_{3,f,i,j}\right) + \lambda_{3,f,i,j} \cdot \left(\eta_{1,f,i,j} - \eta_{2,f,i,j} + \eta_{3,f,i,j}\right) \\ &+ \lambda_{2,f,i,j} \cdot \left(\eta_{2,f,i,j} - \eta_{3,f,i,j}\right) - z_{1,f,i,j} \cdot \mu_{3,f,i,j} - \lambda_{1,f,i,j} \cdot \eta_{3,f,i,j} \\ &+ \mu_{3,f,i,j} \cdot v^{\text{in}}_{f,i,j} + \eta_{3,f,i,j} \end{aligned} \tag{5.34}$$

Finally, the nonlinear constraints (5.29a) can be replaced by the linear equation (5.34) and the linear inequalities obtained by using the three transformation properties, three logical variables $\lambda_{1,f,i,j}$, $\lambda_{2,f,i,j}$, $\lambda_{3,f,i,j}$, and three auxiliary variables $z_{1,f,i,j}$, $z_{2,f,i,j}$, $z_{3,f,i,j}$. A similar process can be followed for approximating the nonlinear constraints (5.16), (5.29b), (5.29c), and (5.31) by applying the three transformation properties and introducing extra logical variables and auxiliary variables, thus we do not report those details in this chapter.

In particular, the clearing time constraint (5.16) is approximated by using a piece-wise constant function. We can also use the transformation properties in Williams (2013) to approximate (5.16), similar to the approximating process of (5.29a).

The optimization problem including the objective function (5.1), constraints (5.2)-(5.11), (5.13), (5.18)-(5.27), (5.28), (5.30), (5.34), and those constraints for reformulating (5.12) and (5.15) and for approximating (5.16), (5.29b), (5.29c), and (5.31), which are not detailed in this chapter, is called the P_{PWA} problem.

5.4.3 Formulation of the P_{TSPO} problem: considering multiple train speed profile options generated in a preprocessing step

Aiming at reducing the solving complexity and the approximation errors, in this section, another MILP problem (P_{TSPO}) considering multiple TSPOs is developed. A preprocessing step is used to generate multiple TSPOs by considering discrete speed values, in order to restrict the search only to an efficient subset of all possible TSPOs. Figure 5.4 gives an example to illustrate how TSPOs are generated for a train on cells along its route. Given a set of discrete speed values $[v_1, v_2, v_3, v_4, v_5]$, we can create a complex space-speed network, indicated by the solid gray lines. The created space-speed network respects the formulas of the uniformly accelerating/decelerating motion and the technical requirements of train operations and infrastructures, e.g., train speed limitation, train dwell requirements, train acceleration/deceleration (which depends on traction/braking force), and length of block section. For instance, due to the short length of cell1, the train cannot reach speeds v_4 and v_5 within cell1 based on its acceleration rate; therefore, the options to speeds v_4 and v_5 are not included in the space-speed network. Such an assessment on the feasibility of TSPO is also considered for train deceleration, e.g., when the train approaches its destination or an intermediate station

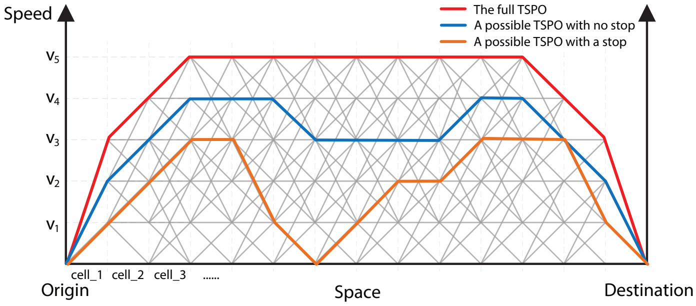

Speed
The full TSPO
A possible TSPO with no stop
A possible TSPO with a stop
v_5
v_4
v_3
v_2
v_1
cell_1 cell_2 cell_3 ......
Origin
Space
Destination

Figure 5.4: Train speed profile generation in the preprocessing step

where a stop is required, the options that cannot let the train stop at the corresponding station are discarded. Based on the space-speed network, we can then select TSPOs for the train from its origin to its destination. Three possible TSPOs are indicated in Figure 5.4: the red line indicates the full TSPO (which lets the train run as fast as possible), the blue line indicates a TSPO with no stop, and the orange line indicates a TSPO with a stop at an intermediate station. Note that the full TSPO may also include a train stop at an intermediate station if the stop is required. Moreover, we can also discard some obviously-inefficient TSPOs, e.g., the TSPOs that allow train deceleration just following a train acceleration within one cell. In such a way, only an efficient subset of all possible TSPOs are generated. We still refer to the notations in Table 5.1, with the changes listed in Table 5.3.

For each train on each cell, some train speed profile vectors $y_{f,i,j,b}$ are given, and each vector contains a possible set of incoming, cruising, and outgoing speeds, i.e., $y_{f,i,j,b} = \begin{bmatrix} y^{\text{in}}_{f,i,j,b} & y^{\text{cru}}_{f,i,j,b} & y^{\text{out}}_{f,i,j,b} \end{bmatrix}^\top$. Logical parameters $\zeta_{1,f,i,j,b}, \dots, \zeta_{6,f,i,j,b}$ are used to indicate the speed conditions in the corresponding train speed profile vector $y_{f,i,j,b}$, as explained in Table 5.2. The problem objective is also to minimize the total train delay times at all visited stations, as formulated in (5.1). In addition, the following constraints are used by the P_{TSPO} problem:

$$v^{\text{in}}_{f,i,j} = \sum_{b=1}^{|Y_{f,i,j}|} \vartheta_{f,i,j,b} \cdot y^{\text{in}}_{f,i,j,b}, \quad \forall f \in F, (i,j) \in E_f, \tag{5.35}$$

$$v^{\text{cru}}_{f,i,j} = \sum_{b=1}^{|Y_{f,i,j}|} \vartheta_{f,i,j,b} \cdot y^{\text{cru}}_{f,i,j,b}, \quad \forall f \in F, (i,j) \in E_f, \tag{5.36}$$

$$v^{\text{out}}_{f,i,j} = \sum_{b=1}^{|Y_{f,i,j}|} \vartheta_{f,i,j,b} \cdot y^{\text{out}}_{f,i,j,b}, \quad \forall f \in F, (i,j) \in E_f, \tag{5.37}$$

$$\sum_{b=1}^{|Y_{f,i,j}|} \vartheta_{f,i,j,b} = 1, \quad \forall f \in F, (i,j) \in E_f \tag{5.38}$$

Table 5.3: Changes of sets, subscripts, parameters, and variables for the $\mathrm{P_{TSPO}}$ problem, compared to Table 5.1

Type of changes	Symbol	Description
added set	$Y_{f,i,j}$	set of options of train speed profile vectors that train f may follow on cell (i,j), $\left\lvert Y_{f,i,j} \right\rvert$ is the number of TSPOs for train f on cell (i,j)
added subscript	b	TSPO index, $b_{f,i,j} = 1, \ldots, \left\lvert Y_{f,i,j} \right\rvert$, which indicates the TSPO index of train f on cell (i,j)
added variable	$\vartheta_{f,i,j,b}$	binary variables, $\vartheta_{f,i,j,b} = 1$ if the corresponding train speed vector $y_{f,i,j,b}$ is used by train f on cell (i,j), and otherwise $\vartheta_{f,i,j,b} = 0$
added parameter	$y^{\mathrm{in}}_{f,i,j,b}$	b^{th} incoming speed of train f on cell (i,j)
added parameter	$y^{\mathrm{cru}}_{f,i,j,b}$	b^{th} cruising speed of train f on cell (i,j)
added parameter	$y^{\mathrm{out}}_{f,i,j,b}$	b^{th} outgoing speed of train f on cell (i,j)
added parameter	$y_{f,i,j,b}$	b^{th} train speed profile vector, $y_{f,i,j,b} \in Y_{f,i,j}$, $y_{f,i,j,b} = \begin{bmatrix} y^{\mathrm{in}}_{f,i,j,b} & y^{\mathrm{cru}}_{f,i,j,b} & y^{\mathrm{out}}_{f,i,j,b} \end{bmatrix}^{\top} \in Y_{f,i,j}$
modified parameter	$L^{\mathrm{in}}_{f,i,j,b}$	distance that train f runs over on cell (i,j) in the incoming phase in the b^{th} train speed profile vector $y_{f,i,j,b}$
modified parameter	$L^{\mathrm{out}}_{f,i,j,b}$	distance that train f runs over on cell (i,j) in the outgoing phase in the b^{th} train speed profile vector $y_{f,i,j,b}$
modified parameter	$\zeta_{1,f,i,j,b}, \ldots, \zeta_{6.f,i,j,b}$	logical parameters to indicate the relation of the incoming, cruising, outgoing speed, and switching speed v^{turn}_{f} in the b^{th} train speed profile vector $y_{f,i,j,b}$, see Table 5.2

$$\begin{aligned}\vartheta_{f,i,j,b}\cdot\frac{\left(L^{\text{cell}}_{i,j}-L^{\text{in}}_{f,i,j,b}-L^{\text{out}}_{f,i,j,b}\right)}{y^{\text{cru}}_{f,i,j,b}} \leq d^{\text{cru}}_{f,i,j}-a^{\text{cru}}_{f,i,j} \leq \\ \frac{\left(L^{\text{cell}}_{i,j}-L^{\text{in}}_{f,i,j,b}-L^{\text{out}}_{f,i,j,b}\right)}{y^{\text{cru}}_{f,i,j,b}}+M\cdot\left(1-\vartheta_{f,i,j,b}\right)\end{aligned} \tag{5.39}$$

$$\begin{aligned}\frac{(d_{f,i,j}-a_{f,i,j})\cdot y^{\text{out}}_{f,i,j,b}}{\varepsilon+y^{\text{out}}_{f,i,j,b}}-M\cdot(1-\vartheta_{f,i,j,b}) \leq \tau^{\text{approach}}_{f,j,k} \leq \\ \frac{(d_{f,i,j}-a_{f,i,j})\cdot y^{\text{out}}_{f,i,j,b}}{\varepsilon+y^{\text{out}}_{f,i,j,b}}+M\cdot(1-\vartheta_{f,i,j,b}), \\ \forall f\in F,(i,j)\in E_f,(j,k)\in E_f,b=1,\dots,\left|Y_{f,i,j}\right|\end{aligned} \tag{5.40}$$

$$\tau^{\text{clear}}_{f,p,i}=\sum_{b=1}^{\left|Y_{f,i,j}\right|}\frac{2\cdot L^{\text{train}}_{f}\cdot\vartheta_{f,i,j,b}}{y^{\text{in}}_{f,i,j,b}+y^{\text{cru}}_{f,i,j,b}},\quad \forall f\in F,(p,i)\in E_f,(i,j)\in E_f \tag{5.41}$$

Constraints (5.35)-(5.37) determine the selected incoming, cruising, and outgoing speed respectively, i.e., if $\vartheta_{f,i,j,b}=1$, then $v^{\text{in}}_{f,i,j}=y^{\text{in}}_{f,i,j,b}$, $v^{\text{cru}}_{f,i,j}=y^{\text{cru}}_{f,i,j,b}$, and $v^{\text{out}}_{f,i,j}=y^{\text{out}}_{f,i,j,b}$. Constraint (5.38) ensures that one and only one TSPO is selected for each train on each cell. Constraint (5.39) is the cell length constraint, which restricts the distance that a train runs over on a cell. Specifically, if $\vartheta_{f,i,j,b}=1$, i.e., the b^{th} train speed profile vector $y_{f,i,j,b}$ is used, constraint (5.39) reduces to a linear equation $d^{\text{cru}}_{f,i,j}-a^{\text{cru}}_{f,i,j}=\frac{\left(L^{\text{cell}}_{i,j}-L^{\text{in}}_{f,i,j,b}-L^{\text{out}}_{f,i,j,b}\right)}{y^{\text{cru}}_{f,i,j,b}}$, which satisfies the basic formula "time $=\frac{\text{distance}}{\text{constant speed}}$" of the constant-speed motion. Constraints (5.40) and (5.41) define the approach time and clearing time respectively. Note that if train f stops on cell (i,j), i.e., $\vartheta_{f,i,j,b}=1$ and $y^{\text{out}}_{f,i,j,b}=0$, the approach time of train f on the successive cell (j,k) should be zero. To avoid the denominator from becoming zero, a sufficiently small positive number ε is used in (5.40).

The optimization problem including the objective function (5.1), constraints (5.2)-(5.4), (5.8)-(5.11), (5.18)-(5.24), (5.26)-(5.27), and (5.35)-(5.41), is called the P_{TSPO} problem.

5.5 Solution approaches

In this section, we introduce the solution approaches for solving the proposed optimization approaches, i.e., a two-level approach for solving the P_{NLP} problem and a custom-designed two-step approach for solving the P_{TSPO} problem. Regarding the solution approach of the P_{PWA} problem, an MILP solver can be used, such as CPLEX or Gurobi.

5.5.1 A two-level approach for solving the P_{NLP} problem

The nonlinear dynamics of the P_{NLP} problem limit its scalability and applicability for large-scale instances. Therefore, we propose a two-level approach to solve the P_{NLP}

problem, as illustrated in Figure 5.5(a), where a genetic-algorithm-based heuristic is introduced to generate the possible train orders based on the track layouts, train routes, delays, etc. in the upper level, and a nonlinear programming method is used in the lower level to optimize the departure/arrival times and the train speed profiles under the fixed train orders.

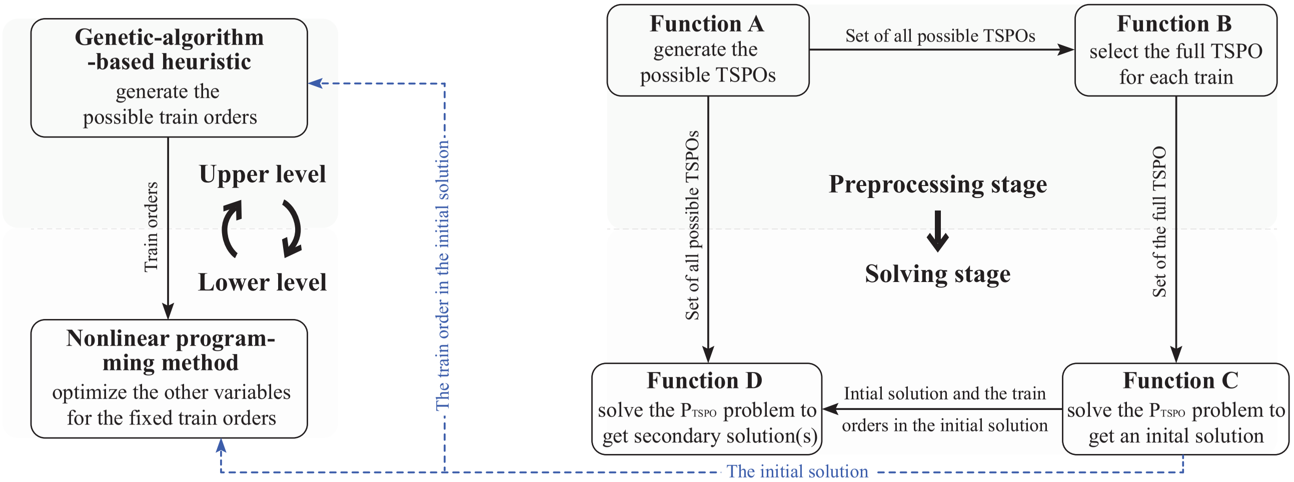
In the upper level, to describe the entire set of train orders in the network, we use a chromosome. This is defined as a vector that is composed by several sub-vectors. There is a sub-vector for each merging/diverging point (i.e., where train orders can change; we call them relevant points in what follows) of the network. A sub-vector is used to indicate the train orders at that specific relevant point. In order to generate feasible initial populations, the train orders defined in the original train timetable or the initial solution can be used as a starting point. In addition, we only adopt the mutation operation for the genetic algorithm used in this research to generate feasible chromosomes. In particular, the mutation operation is carried out by swapping the order of two trains at a relevant point inside the chromosomes. Since the orders of trains at the relevant points are related to each other, the order of these two chosen trains at other relevant points may need to be swapped accordingly. Furthermore, the train delays at the relevant points are also used as a supplement for the decision of swapping trains. After a new population is generated, the nonlinear programming method in the lower level is used to optimize the departure/arrival times and train speed profiles and to obtain the fitness for each chromosome. We terminate the genetic algorithm in the upper level of the two-level approach after a given number of generations.

Due to the non-convexity of the $\mathrm{P_{NLP}}$ problem, the two-level approach can only obtain a local minimum for the departure/arrival times and speeds, by given the train orders; therefore, the final solution of the nonlinear optimization problem is a local minimum associated with the best upper level solution. The two-level approach with multiple initial solutions (including multiple initial train orders for the upper level and multiple initial departure/arrival times and train speeds for the lower level) could improve the performance, but reaching the global optimum can in general not be guaranteed. The initial solution could be the original timetable or the initial solution obtained by the $\mathrm{P_{TSPO}}$ problem through considering a fixed full TSPO for each train, as indicated by the blue dashed line in Figure 5.5.

5.5.2 A custom-designed two-step approach for solving the $\mathrm{P_{TSPO}}$ problem

The $\mathrm{P_{TSPO}}$ problem is an MILP problem that can be solved by a standard MILP solver. Inspired by the good performance on similar problems in Xu et al. (2017), a custom designed two-step approach is particularly developed to solve the $\mathrm{P_{TSPO}}$ problem, in order to speed up the solving procedure, as illustrated in Figure 5.5(b).

As the $\mathrm{P_{TSPO}}$ problem is defined by considering multiple pre-determined TSPOs, a


Genetic-algorithm -based heuristic
generate the possible train orders
Train orders
Upper level
Lower level
Nonlinear programming method
optimize the other variables for the fixed train orders
The train order in the initial solution
Function A
generate the possible TSPOs
Set of all possible TSPOs
Function B
select the full TSPO for each train
Preprocessing stage
Solving stage
Set of all possible TSPOs
Set of the full TSPO
Function D
solve the PTSPO problem to get secondary solution(s)
Intial solution and the train orders in the initial solution
Function C
solve the PTSPO problem to get an inital solution
The initial solution

(a) Two-level approach for solving the P_{NLP} model

(b) Custom-designed two-step approach for solving the P_{TSPO} problem

Figure 5.5: Illustration of the solution approaches

preprocessing stage is used to generate the possible TSPOs (by Function A) and to clarify the full TSPO (by Function B). Each TSPO generated by Function A respects the formulas of the uniformly accelerating/decelerating motion and the technical requirements of train operations and infrastructures, e.g., train speed limitation, train acceleration/deceleration (which depends on traction/braking force), and length of block section. The full TSPO for each train derives from the corresponding set of all possible TSPOs, by selecting the fastest TSPO from this set that lets the train run as fast as possible. With a given set of possible TSPOs, the full TSPO is unique, i.e., the fastest one while respecting all operational requirements. In Function C of the solving stage, we consider the selected full TSPO only to solve the $\mathrm{P_{TSPO}}$ problem by using a standard MILP solver, which results in an initial solution (i.e., an upper bound with a fixed full TSPO for each train). Then, the obtained solution is given as a feasible initial solution to the MILP solver, for solving the $\mathrm{P_{TSPO}}$ problem with the larger set of all possible TSPOs. Therefore, in Function D, an improved secondary solution can be obtained through optimizing the TSPOs (and optimizing the train orders as well). Moreover, the train orders of the initial solution can also be given as an input of the problem in Function D; as a result, we can obtain an improved secondary solution with fixed train orders. Due to the limited number of TSPOs resulting from the preprocessing stage, only a local optimal solution can be obtained for the $\mathrm{P_{TSPO}}$ problem and its performance strongly depends on the given subset of TSPOs.

5.6 Case study

Before reporting the experimental results, we first describe the case study in Section 5.6.1, i.e., a Dutch railway network. In Section 5.6.2(1), we compare the overall performance of the three proposed optimization approaches based on the Dutch test case described in Section 5.6.1. For the $\mathrm{P_{PWA}}$ problem and the $\mathrm{P_{TSPO}}$ problem, we have multiple computational configurations; therefore, we further investigate the impact of these configurations on the results. In Section 5.6.2(2), the analysis of the $\mathrm{P_{PWA}}$ problem focuses on assessing the effectiveness of the approximation when using different line fitting methods, from the viewpoints of feasibility and approximation error. For the $\mathrm{P_{TSPO}}$ problem, Section 5.6.2(3) investigates the impact of the TSPOs generated in the preprocessing step on the solution quality, by considering different sets of discrete speed values. Moreover, we explore the benefits of changing train orders and managing train speeds. Finally, a lower bound is generated to evaluate the quality of the $\mathrm{P_{TSPO}}$ solution obtained within a given computation time limit. Moreover, we additionally report the detailed data about the solutions of this test case in the online repository (Research Collection ETH Zurich). In Appendix A.3, we explore the applicability of the proposed approach to a different test case adapted from INFORMS RAS (2012), in order to show the generality of the conclusions.

We use the SNOPT solver implemented in the MATLAB (R2016a) TOMLAB toolbox to solve the MINLP problem, i.e., the $\mathrm{P_{NLP}}$ problem, by applying the two-level

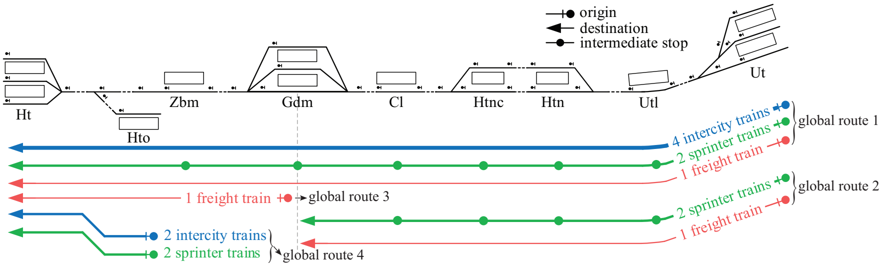


Figure 5.6: A real-world experimental network adapted from the Dutch railway network

approach introduced in Section 5.5. We adopt the IBM ILOG CPLEX optimization studio 12.6.3 with default settings to solve the MILP problems, i.e., the P_{PWA} problem and the P_{TSPO} problem. The custom-designed two-step approach described in Section 5.5.2 is particularly considered for the P_{TSPO} problem. Functions A and B of the custom-designed two-step approach are implemented in Visual C++ 2013. The experiments are all performed on a computer with an Intel® Core$^{\text{TM}}$ i7 @ 2.00 GHz processor and 16GB RAM.

5.6.1 Setup

We consider the line of the Dutch railway network connecting Utrecht (Ut) to Den Bosch (Ht), of about 50 km length. The network under consideration is shown in Figure 5.6. The network is composed of 40 nodes and 42 cells, with 2 main tracks, divided into a long corridor for each traffic direction and 9 stations. The two tracks in different directions are independent, so only one direction is considered, i.e., from Utrecht (Ut) to Den Bosch (Ht). Three categories of trains are considered: intercity, sprinter, and freight trains, with different acceleration, deceleration, and dynamic characteristics. Four global[1] routes (identified by colors: blue for intercity trains, green for sprinter trains, and red for freight trains) are determined and graphically presented in the lower part of Figure 5.6, in terms of origin, intermediate stop, destination, and number of trains per hour. Sprinter trains stop at all stations; intercity and freight trains stop only at the origin and destination stations. We consider one hour of traffic based on a regular-interval timetable, with 15 trains.

Each train is given a randomly generated primary delay time c_f^{pri} at its origin. More specifically, we consider 10 delay cases of the primary delays following a 3-parameter Weibull distribution. The delay distributions differ per train category, and the following parameters in the form of [scale, shape, shift] are used: 1) for intercity trains, [394,

[1] A global route identifies the origin and destination of a train service, but does not specify tracks and platforms used in station areas. The tracks and platforms used in a station area are described as local routes.

2.27, 315]; 2) for sprinter trains, [235, 3.00, 186]; 3) for freight trains, [1099, 2.62, 885]. These values come from fitting to real-life data as explained in Corman et al. (2011b).

5.6.2 Experimental results

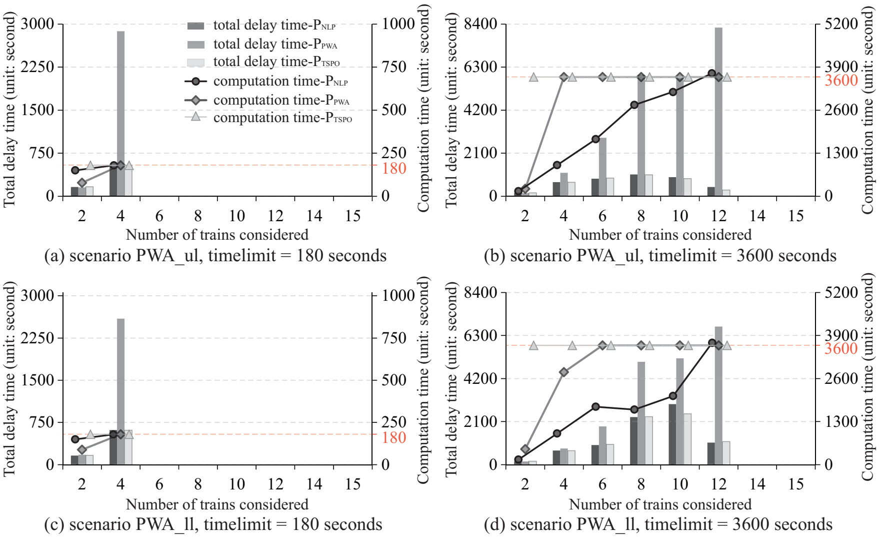
(1) Performance evaluation of the $\mathrm{P_{NLP}}$ problem, the $\mathrm{P_{PWA}}$ problem, and the $\mathrm{P_{TSPO}}$ problem

In this section, we use the Dutch test case introduced in Section 5.6.1 to evaluate the overall performance of the three proposed optimization approaches, from the point of view of effectiveness and efficiency.

We assess the performance of the three proposed optimization approaches on multi-scale instances, i.e., considering several instances with different numbers of trains (ranging from 2 to 15, a subset of the 15 trains described in Figure 5.6) and with heterogeneous traffic. We here consider two computation time limits (i.e., 180 and 3600 seconds) for all three proposed optimization problems, and we output the best feasible solution obtained within each given computation time limit. A large set of TSPOs (i.e., Set_1 in Table 5.4) is used here for the $\mathrm{P_{TSPO}}$ problem, due to its good solution quality, as will be discussed in Section 5.6.2(3). Moreover, we consider two scenarios for the $\mathrm{P_{PWA}}$ problem regarding the upper and lower line fitting methods used for approximating the nonlinear constraints, indicated as "PWA_ul" and "PWA_ll", as will be explained in Section 5.6.2(2). We terminate the genetic algorithm in the upper level of the two-level approach after 10 generations.

In some experiments of the $\mathrm{P_{PWA}}$ problem, we cannot obtain any feasible solution within the given computation time limit; therefore, in Figure 5.7, we particularly report the average results of the three proposed optimization approaches respectively for the corresponding feasible cases of the $\mathrm{P_{PWA}}$ problem. The bars indicate the total train delay time, and refer to the Y-axis on the left-hand side, and the lines (with symbols) indicate the actual computation time, and refer to the Y-axis on the right-hand side. A missing bar/line means that no feasible solution is found for the given instance. Figure 5.7(a) and (b) correspond to the "PWA_ul" scenario of the $\mathrm{P_{PWA}}$ problem, and Figure 5.7(c) and (d) correspond to the "PWA_ll" scenario. Figure 5.7(a) and (c) illustrate the results obtained within 180 seconds of computation time, and Figure 5.7(b) and (d) give the results obtained within 3600 seconds.

We can see that the solution quality of the $\mathrm{P_{PWA}}$ problem is the worst in most instances, as the dark gray bars are much higher than the other bars, even when the computation time is extended to 3600 seconds. The solution quality of the $\mathrm{P_{NLP}}$ problem and the $\mathrm{P_{TSPO}}$ problem is similar in most instances; the largest deviation is less than 33% (corresponding to a delay time of 151 seconds). When focusing on the computational



(a) scenario PWA_ul, timelimit = 180 seconds (b) scenario PWA_ul, timelimit = 3600 seconds

(c) scenario PWA_ll, timelimit = 180 seconds (d) scenario PWA_ll, timelimit = 3600 seconds

Figure 5.7: Results of the three optimization approaches, corresponding to the feasible cases of the P_{PWA} approach

efficiency, the P_{NLP} problem appears to perform better on small-scale instances, because the black line (with dots) is mostly lower than the light gray line (with triangles) for the instances with less than 10 trains, as is shown in Figure 5.7(b) and (d).

As the P_{NLP} problem and the P_{TSPO} problem can obtain feasible solutions for all delay cases, we next focus on all the results of the 10 delay cases to further evaluate the performance of these two optimization approaches, instead of only considering the corresponding feasible cases of the P_{PWA} problem. Figure 5.8 comparatively presents the results of these two models, as an average of the 10 delay cases, in terms of the objective value (i.e., the total train delay time), the actual computation time, and the improvement in solution quality. Figure 5.8(a) has the same structure as Figure 5.7. In Figure 5.8(b), each black (white) bar indicates the average improvement in solution quality for each instance, when comparing the P_{NLP} solution with the P_{TSPO} solution obtained within 180 (3600) seconds respectively, i.e., $\frac{P_{NLP}\,\text{solution} - P_{TSPO}\,\text{solution}}{P_{NLP}\,\text{solution}} \times 100\%$. A positive value means that the solution quality of the P_{TSPO} problem is better, while a negative value implies a better solution quality of the P_{NLP} problem.

As illustrated in Figure 5.8, the solution quality of the P_{NLP} problem and the P_{TSPO} problem differs among instances. Regarding the instances with a larger number of trains (i.e., 8-15 trains), much better solutions are found by the P_{TSPO} problem within the computation time limit, attaining a 30% improvement in the solution quality at most. The P_{TSPO} solution found within 180 seconds is even better than the P_{NLP} solution obtained by consuming a longer computation time (which extends to 3600 seconds). In the other instances with smaller scales, the P_{NLP} problem performs better, as a solution with a smaller train delay time can be found. Although the P_{NLP} problem

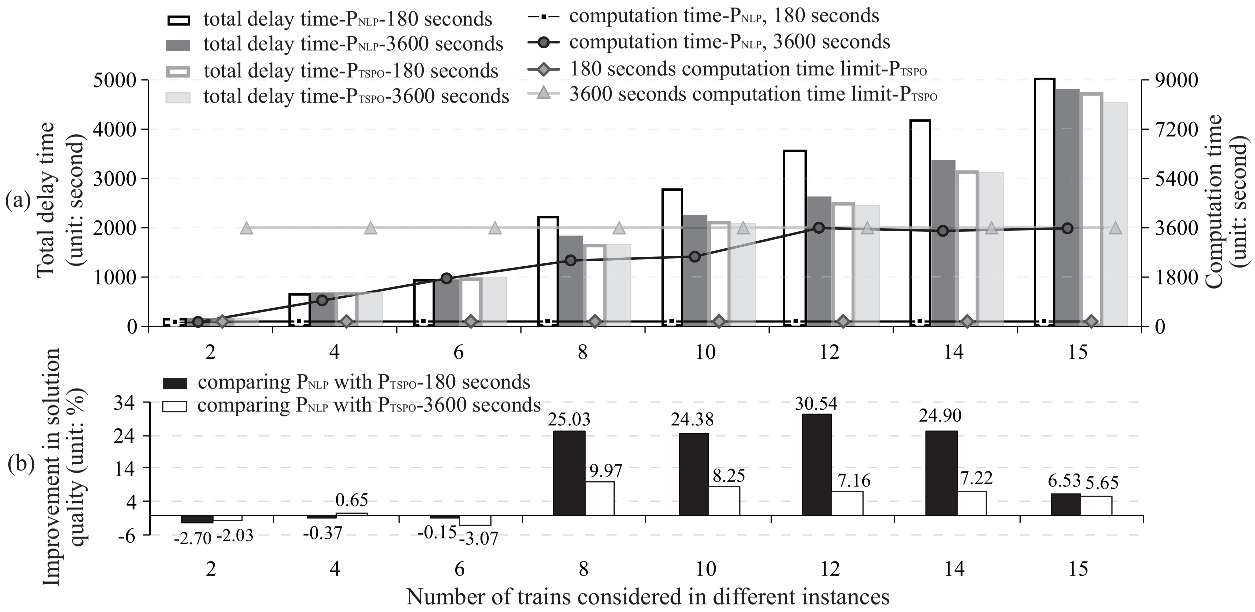


Figure 5.8: Results of the $\mathrm{P_{NLP}}$ problem and the $\mathrm{P_{TSPO}}$ problem

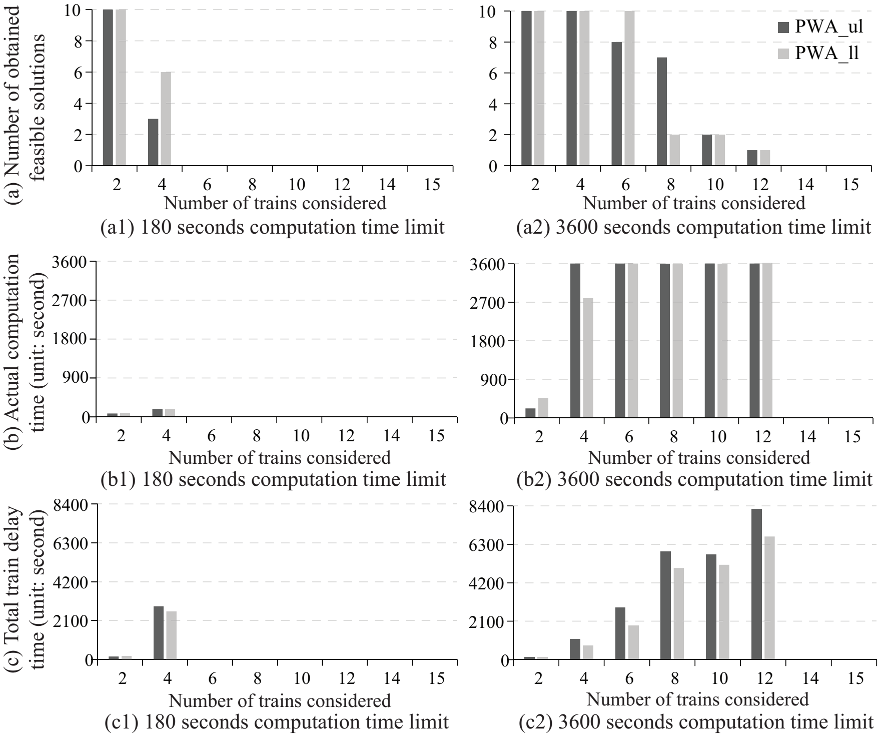
can find better solutions in small-scale instances, in comparison, the P_{TSPO} solution obtained within 180 seconds of computation time is still satisfactory. The P_{TSPO} solution is 3% worse at most than the P_{NLP} solution, which is relatively small when comparing to the 30% improvement of the P_{TSPO} problem achieved for larger-scale instances. Overall, the performance of the P_{TSPO} problem is the best, as a solution with a good quality can be found efficiently (within 180 seconds). Moreover, the train timetables (dispatching solutions) and the speed-space graphs obtained by the P_{NLP} problem and the P_{TSPO} problem for the Dutch test case are provided in Figure A.1 of Appendix A.2.

(2) Further analysis of the experimental results of the P_{PWA} problem

We now study the solution quality and computational efficiency of the P_{PWA} problem by considering different line fitting methods (namely the upper and lower line fitting methods, as illustrated in Figure 5.3), and we also analyze the resulting approximation errors. As discussed before, in order to guarantee the feasibility of the approximated constraints, we only use the lower line fitting method in Figure 5.3(b) to approximate (5.29). Regarding the approximation of (5.31), we consider both the upper and lower line fitting methods, which results in two scenarios, indicated as "PWA_ul" and "PWA_ll" respectively, and we further explore the impact of the line fitting method on the solution quality. We also use the Dutch railway network in Figure 5.6 as test bed, and we consider different instances with different numbers of trains (ranging from 2 to 15, a subset of the 15 trains described in Figure 5.6) and with heterogeneous traffic.

The CPLEX solving process of the P_{PWA} problem is terminated by considering a given computation time limit (i.e., 180 seconds and 3600 seconds), and we then output the best feasible solution obtained within the given computation time limit. Figure 5.9 illustrates the relevant results of "PWA_ul" and "PWA_ll" for each computation time limit, indicated as dark bars and light bars respectively. A missing bar means that no feasible solution is found for the instance within the given computation time limit. Figure 5.9(a) gives the number of the obtained feasible solutions, out of the 10 delay cases. Figure 5.9(b) and (c) present the actual computation time and the objective value as an average of the 10 delay cases.

The optimal solution can be obtained when considering only 2 trains (and 4 trains in "PWA_ll" scenario as well), as the actual computation time of these instances is less than the given computation time limit. For the other instances, the optimality cannot be guaranteed. A longer computation time leads to better objective values and a larger number of cases for which a feasible solution can be attained. No feasible solution can be obtained within 180 seconds for the instances with more than 4 trains, and no feasible solution is obtained within 3600 seconds for the instances with more than 12 trains. Moreover, "PWA_ll" yields a better performance in most instances, as it attains more feasible solutions, relatively shorter computation times, and smaller objective function values.



(a1) 180 seconds computation time limit

(a2) 3600 seconds computation time limit

(b1) 180 seconds computation time limit

(b2) 3600 seconds computation time limit

(c1) 180 seconds computation time limit

(c2) 3600 seconds computation time limit

Figure 5.9: Results of the $\mathrm{P_{PWA}}$ problem, for "PWA_ul" and "PWA_ll"

The approximation errors of "PWA_ul" and "PWA_ll" for different constraints of the $\mathrm{P_{PWA}}$ problem are presented in Figure 5.10, as the percentage, i.e.,

$$\frac{|\text{approximated value} - \text{actual value}|}{\text{actual value}} \times 100\%,$$

and as an average of the 10 delay cases. The errors caused by approximating (5.31a) and (5.31b) lead to a deviation for calculating L^{cru} in (5.30), so we directly analyze the deviation value (approximation error) of L^{cru} in (5.30). The (blue) diamond, (green) square, (pink) dot, and (orange) triangle symbols indicate the approximation errors in the final solution for (5.30), (5.29a), (5.29b), and (5.29c) respectively. The dark small symbols indicate the approximation error of the solution obtained within 180 seconds of computation time, and the light large symbols represent the approximation error of the solution obtained within 3600 seconds. A missing symbol means that no feasible solution is found within the given computation time limit, i.e., the dark small symbols for the instances considering more than 4 trains and the light large symbols for the instances with 14-15 trains.

As illustrated in Figure 5.10, the performance of "PWA_ll" and "PWA_ul" differs among instances, i.e., "PWA_ll" performs better for the instances with 2, 4, 10, and 12 trains, while "PWA_ul" performs better for the instances with 6 and 8 trains. How-

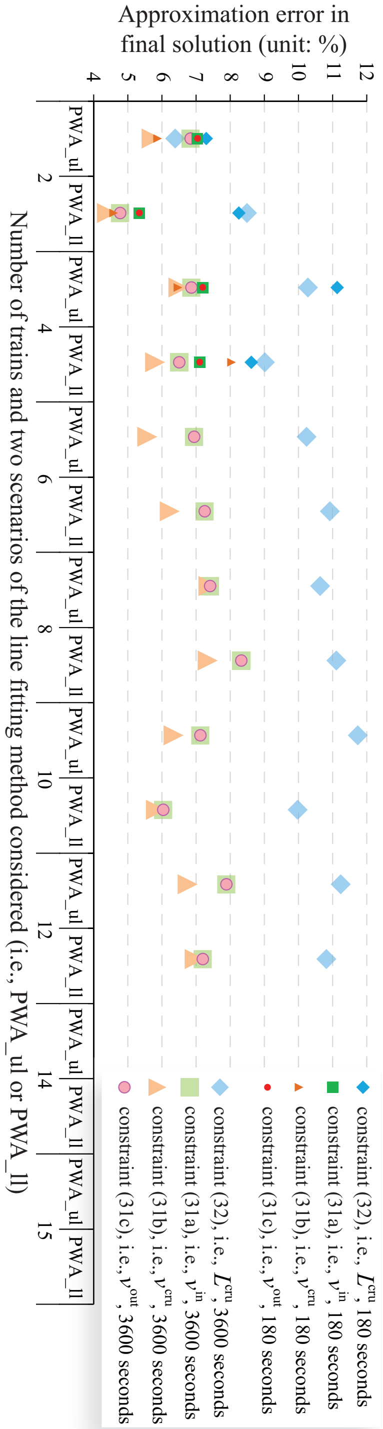


Figure 5.10: Approximation errors of the constraints in the PWA problem, for “PWA_ul” and “PWA_ll”

ever, “PWA_ll” and “PWA_ul” overall perform similarly, with less than 2.5% difference of errors between them. Moreover, the approximation error of (5.30) is larger than that of the others, ranging from 6% to 12%, which results from the different magnitudes of the speed variable (v) and the time variables (a and d). The approximation error of (5.29b) is the smallest, and it ranges from 4% to 8%. For reducing the errors further, we can consider a PWA approximation using more affine subfunctions, and follow the approach described in Section 5.4.2.

Furthermore, we analyze the number of constraint violations caused by the PWA approximation. Regarding (5.29), no constraint is violated, as we apply the lower line fitting method to keep a smaller (positive) approximated value of the train speed than its actual value. For (5.30), around 5% (ranging from 4.2% to 5.0% for “PWA_ll” and from 4.1% to 5.6% for “PWA_ul”) of the constraints is violated, in the sense that the approximated distance that a train travels in the cruising phase is larger than the actual distance that a train can move.

In summary, from all perspectives, i.e., the solution quality, the computational efficiency, the feasibility, and the errors, the $\mathrm{P_{PWA}}$ problem does not seem to perform good enough for addressing the integrated problem of traffic management and train control.

(3) Further analysis of the experimental results of the $\mathrm{P_{TSPO}}$ problem

We now study the impact of the TSPOs generated in the preprocessing step on the solution quality. Six sets of TSPOs are given by considering different discrete speed values for different train categories; they are presented in Table 5.4, denoted as Set_1, ..., Set_6 respectively. Note that intercity and sprinter trains use the same speed pattern in each set. The number of the discrete speed values used in Set_1, ..., Set_6 is decreasing, which implies that the resolution of the train speed becomes lower and less TSPOs are available. The total number of TSPOs corresponding to the 6 sets is provided in columns 4-5 of Table 5.4. Column 4 gives the total number of TSPOs per train per block section, i.e., summing up the number of TSPOs for each train on each block section; column 5 presents the number of possibilities of combining the TSPOs for the train services, which indicates the scale of the feasible solution space.

Figure 5.11 illustrates the results of the 6 sets as a function of the computation time, in particular, the total train delay time on average of the 10 delay cases. Note that the CPLEX solving process is terminated by considering 8 computation time limits ranging from 180 to 3600 seconds, and the best feasible solution obtained within each given computation time limit is presented. The 6 sets are distinguished by colors: green, blue, purple, pink, orange and yellow for Set_1, ..., Set_6 respectively. For each set, the result with fixed train orders is drawn as a solid line and the result considering variable train orders is indicated by a dashed line. Each line presents an initial solution (represented by a star) and secondary solutions (indicated by dot and square symbols)

Table 5.4: Six sets of TSPOs generated by using different discrete speed values, see Figure 5.4 for more information

	Discrete speed values for intercity train and sprinter train (unit: km/h)	Discrete speed values for freight train (unit: km/h)	Total number of TSPOs per train per block section	Number of all possibilities of combining the TSPOs
Set_1	$\{0,40,60,80,90,100,110,120,130\}$	$\{0,20,30,40,50,60,70,80\}$	16402	5.70×10^{50}
Set_2	$\{0,40,70,90,100,110,120,130\}$	$\{0,20,40,50,60,70,80\}$	12370	5.28×10^{46}
Set_3	$\{0,40,70,90,110,120,130\}$	$\{0,20,40,60,70,80\}$	9084	3.16×10^{43}
Set_4	$\{0,40,70,100,120,130\}$	$\{0,20,50,70,80\}$	6332	5.56×10^{39}
Set_5	$\{0,40,100,130\}$	$\{0,40,80\}$	2388	8.27×10^{28}
Set_6	$\{0,40,130\}$	$\{0,40,80\}$	1278	6.71×10^{19}

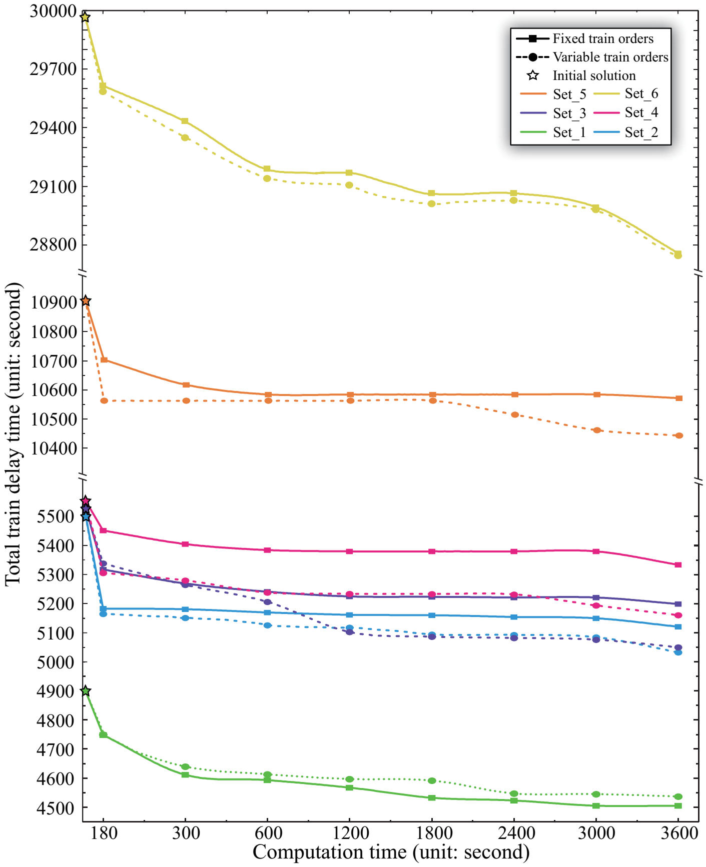


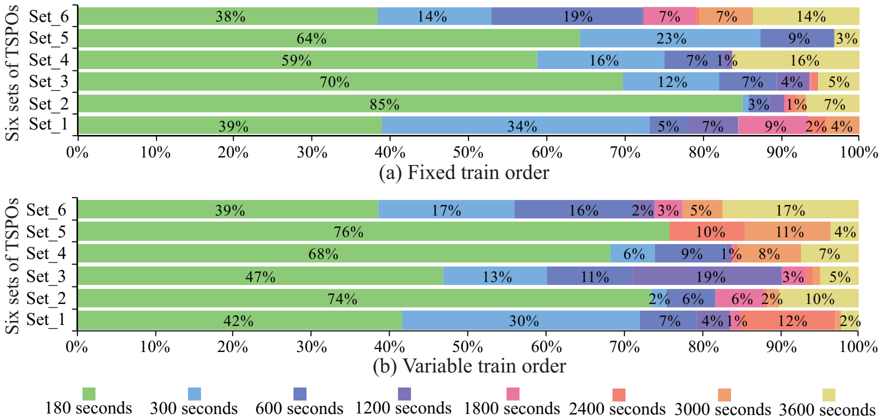
Figure 5.11: Total train delay time of the 6 sets as a function of computation time

as a function of computation time. Recall that the initial solution is obtained by considering a fixed full TSPO for each train on each block section and then improved to generate the secondary solutions by considering a larger set of multiple TSPOs.

We first focus on the results with fixed train orders, represented as solid lines in Figure 5.11. The initial optimal solution considering a fixed full TSPO for each train on each block section (i.e., each train is required to run as fast as possible with respect to the safety, technical, and operational requirements) can be obtained efficiently (i.e., in less than 6 seconds). The initial solution is further improved to generate the secondary solutions by considering a larger set of multiple TSPOs. As shown, when focusing on one set, the total delay time decreases as a function of the computation time, implying an improvement in solution quality. *This demonstrates the benefit of integrating traffic management and train control, i.e., train delays can be reduced by managing train speed.* Moreover, focusing on all the 6 sets, the total delay time increases in both the initial solution and the secondary solutions, if fewer discrete speed values are considered. So the total delay time increases with a decreasing resolution of the train speed in Set_1, ..., Set_6 sequentially. This results from the reduced solution space, i.e., the reduced number of TSPOs available. The improvement in train delay time of Set_1 (the best/significant one with the lowest total delay time) is 3.14% at 180 seconds, and it increases to 8.08% when extending the computation time to 3600 seconds.

When comparing with the results with fixed train orders, the solution quality considering variable train orders is better for Set_2, ..., Set_6, i.e., the dashed line is mostly lower than the corresponding solid line. For Set_1, which contains the largest number of TSPOs among the 6 sets, the result considering variable train order is worse than that for fixed train orders. This may result from the large solution space caused by the huge number of TSPOs and various possibilities of train orders, and the high sensitivity of the solutions to the train speed. The sensitivity of the solutions to the train speed is higher with an increasing number of TSPOs. Therefore, the MILP solver is unable to effectively explore the solution space (regarding train speed) within a given computation time limit. When reducing the solution space by fixing train orders, the MILP solver has a higher chance to explore the solution space more efficiently within the same time limit. To conclude, we may consider variable train orders for the case with a low resolution of the train speed, and fixed train orders for the case with a high train speed resolution, in order to obtain a better solution within a given computation time limit.

Figure 5.12(a) and (b) present the percentage of improvement in solution quality from the initial solution as a function of computation time, for the cases considering fixed and variable train order respectively. This percentage of improvement is calculated by the formula $\frac{\Phi_{\ell-1}-\Phi_{\ell}}{\Phi_1-\Phi_9}$, for $\ell = 2,...,9$. Note that ℓ is the index of the computation time limits considered, i.e., $\ell = 1,...,9$ represents 0 (initial solution), 180, 300, 600, 1200, 1800, 2400, 3000, and 3600 seconds of computation time limits respectively;



(a) Fixed train order

(b) Variable train order

Figure 5.12: Percentage of the improvement in solution quality as a function of computation time for the 6 sets

and Φ_ℓ indicates the total train delay time at the corresponding computation time limit ℓ. For instance, the delay time of the initial solution for Set_1 is 4902 seconds (i.e., $\Phi_1 = 4902$), which is reduced to 4748 and 4506 seconds in the secondary solutions obtained at 180 and 3600 seconds of computation time respectively (i.e., $\Phi_2 = 4748$ and $\Phi_9 = 4506$); the percentage of improvement in solution quality within 180 seconds is then $\frac{\Phi_1-\Phi_2}{\Phi_1-\Phi_9} = \frac{4902-4748}{4902-4506}$=39%. In each figure, the percentages of improvement in solution quality at the 8 computation time limits are respectively drawn from the left to the right using different colors, and each horizontal bar represents a set of TSPOs.

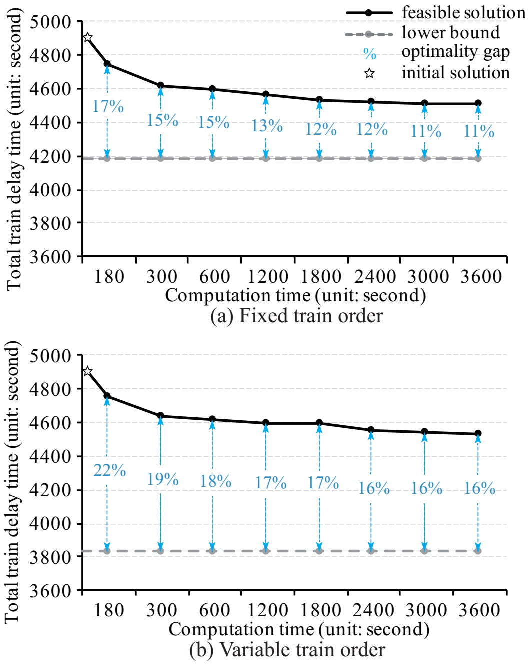
As illustrated, the green region (i.e., the improvement in solution quality at the first 180 seconds) occupies most of the space for each bar, ranging from 38% to 85% in Figure 5.12(a) and from 39% to 76% in Figure 5.12(b). When expanding the focus to the green and light blue portions, the percentage of the quality improvement from 0 to 300 seconds of computation time is more than a half for all the sets, achieving 52% - 87% in Figure 5.12(a) and 56% - 76% in Figure 5.12(b). This implies that a significant improvement in solution quality can be achieved efficiently. Although the solution quality can be improved by considering a longer computation time, the improvement is not as significant as that achieved within the first 180 seconds. Hence, practically, it is not a good choice to consume a much longer computation time for obtaining a small improvement only.

Although a significant improvement from the initial solution can be achieved efficiently, the solution quality is still unknown, i.e., how far is the solution away from the optimal one (an estimation of the optimality gap). Therefore, we generate lower bounds for the $\mathrm{P_{TSPO}}$ problem to assess their solution quality. Note that the so-called lower bound here is not physically feasible and therefore not the best lower bound.

Figure 5.13(a) and (b) illustrate the obtained lower bounds, feasible solutions, and the corresponding estimation of optimality gaps[1] as a function of the computation time, and as an average of 10 delay cases, considering fixed and variable train orders respectively. The largest set of TSPOs (i.e., Set_1) is used for computing the lower bounds, due to its good solution quality. The best feasible solutions obtained within the given computation time limits are represented by black dots (connected by a solid line), and the lower bound is indicated by a horizontal dashed line. The percentage in blue color indicates the optimality gap. To calculate these lower bounds, we have neglected train acceleration and deceleration characteristics, i.e., we assume that a train can suddenly and instantly accelerate or decelerate to any given speed value (listed in row 2 of Table 5.4). This leads to a reduction of the optimization problem to identify an optimal cruising speed for each train on each block section, as the incoming speed and the outgoing speeds do not affect the final results anymore. The calculation of the lower bounds is also an MILP problem, so we use the CPLEX solver to get them.

The lower bound of the case with fixed train order in Figure 5.13(a) is tighter than that of the case considering variable train order in Figure 5.13(b), which results from

[1] Note that the gap between the feasible solution obtained and the lower bound is considered as an estimation of the optimality gap.



(a) Fixed train order

(b) Variable train order

Figure 5.13: Lower bounds, feasible solutions, and estimation of optimality gaps

the reduced solution space by fixing train orders. As shown in Figure 5.13(a), when fixing the train orders, the optimality gap is 17% within 180 seconds of computation time, and it is then reduced to 11% by extending the computation time to at most 3600 seconds. In comparison, the optimality gap of the case considering variable train orders is larger, ranging from 22% to 16%, as shown in Figure 5.13(b).

5.6.3 Discussion

We here derive the main conclusions, sketched quantitatively in Figure 5.14, from the viewpoints of solution feasibility (constraint violation), solution quality, computational efficiency (reported approximately), and applicability for large-scale instances (measured by the total number of the cases, for which at lease one feasible solution is obtained within the given computation time limit). The center indicates the worst performance for all the four items.

In view of the solution feasibility and the applicability for large-scale instances, the P_{NLP} problem and the P_{TSPO} problem have a similar performance, as they can find feasible solutions for all instances (and for all delay cases, even the instance with 15 trains). These two approaches perform better than the P_{PWA} approach, because some constraints are violated in the P_{PWA} solution, and for some large-scale instances no

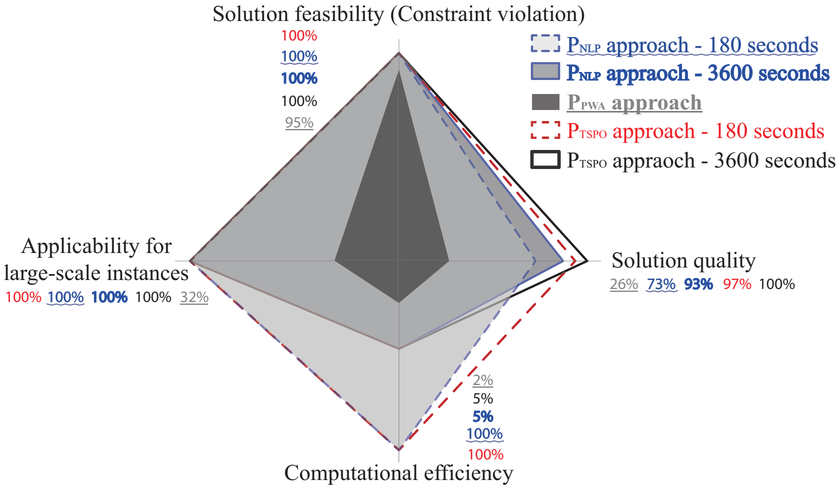


Figure 5.14: Overview of the performance of the three proposed approaches

feasible solution is obtained for the P_{PWA} problem within the given computation time limit.

Regarding the solution quality, the P_{PWA} approach is also the worst among the three approaches. The solution quality of the P_{NLP} problem and the P_{TSPO} problem differs among instances. The P_{TSPO} approach has a better performance on the instances with a larger number of trains, and the P_{NLP} approach performs a little better on the instances with a smaller number of trains. Overall, the P_{TSPO} solution is better than the P_{NLP} solution, achieving a 23.2% improvement, corresponding to a total delay time of 3727 seconds, within 180 seconds of computation time. The improvement of the P_{TSPO} approach in solution quality reduces to 6.7%, when extending the computation time to 3600 seconds.

From the perspective of computational efficiency, the P_{PWA} approach does not yield any feasible solution within the given time limit for many instances, so the computational efficiency of the P_{PWA} approach is recognized as being the worst. In the experiments, feasible solutions (having satisfactory quality in fact) can always be found by the P_{TSPO} approach within the shortest computation time limit (i.e., 180 seconds), and a significant improvement (with respect to the corresponding initial solution) in solution quality can be achieved efficiently. Regarding the computational efficiency of the P_{NLP} approach, feasible solutions can also be obtained within the given computation time limit, but with a worse quality in comparison with the P_{TSPO} solution. As computation time limits are considered and feasible solutions can be found by both the P_{NLP} approach and the P_{TSPO} approach for all delay cases, within 180 seconds of computation time, we cannot draw conclusions on their computational efficiency. Their computational efficiency is therefore reflected by the quality of the solutions obtained within the given computation time limits.

Computational efficiency is a key factor for addressing real-time problems, and the problem of integrating real-time traffic management and train control is such a case. Therefore, the overall performance of the P_{TSPO} approach is recognized as being the best, as a solution with better and satisfactory quality can be found efficiently (within 180 seconds), see Figure 5.8. Using a larger set of TSPOs for the P_{TSPO} approach leads to a better solution. The results show that we could consider to fix the train orders when using a larger set of TSPOs, in order to better explore a smaller solution space regarding the train speed within a time limit.

The experimental results demonstrate the benefits of integrating traffic management and train control. The benefit is reflected by the reduced train delays, i.e., train delays can be reduced by managing the train speed and by changing the train orders. In our test case, the consideration of multiple TSPOs leads to respectively 3.14% and 8.08% reduction of train delays for Set_1 within 180 and 3600 seconds of computation time, and the consideration of changing train orders results in an additional 1.59% improvement in the solution quality for Set_2, as discussed in Section 5.6.2(3).

5.7 Conclusions

In this chapter, we have tackled the integration of real-time traffic management and train control by using mixed-integer nonlinear programming (MINLP) and mixed-integer linear programming (MILP) methods. Three optimization approaches are developed, i.e., one MINLP problem (P_{NLP}) and two MILP problems (P_{PWA} and P_{TSPO}), for delivering both a train dispatching solution (i.e., binary/integer decisions on a set of times, orders, and routes to be followed by trains) and a train control solution (i.e., train speed trajectories following nonlinear dynamics) simultaneously. A preprocessing step is used for the P_{TSPO} problem to generate multiple TSPOs by considering discrete speed values, in order to restrict the search only to an efficient subset of all possible TSPOs. In these optimization problems, the train speed is considered variable, and the blocking time of a train on a block section dynamically depends on its real speed. Regarding the solution approaches, we have presented a two-level approach for solving the P_{NLP} problem and proposed a custom-designed two-step approach for solving the P_{TSPO} problem. The performance of the three proposed optimization approaches has been evaluated comparatively from the viewpoints of solution feasibility, solution quality, computational efficiency, and applicability for large-scale instances, based on a real-world test case adapted from the Dutch railway network. According to the experimental results, the P_{TSPO} approach overall yields the best performance among the three optimization approaches, as it is able to exploit the solution space efficiently. Moreover, the benefits of integrating real-time traffic management and train control are demonstrated: for the given test case, the train delay can be reduced up to 8% by managing the train speed and by changing the train orders.

For future research, a comprehensive system could be developed based on the proposed optimization approaches to integrate the multiple steps in the solving procedure, e.g., the preprocessing step for generating a set (or an efficient subset) of the possible TSPOs, the solving step to solve the optimization problem, and the displaying step to show train timetables and speed-space graphs.

Chapter 6

Integration of traffic control and train control-Part 2: Extensions towards energy-efficient train operations[1]

We here still study the integration of real-time traffic management and train control by using mixed-integer nonlinear programming (MINLP) and mixed-integer linear programming (MILP) approaches. In this chapter, aiming at energy-efficient train operation, we extend the three optimization problems proposed in Chapter 5 by means of introducing energy-related formulations.

This chapter is organized as follows. Section 6.1 gives a detailed introduction of the integrated problem of real-time traffic management and train control, focusing on the energy-related aspects. In Section 6.2, after introducing the notations used in the mathematical formulations, we calculate the energy consumption of the train motion for accelerating trains and for overcoming resistances respectively. Then, formulations for calculating the utilization of the energy obtained by braking trains are constructed. In Section 6.3, the experimental results based on a real-world railway network are given for evaluating the performance of the optimization approaches, exploring the trade-off between train delay and energy consumption, and investigating the benefits of regenerative braking. Moreover, we examine the quality of the train speed trajectories obtained by the proposed integrated optimization approaches, by means of comparing them with the train speed trajectories obtained by using the detailed nonlinear train models as proposed by Wang et al. (2013); Liu and Golovitcher (2003); Khmelnitsky (2000). Finally, Section 6.4 ends the chapter with conclusions.

[1] With minor updates, this chapter has been published in “Luan, X., Wang, Y., De Schutter, B., Meng, L., Lodewijks, G., Corman, F. (2018). Integration of real-time traffic management and train control for rail networks-Part 2: Extensions towards energy-efficient train operations. *Transportation Research Part B: Methodological*, 115, 72-94.”

6.1 Introduction

Railway transport systems are of crucial importance for the competitiveness of national or regional economy as well as for the mobility of people and goods. To maintain the environmental advantage and business benefits of railway sectors, targets have been set by the International Union of Railways (UIC, 2012) to reduce the carbon dioxide (CO_2) emissions and energy consumption from train operations by 50% and 30% respectively in 2030, compared to 1990. Such policies reflect an increasing concern for sustainability and energy efficiency. Consequently, energy-efficient train operation is attracting more and more attention, as it is seen as the most important measure to reduce the environmental impacts and the costs used to power trains.

In railway transport systems, the energy efficiency is greatly influenced by the train operation strategy, which consists of the operational train timetables and the applied driving actions. The former relates to the real-time traffic management problem, i.e., (re-)scheduling train routes, orders, and passing times at stations, aiming at adjusting the impacted schedules from perturbations and reducing negative consequences. The latter concerns the train control problem, i.e., optimizing the sequence of driving regimes (maximum acceleration, cruising, coasting, and maximum braking) and the switching points between the regimes, with the aim of minimizing energy consumption. As discussed in Chapter 5, the two problems are closely related to each other. In order to achieve energy-efficient train operation, one of the most promising options is to jointly consider the two problems, i.e., (re-)constructing a timetable in a way that not only allows different driving actions, but enables eco-driving actions (resulting in better energy performance). This comes from, e.g., avoiding unneeded accelerating and braking actions, which do not only lead to trains delays, but also unnecessary waste of energy. Another promising option is to incorporate regenerative braking, so that the energy generated by braking trains can be further utilized for accelerating trains, and then the overall energy consumption of train operations decreases. As a result, to compute energy-efficient train trajectories and to further achieve the energy efficiency of train operations, the focus on only train delay is not enough; approaches that not only include train delays but also evaluate energy consumption and consider regenerative energy utilization are desired.

In most studies of the real-time traffic management problem, train delay is a commonly used objective, and any dynamics-related objective, such as energy consumption, cannot be directly considered, due to the disregard of train dynamics, However, the objective of energy consumption is considered only in train control studies (see Lu and Feng, 2011; Wang et al., 2013). In Chapter 5, the integration of the two problems has been addressed, and three integrated optimization approaches have been developed to consider both traffic-related properties (i.e., a set of times, orders, routes to be followed by trains) and train-related properties (i.e., speed trajectories) at the same time, focusing on delay recovery only. These integrated optimization approaches build up a good foundation and enable us to introduce energy-related formulations and to focus on delay recovery and energy efficiency at the same time.

In this chapter, we focus on the train control part of the integrated optimization approaches while including energy-related formulations. We first introduce the evaluation of energy consumption into the integrated optimization problems. To calculate the energy consumption, a set of linear constraints is proposed for the $\mathrm{P_{TSPO}}$ problem; for the $\mathrm{P_{NLP}}$ and $\mathrm{P_{PWA}}$ problems, the resistance function with a quadratic term of the train speed is approximated with a piecewise constant function, in order to maintain the nature of these two optimization approaches. In addition, we consider the option of regenerative braking and present linear formulations to calculate the utilization of the energy obtained through regenerative braking. With the inclusion of the energy-related formulations, we consider two objectives, i.e., delay recovery and energy efficiency, by using a weighted-sum formulation and an ε-constraint formulation. Train coasting is not included due to the concern of problem complexity; however, a coasting phase can be introduced by assuming a piecewise constant deceleration function of the cruising speed, as discussed in Chapter 6 We use the Dutch test case to conduct experiments, just as in Chapter 5. We compare the performance of the optimization approaches and investigate the trade-off between train delay and energy consumption. By our approaches, train delay and energy consumption can be reduced at the same time through managing the train speed, by up to 4.0% and 5.6% respectively. This demonstrates the benefit of the integration and shows great potential for energy efficiency of train operations. Moreover, the benefit of regenerative braking is shown. In our case study, when applying regenerative braking, up to 53.3% of the kinetic energy can be stored, and up to 46.6% of the stored energy is re-utilized for train acceleration, which further leads to a 22.9% reduction of the total energy consumption. In the experiments, the proposed optimization approaches can obtain feasible solutions (with good quality) of the train delay and energy consumption minimization problem, for a single direction along a 50 km corridor with 9 stations and 15 trains each hour within a computation time of 3 minutes.

6.2 Mathematical formulation

In Section 6.2.1, we first describe the notations used for formulating, modeling, and optimizing the energy-related aspects. Section 6.2.2 discusses and formulates the energy consumption of the train motion for accelerating trains and for overcoming resistances respectively. As incorporating regenerative braking is an effective way to achieve energy efficiency, in Section 6.2.3, we consider the possibility of regenerative braking and provide formulations for calculating the utilization of the regenerative energy obtained by braking trains. In this chapter, we still follow the assumptions proposed in Section 5.3 of Chapter 5.

6.2.1 Notations

We use the notations in Table 5.1, with the additional sets, subscripts, input parameters, and decision variables given in Table 6.1 to formulate the train energy consumption. Movement of a train on a block section is considered to be made up by an incoming phase, (accelerating or braking from a starting speed to a cruising speed), a cruising phase with a constant cruising speed, and an outgoing phase (accelerating or braking from the cruising speed to an exit speed of the block section).

Table 6.1: Sets, subscripts, input parameters, and decision variables

	Subscripts and sets
K	set of regions, i.e., electric regions, $\lvert K\rvert$ is the number of regions
κ	region index, $\kappa \in K$
$Y_{f,i,j}$	set of options of train speed profile vectors that train f may follow on cell (i,j), $\lvert Y_{f,i,j}\rvert$ is the number of train speed profile options (TSPOs) for train f on cell (i,j)
b	TSPO index, $b_{f,i,j} \in \{1,...,\lvert Y_{f,i,j}\rvert\}$
E^{r}_{κ}	set of cells in region κ where trains can utilize regenerative energy
	Input parameters
m_f	mass of train f
$y^{\mathrm{in}}_{f,i,j,b}$	b^{th} incoming speed of train f on cell (i,j)
$y^{\mathrm{cru}}_{f,i,j,b}$	b^{th} cruising speed of train f on cell (i,j)
$y^{\mathrm{out}}_{f,i,j,b}$	b^{th} outgoing speed of train f on cell (i,j)
$y_{f,i,j,b}$	b^{th} train speed profile vector, $y_{f,i,j,b} \in Y_{f,i,j}$, train speed profile vector $y_{f,i,j,b} = \begin{bmatrix} y^{\mathrm{in}}_{f,i,j,b} & y^{\mathrm{cru}}_{f,i,j,b} & y^{\mathrm{out}}_{f,i,j,b} \end{bmatrix}^{\top} \in Y_{f,i,j}$
$L^{\mathrm{in}}_{f,i,j,b}$	distance that train f runs over on cell (i,j) in the incoming phase in the b^{th} train speed profile vector $y_{f,i,j,b}$
$L^{\mathrm{out}}_{f,i,j,b}$	distance that train f runs over on cell (i,j) in the outgoing phase in the b^{th} train speed profile vector $y_{f,i,j,b}$
$\zeta_{1,f,i,j,b},\ldots,$ $\zeta_{6.f,i,j,b}$	logical parameters to indicate the relation of the incoming, cruising, outgoing speed, and switching speed v^{turn}_f in the b^{th} train speed profile vector $y_{f,i,j,b}$ (see the explanation in Table 5.2)
$r_{1,f,i,j}, r_{2,f,i,j},$ $r_{3,f,i,j}$	coefficients of the total resistance function for train f on cell (i,j)
$\eta_{i,j,p,k}$	recuperation coefficient for utilizing the regenerative energy between cells (i,j) and (p,k) depending on the distance between the two cells
	Decision variables
$L^{\mathrm{cru}}_{f,i,j}$	distance that train f runs through on cell (i,j) in the cruising phase

continued on next page

continued from previous page

Symbol	Description
$\vartheta_{f,i,j,b}$	binary variables, $\vartheta_{f,i,j,b} = 1$ if the corresponding train speed vector $y_{f,i,j,b}$ is used by train f on cell (i,j), and otherwise $\vartheta_{f,i,j,b} = 0$
$J^{\text{acc\_in}}_{f,i,j}, J^{\text{acc\_out}}_{f,i,j}$	energy consumption for accelerating train f in the incoming and outgoing phases on cell (i,j)
$J^{\text{res\_in}}_{f,i,j}, J^{\text{res\_cru}}_{f,i,j}, J^{\text{res\_out}}_{f,i,j}$	energy consumption for overcoming the resistances of train f in the incoming, cruising, and outgoing phases on cell (i,j)
$J^{\text{reg\_in}}_{f,i,j}, J^{\text{reg\_out}}_{f,i,j}$	regenerative energy obtained by braking train f in the incoming and outgoing phases on cell (i,j)
$u_{f,f',i,j,p,k}$	energy generated by braking train f on cell (i,j) and further used for accelerating train f' on cell (p,k)

We model train movements over block sections, such that their timing can be determined, and the energy can be related to the accelerating, cruising, and braking actions occuring in the train movements. Compared to Chapter 5, some variables for calculating the energy consumption and the regenerative energy utilization are newly added, e.g., $J^{\text{acc\_in}}_{f,i,j}$, $J^{\text{res\_in}}_{f,i,j}$, $J^{\text{reg\_in}}_{f,i,j}$, and $u_{f,f',i,j,p,k}$. Basically, these variables are a consequence of the interactions among the key variables for formulating the traffic and train related decisions introduced in Chapter 5, i.e., arrival time variables a, departure time variables d, and train speed variables v, for all trains in the network, with respect to the work formula, Newton's second law of motion, the formulas of the uniformly accelerating and decelerating motions, and operational requirements.

Note that the maximum acceleration and deceleration depend on the traction and braking force. In the literature, researchers either consider the tractive force as a well-defined function of the speed and the control input (Howlett, 2000), or assume a constant power (then the tractive force is a function of the speed, e.g., Howlett, 2000), or assume a constant acceleration (Wang et al., 2016). As this chapter aims to include the minimization of the energy consumption into the integrated optimization problems proposed in Chapter 5 and to compare the integrated optimization approaches in a clear way, we still assume a piecewise constant acceleration (with a switching point v^{turn}_{f}) and a constant deceleration for each train category, just as in Chapter 5. So a train follows a uniform acceleration and deceleration motion in a given speed interval. According to the equation $q = r + m \cdot \alpha$ (where q, r, m, and α indicate the tractive force, resistance force, train mass, and train acceleration respectively), the introduction of resistance would result in a larger tractive force (compared with the case where the resistance is neglected), which further has impact on the energy consumption. We consider a piecewise constant acceleration for the train motion, and the resistance is taken into account for determining this piecewise constant acceleration, i.e., ensuring that the tractive force used for accelerating trains and for overcoming the resistance together is technically feasible (not greater than the maximum tractive force).

6.2.2 Optimization of energy consumption

Energy is mostly consumed for accelerating trains and for overcoming the resistance in a train movement. In previous studies on the train control problem (Wang et al., 2013; Wang and Goverde, 2016), the energy used for a train that travels from position x_1 to position x_2 is calculated by using the following equation:

$$J = \int_{x_1}^{x_2} q(x) \cdot \mathrm{d}x = \int_{x_1}^{x_2} m \cdot \alpha \cdot \mathrm{d}x + \int_{x_1}^{x_2} r^{\mathrm{x}}(x) \cdot \mathrm{d}x, \tag{6.1}$$

where J indicates the work (energy consumption), $q(\cdot)$ and $r^{\mathrm{x}}(\cdot)$ indicate the tractive force and the resistance force respectively, given as a function of the distance x, m is the train mass, and α indicates the train acceleration. By using the formulas $\mathrm{d}x = v \cdot \mathrm{d}t$ and $\mathrm{d}v = \alpha \cdot \mathrm{d}t$, we can rewrite (6.1) as follows:

$$J = \int_{v_1}^{v_2} m \cdot v \cdot \mathrm{d}v + \int_{v_1}^{v_2} \frac{r(v) \cdot v}{\alpha} \cdot \mathrm{d}v, \tag{6.2}$$

where the resistance force $r(\cdot)$ is given as a function of the train speed v, and v_1 and v_2 indicate the train speeds at the positions x_1 and x_2 respectively. The first term of (6.2) in fact indicates the energy used for accelerating the train, and the second term calculates the energy consumption for overcoming the resistance.

In Section 6.2.2(1) and Section 6.2.2(2), we discuss and formulate the energy consumption in these two usages respectively. Thus, the problem of modeling train movements is that we have to relate energy consumption to resistance and train speed, resistance to train speed, and departure and arrival times (which are the optimization variables for the traffic management problem) to distance and train speed.

(1) Energy used for accelerating trains

With integral calculation, the energy consumption for train acceleration, i.e., the first term of (6.2), can be easily calculated as $\frac{m}{2} \cdot (v_2^2 - v_1^2)$, which is in fact the difference of the train kinetic energy when changing the speed of a train with a mass m from v_1 to v_2. Based on the notations in Tables 5.1 and 6.1 and the formulations proposed in Chapter 5, we add the following constraints to determine the energy consumption used for accelerating trains in the incoming and outgoing phases respectively:

$$J_{f,i,j}^{\mathrm{acc\_in}} = \max\left\{0, \frac{1}{2} \cdot m_f \cdot [(v_{f,i,j}^{\mathrm{cru}})^2 - (v_{f,i,j}^{\mathrm{in}})^2]\right\}, \quad \forall f \in F, (i,j) \in E_f \tag{6.3}$$

$$J_{f,i,j}^{\mathrm{acc\_out}} = \max\left\{0, \frac{1}{2} \cdot m_f \cdot [(v_{f,i,j}^{\mathrm{out}})^2 - (v_{f,i,j}^{\mathrm{cru}})^2]\right\}, \quad \forall f \in F, (i,j) \in E_f. \tag{6.4}$$

In the cruising phase, energy is only used for maintaining a constant cruising speed, i.e., overcoming the resistance; so no energy is consumed for train acceleration in the cruising phase.

Constraints (6.3)-(6.4) contain quadratic terms of the speed variables $v^{\text{in}}_{f,i,j}$, $v^{\text{cru}}_{f,i,j}$, and $v^{\text{out}}_{f,i,j}$. These quadratic terms will not affect the nature of the three optimization problems proposed in Chapter 5, i.e., the P_{NLP} problem is still an NLP problem and the P_{PWA} and P_{TSPO} problems are still MILP problems. Therefore, with the inclusion of (6.3)-(6.4), the solution approaches proposed in Section 5.5 of Chapter 5 can still be used to solve these problems.

(2) Energy used for overcoming the resistance

The energy used for overcoming resistance while changing the train speed from v_1 to v_2 can be formulated as $\int_{v_1}^{v_2} \frac{r(v)\cdot v}{\alpha} \cdot \text{d}v$, i.e., the second time of (6.2), where $r(\cdot)$ indicates the resistance force as a function of the train speed v and α indicates the train acceleration.

Regarding the resistance force $r(\cdot)$, it is typically assumed that there are two categories of resistances for trains, i.e., the train resistance and the line resistance. Besides the common impact factor of the infrastructure (train) characteristics, the train resistance only depends on the train driving strategy (i.e., the operating speed), and the line resistance is mostly determined by the characteristics of the rail network (track). In the studies on the train control problem (Davis, 1926; Brünger and Dahlhaus, 2008; Wang et al., 2013; Hansen et al., 2017), the resistance force $r(\cdot)$ is commonly expressed as a quadratic function of the speed, i.e., $r_{1,x} \cdot v^2 + r_{2,x} \cdot v + r_{3,x}$, where $r_{\rho,x}$ for $\rho \in \{1,2,3\}$ are non-negative coefficients that depend on the train characteristics and the rail network (track) characteristics. We assume that the gradients and curve radii are constant for each cell (block section); the difference of the gradient and curve radii within a cell is neglected. The tunnel resistance occurs in the cells inside the tunnels (even if the cell is partially inside the tunnel) and is equal to zero for the cells completely outside the tunnels. With this assumption, the coefficients $r_{\rho,x}$ for $\rho \in \{1,2,3\}$ are then constant for each train category on each cell; thus, they could be rewritten as $r_{\rho,f,i,j}$ for $\rho \in \{1,2,3\}$ for train f on cell (i,j). As a result, we can express the total resistance of train f on cell (i,j) as $r_{f,i,j}(v_{f,i,j}) = r_{1,f,i,j} \cdot v^2_{f,i,j} + r_{2,f,i,j} \cdot v_{f,i,j} + r_{3,f,i,j}$, which only depends on its running speed $v_{f,i,j}$. Note that the total resistance is a strictly increasing quadratic function of the speed.

Let us define a function $\Xi(v) = \frac{r_1}{4} \cdot v^4 + \frac{r_2}{3} \cdot v^3 + \frac{r_3}{2} \cdot v^2$, where $r(v) \cdot v$ is the derivative of the function $\Xi(v)$, i.e., $[\Xi(v)]' = r(v) \cdot v$. Then, we could calculate the integral formula $\int_{v_1}^{v_2} \frac{r(v)\cdot v}{\alpha} \cdot \text{d}v$ as $\frac{\Xi(v_2)-\Xi(v_1)}{\alpha}$, which computes the energy used for overcoming resistance when accelerating a train from speed v_1 to speed v_2 at a steady acceleration α. Note that the function $\Xi(v)$ should be train and block section dependent, due to the train and block section specified coefficients $r_{\rho,f,i,j}$ for $\rho \in \{1,2,3\}$.

By assuming a piecewise constant acceleration (with a switching point v^{turn}_f) and a constant deceleration for each train category, a train follows a uniform acceleration and deceleration motion in a given speed interval. We apply the formulation $\frac{\Xi(v_2)-\Xi(v_1)}{\alpha}$ to compute the energy used by train f on cell (i,j) for overcoming the resistance in the incoming phase, meanwhile taking the piecewise constant acceleration into account; the formulation is given as follows:

$$J^{\text{res\_in}}_{f,i,j} = \begin{cases} \frac{\Xi_{f,i,j}(v^{\text{cru}}_{f,i,j}) - \Xi_{f,i,j}(v^{\text{in}}_{f,i,j})}{\alpha_{1,f,i,j}}, & \text{if } v^{\text{in}}_{f,i,j} \leq v^{\text{cru}}_{f,i,j} \leq v^{\text{turn}}_{f} \\ \frac{\Xi_{f,i,j}(v^{\text{cru}}_{f,i,j}) - \Xi_{f,i,j}(v^{\text{in}}_{f,i,j})}{\alpha_{2,f,i,j}}, & \text{if } v^{\text{turn}}_{f} \leq v^{\text{in}}_{f,i,j} < v^{\text{cru}}_{f,i,j} \\ \frac{\Xi_{f,i,j}(v^{\text{turn}}_{f}) - \Xi_{f,i,j}(v^{\text{in}}_{f,i,j})}{\alpha_{1,f,i,j}} + \frac{\Xi_{f,i,j}(v^{\text{cru}}_{f,i,j}) - \Xi_{f,i,j}(v^{\text{turn}}_{f})}{\alpha_{2,f,i,j}}, & \\ & \text{if } v^{\text{in}}_{f,i,j} < v^{\text{turn}}_{f} < v^{\text{cru}}_{f,i,j} \\ 0, & \text{if } v^{\text{in}}_{f,i,j} > v^{\text{cru}}_{f,i,j} \end{cases} \quad \forall f \in F, (i,j) \in E_f. \tag{6.5}$$

A formulation similar to (6.5) can also be constructed for calculating the energy consumption $J^{\text{res\_out}}_{f,i,j}$ in the outgoing phase. For the sake of compactness, we do not report those details here.

In the cruising phase, a train follows a uniform motion at a certain cruising speed; so the resistance force does not change and can be easily computed by $r_{f,i,j}(v^{\text{cru}}_{f,i,j})$. Therefore, based on the work formula $J = r \cdot x$ (where J, r, and x indicate the work, force, and distance that a train travelled respectively), we can formulate the energy used by train f on cell (i,j) in the cruising phase as follows:

$$\begin{aligned} J^{\text{res\_cru}}_{f,i,j} = r_{f,i,j}(v^{\text{cru}}_{f,i,j}) \cdot L^{\text{cru}}_{f,i,j} = (r_{1,f,i,j} \cdot {v^{\text{cru}}_{f,i,j}}^2 + r_{2,f,i,j} \cdot v^{\text{cru}}_{f,i,j} \\ + r_{3,f,i,j}) \cdot L^{\text{cru}}_{f,i,j}, \quad \forall f \in F, (i,j) \in E_f. \end{aligned} \tag{6.6}$$

Constraints (6.5)-(6.6) contain either a quartic term of the train speed or a product term of the speed and distance. The inclusion of these two equations changes the nature of the P_{NLP} problem and the P_{PWA} problem and leads to difficulties in solving these two problems (i.e., the resulting problems cannot be solved directly). However, as a set $Y_{f,i,j}$ of train speed profile options (TSPOs) is pre-defined in a preprocessing step for the P_{TSPO} problem, that problem is still an MILP problem.

In order to help readers understand the unchanged nature of the P_{TSPO} problem, we reformulate (6.5) and (6.6) for the P_{TSPO} problem as follows:

$$\begin{aligned} J^{\text{res\_in}}_{f,i,j} = & \sum_{b=1}^{|Y_{f,i,j}|} \vartheta_{f,i,j,b} \cdot \zeta_{1,f,i,j,b} \cdot \zeta_{4,f,i,j,b} \cdot \frac{\Xi_{f,i,j}(y^{\text{cru}}_{f,i,j,b}) - \Xi_{f,i,j}(y^{\text{in}}_{f,i,j,b})}{\alpha_{1,f,i,j}} \\ & + \sum_{b=1}^{|Y_{f,i,j}|} \vartheta_{f,i,j,b} \cdot \zeta_{1,f,i,j,b} \cdot \zeta_{3,f,i,j,b} \cdot \frac{\Xi_{f,i,j}(y^{\text{cru}}_{f,i,j,b}) - \Xi_{f,i,j}(y^{\text{in}}_{f,i,j,b})}{\alpha_{2,f,i,j}} \\ & + \sum_{b=1}^{|Y_{f,i,j}|} \vartheta_{f,i,j,b} \cdot \zeta_{1,f,i,j,b} \cdot (1 - \zeta_{3,f,i,j,b}) \cdot (1 - \zeta_{4,f,i,j,b}) \\ & \cdot \left[\frac{\Xi_{f,i,j}(v^{\text{turn}}_{f}) - \Xi_{f,i,j}(y^{\text{in}}_{f,i,j})}{\alpha_{1,f,i,j}} + \frac{\Xi_{f,i,j}(y^{\text{cru}}_{f,i,j}) - \Xi_{f,i,j}(v^{\text{turn}}_{f})}{\alpha_{2,f,i,j}} \right] \\ & \forall f \in F, (i,j) \in E_f. \end{aligned} \tag{6.7}$$

$$\begin{aligned} J^{\text{res\_cru}}_{f,i,j} = & \sum_{b=1}^{|Y_{f,i,j}|} \vartheta_{f,i,j,b} \cdot \left(L^{\text{cell}}_{i,j} - L^{\text{in}}_{f,i,j,b} - L^{\text{out}}_{f,i,j,b} \right) \cdot \left(r_{1,f,i,j} \cdot {y^{\text{cru}}_{f,i,j,b}}^2 \right. \\ & \left. + r_{2,f,i,j} \cdot y^{\text{cru}}_{f,i,j,b} + r_{3,f,i,j} \right), \forall f \in F, (i,j) \in E_f. \end{aligned} \tag{6.8}$$

The binary variable $\vartheta_{f,i,j,b}$ and the input parameters $y^{\text{in}}_{f,i,j,b}$, $y^{\text{cru}}_{f,i,j,b}$, $y^{\text{out}}_{f,i,j,b}$, $L^{\text{cell}}_{i,j}$, etc. have all been introduced in Chapter 5, and they are described in Table 5.1. The speed indicators $\zeta_{1,f,i,j,b}$, ..., $\zeta_{6,f,i,j,b}$ are used in Chapter 5 and explained in Table 5.2. A formulation similar to (6.7) can also be constructed for calculating the energy consumption $J^{\text{res\_out}}_{f,i,j}$ in the outgoing phase; we skip the details here for the sake of compactness. With the inclusion of the linear constraint (6.7) and (6.8), the nature of the P_{TSPO} problem will not change, i.e., it is still an MILP problem.

To address the difficulties in solving the P_{NLP} problem and the P_{PWA} problem with constraints (6.5) and (6.6), we can approximate the resistance function $r_{f,i,j}(\cdot)$ by using a piecewise constant function with 2 affine parts and with constant values $r^{\text{cs}}_{1,f,i,j}$ and $r^{\text{cs}}_{2,f,i,j}$. As a result, we can use the following formulations to calculate the energy used by train f on cell (i,j) for overcoming resistance in the incoming and cruising phases respectively:

$$J^{\text{res\_in}}_{f,i,j} = \begin{cases} \frac{r^{\text{cs}}_{1,f,i,j}}{2\alpha_{1,f,i,j}} \cdot \left[(v^{\text{cru}}_{f,i,j})^2 - (v^{\text{in}}_{f,i,j})^2\right], \text{if } v^{\text{in}}_{f,i,j} \leq v^{\text{cru}}_{f,i,j} \leq v^{\text{turn}}_{f} \\ \frac{r^{\text{cs}}_{2,f,i,j}}{2\alpha_{2,f,i,j}} \cdot \left[(v^{\text{cru}}_{f,i,j})^2 - (v^{\text{in}}_{f,i,j})^2\right], \text{if } v^{\text{turn}}_{f} \leq v^{\text{in}}_{f,i,j} < v^{\text{cru}}_{f,i,j} \\ \frac{r^{\text{cs}}_{1,f,i,j}}{2\alpha_{1,f,i,j}} \cdot \left[(v^{\text{turn}}_{f,i,j})^2 - (v^{\text{in}}_{f,i,j})^2\right] + \frac{r^{\text{cs}}_{2,f,i,j}}{2\alpha_{2,f,i,j}} \cdot \left[(v^{\text{cru}}_{f,i,j})^2 - (v^{\text{turn}}_{f,i,j})^2\right], \\ \qquad\qquad\qquad\qquad \text{if } v^{\text{in}}_{f,i,j} < v^{\text{turn}}_{f} < v^{\text{cru}}_{f,i,j} \\ 0, \quad \text{if } v^{\text{in}}_{f,i,j} > v^{\text{cru}}_{f,i,j} \end{cases}, \tag{6.9}$$
$$\forall f \in F, (i,j) \in E_f$$

$$J^{\text{res\_cru}}_{f,i,j} = \begin{cases} r^{\text{cs}}_{1,f,i,j} \cdot L^{\text{cru}}_{f,i,j}, & \text{if } v^{\text{cru}}_{f,i,j} \leq v^{\text{turn}}_{f} \\ r^{\text{cs}}_{2,f,i,j} \cdot L^{\text{cru}}_{f,i,j}, & \text{if } v^{\text{cru}}_{f,i,j} > v^{\text{turn}}_{f} \end{cases}, \quad \forall f \in F, (i,j) \in E_f. \tag{6.10}$$

These two equations derive from the work formula $J = r \cdot x$, where J and r indicate the energy consumption and the resistance force and $x = \frac{v_2^2 - v_1^2}{2\alpha}$ computes the distance travelled by a train for accelerating from speed v_1 to v_2 in a uniform acceleration motion. A formulation similar to (6.9) can also be constructed for calculating the energy consumption $J^{\text{res\_out}}_{f,i,j}$ in the outgoing phase. For the sake of compactness, we do not report those details here.

We consider two objectives: one is for delay recovery, just as in Chapter 5, i.e., reducing the sum over all trains of the mean absolute delay time at all visited stations:

$$Z^{\text{delay}} = \sum_{f \in F} \sum_{(i,j) \in E^{\text{stop}}_f} \frac{\left|d_{f,i,j} - w_{f,i,j} - D_{f,i,j}\right|}{\left|E^{\text{stop}}_f\right|}, \tag{6.11}$$

and another one is to achieve energy efficiency, i.e., reducing the total energy consumption for both accelerating trains and overcoming the resistance:

$$Z^{\text{energy}} = \sum_{f \in F} \sum_{(i,j) \in E_f} \left(J^{\text{acc\_in}}_{f,i,j} + J^{\text{acc\_out}}_{f,i,j} + J^{\text{res\_in}}_{f,i,j} + J^{\text{res\_cru}}_{f,i,j} + J^{\text{res\_out}}_{f,i,j}\right), \tag{6.12}$$

where $J^{\text{acc\_in}}_{f,i,j}$ and $J^{\text{acc\_out}}_{f,i,j}$ are computed in (6.3)-(6.4), and $J^{\text{res\_in}}_{f,i,j}$, $J^{\text{res\_cru}}_{f,i,j}$, and $J^{\text{res\_out}}_{f,i,j}$ are calculated by using (6.7)-(6.8) for the P_{TSPO} problem and by following (6.9)-(6.10) for the other two problems.

For multi-objective optimization problems, the weighted-sum formulation and the ε-constraint formulation are commonly used. Therefore, aiming at both delay recovery and energy efficiency, we use the following two ways:

1) One way is to minimize the weighted sum of the two objectives. Then, the overall objective function can be presented as

$$\min Z = \iota^{\text{delay}} \cdot Z^{\text{delay}} + \iota^{\text{energy}} \cdot Z^{\text{energy}}, \tag{6.13}$$

where the weights ι^{delay} and ι^{energy} are used to balance their importance, and for normalization as well.

2) Another way is to minimize the energy consumption with respect to a given upper bound $Z^{\text{delay\_ub}}$ of the train delay, formulated as

$$\min Z^{\text{energy}} \tag{6.14a}$$

$$\text{s.t.} \quad Z^{\text{delay}} \leq Z^{\text{delay\_ub}}. \tag{6.14b}$$

Additionally, in this research, the constraints proposed in Chapter 5 for the three proposed optimization problems should also be included.

6.2.3 Utilization of regenerated energy

An option for further improving the energy efficiency of train operations is to incorporate regenerative braking, where the kinetic energy of a running train can be converted into electrical energy when the train brakes. This electrical energy can be fed back to the catenary system for immediately accelerating other trains or stored in energy storage devices (e.g., batteries, super-capacitors, and flywheels) for train acceleration when required. Regenerative braking is a mature technology. In practice, it has been used in urban rail transit systems and also in railway transportation systems (UIC, 2002). The use of regenerative braking decreases the overall energy consumption of the train motion and changes the optimal solution to the energy-efficient train control (operation) problem. Therefore, in this section, we present formulations to calculate the regenerative energy and to determine the utilization of the regenerative energy, for railway and metro systems that are equipped with this technology.

Aiming at maximizing the use of the regenerative energy and at minimizing the need of on-board resistors (which are used for dissipating the regenerative energy that cannot be used within the system), energy storage technologies have been well studied in the literature (see the review papers by Khaligh and Li (2010); González-Gil et al. (2013)) and applied in the railway industry (see the recent review paper by Ghaviha et al., 2017). Therefore, we consider the use of energy storage systems, and then the regenerative energy can be used for train acceleration when it is required. Energy storage systems can by divided into two types, i.e., on-board energy storage systems and the wayside (or stationary) energy storage systems, which result in different rules for utilizing the regenerative energy, explained as follows:

- For the on-board energy storage systems, i.e., storage devices that are installed on the trains, a train is able to temporarily store its own braking energy and re-utilize it in the next acceleration stages. So, the energy generated by braking a train on a block section can only be further used by the train itself. This kind of system is operated in some countries, e.g., Portugal (a light rail network in the south of Lisbon Meinert, 2009) and Germany (light rail networks in Mannheim Steiner et al., 2007).

- For the wayside (or stationary) energy storage systems, where the storage devices are installed along the track, the surplus regenerated energy could be absorbed and delivered when it is required for accelerating other trains in the same electric region. So, the energy generated by braking a train on a block section can be further used by other trains on the block sections that are in the same electric region as the block section where the the energy is generated. In practice, this kind of system is commonly used in urban rail transit systems, e.g., metro systems in France (Boizumeau et al., 2011), Germany, China, and Spain (Siemens, 2011), and also used or tested for railway transport systems in some countries, e.g., Spain (Garcia-Tabares et al., 2011) and Japan (Shimada et al., 2010; Ogasa, 2010).

The installation of the on-board energy storage devices will greatly increase the train mass and will require a large space, so this option is sometimes used for light rail vehicles and seldom used for railway trains. In comparison, the wayside energy storage systems have less weight and little influence on operation and maintenance (Su et al., 2016). Therefore, we consider wayside energy storage systems to illustrate the construction of the formulations. Some modifications should be made in the following proposed formulations in order to be make them suited for a case with an on-board energy storage system. We discuss these modifications at the end of this section (before Remark 6.1 on train coasting).

Regenerative energy is the energy converted from kinetic energy into electrical energy while braking. According to the definition of regenerative energy (Scheepmaker and Goverde, 2016), we formulate the energy regenerated by the braking of the train as follows:

$$J_{f,i,j}^{\text{reg\_in}} = -\min\left\{0, \frac{1}{2} \cdot m_f \cdot [(v_{f,i,j}^{\text{cru}})^2 - (v_{f,i,j}^{\text{in}})^2]\right\}, \quad \forall f \in F, (i,j) \in E_f, \tag{6.15}$$

$$J_{f,i,j}^{\text{reg\_out}} = -\min\left\{0, \frac{1}{2} \cdot m_f \cdot [(v_{f,i,j}^{\text{out}})^2 - (v_{f,i,j}^{\text{cru}})^2]\right\}, \quad \forall f \in F, (i,j) \in E_f \tag{6.16}$$

i.e., the reduction of the train kinetic energy while braking for the incoming and outgoing phases respectively. Note that implicitly $J_{f,i,j}^{\text{acc\_in}} \cdot J_{f,i,j}^{\text{reg\_in}} = 0$ and $J_{f,i,j}^{\text{acc\_out}} \cdot J_{f,i,j}^{\text{reg\_out}} = 0$, as a train cannot accelerate and decelerate at the same time.

As discussed, the final energy consumption is not just the difference between the energy used for powering trains and the energy generated by braking trains. We need to consider the rules with regards to the temporal and spatial limitations for utilizing the regenerative energy that result from the installation position of the energy storage systems, as well as the efficiency of the regenerative braking system.

We introduce a non-negative variable $u_{f,f',i,j,p,k}$ that indicates the amount of the energy generated by braking train f on cell (i,j) and then used for accelerating train f' on cell (p,k); moreover, we let $\eta_{i,j,p,k} \in [0,1]$ be the recuperation coefficient, which determines the efficiency of the regenerative braking system between cells (i,j) and (p,k), based on the distance between the two cells.

From a temporal perspective, we ensure that the regenerative energy is available when it is used for powering trains. Therefore, the following constraints is added:

$$\begin{aligned} u_{f,f',i,j,p,k} \leq 0, \quad & \text{if } a_{f,i,j} > a_{f',p,k}, \\ & \forall f \in F, f' \in F, (i,j) \in E_f, (p,k) \in E_{f'}, \end{aligned} \tag{6.17}$$

for guaranteeing that the energy generated by train f on cell (i,j) cannot be used by train f' on cell (p,k), if train f arrives at cell (i,j) after the arrival of train f' at cell (p,k). Constraint (6.17) is an if-then constraint, which can be rewritten as a mixed-integer linear constraint by applying the transformation properties in Bemporad and Morari (1999).

Regarding the spatial limitations, we enforce that the energy generated by braking a train can only be used by accelerating the trains located in the same electric region. Without loss of generality, we select $|K|$ regions in the network, where each region corresponds to an electric region. We then assume that the available energy regenerated in a cell can be used by any trains traveling in the region that the cell belongs to. We denote the sets of cells in the regions as $E^{\mathrm{r}}_1, E^{\mathrm{r}}_2, ..., E^{\mathrm{r}}_{|K|}$. We use the following constraint:

$$\begin{aligned} \textstyle\sum_{f \in F, f' \in F} u_{f,f',i,j,p,k} \leq 0, & \\ \forall (i,j) \in E_f, (p,k) \in E_{f'}, \{(i,j),(p,k)\} \not\subset E^{\mathrm{r}}_{\kappa}, \kappa \in K & \end{aligned} \tag{6.18}$$

to ensure that the regenerative energy on cell (i,j) (or (p,k)) cannot be utilized by any train on cell (p,k) (or (i,j)), if the set of the two cells $\{(i,j),(p,k)\}$ is not a subset of set E^{r}_{κ}, for any $\kappa \in K$, i.e., cells (p,k) and (i,j) are not in the same electric region.

To balance the generation and utilization of the regenerative energy, we have the following constraint:

$$\sum_{f' \in F, (p,k) \in E_{f'}} \frac{u_{f,f',i,j,p,k}}{\eta_{i,j,p,k}} \leq (J^{\mathrm{reg\_in}}_{f,i,j} + J^{\mathrm{reg\_out}}_{f,i,j}), \quad \forall f \in F, (i,j) \in E_f, \tag{6.19}$$

for ensuring that the total usage of the regenerative energy obtained by braking train f on cell (i,j) and further used for accelerating all trains $f' \in F$ on all cells $(p,k) \in E_{f'}$ cannot exceed the available amount of the energy generated by braking train f on cell (i,j). By taking the efficiency of the regenerative braking system into account, we divide the total usage of the regenerative energy by the non-negative recuperation coefficient $\eta_{i,j,p,k}$, as presented on the left-hand side of (6.19). Recall that the coefficient $\eta_{i,j,p,k}$ is given based on the distance between the two cells (i,j) and (p,k).

The amount of the regenerative energy that is further utilized for accelerating trains can be determined by

$$Z^{\text{energy\_reg}} = \sum_{f \in F} \sum_{(i,j) \in E_f} \sum_{f' \in F} \sum_{(p,k) \in E_{f'}} u_{f,f',i,j,p,k}. \tag{6.20}$$

The final energy consumption is then calculated as follows:

$$Z^{\text{energy\_final}} = Z^{\text{energy}} - Z^{\text{energy\_reg}}, \tag{6.21}$$

where the term Z^{energy} is computed by using (6.12), including the energy used for train acceleration and for overcoming resistance. We can still use the weighted sum formulation and the ε-constraint formulation, i.e., minimizing the weighted sum of the train delays and the final energy consumption, as presented in (6.13) through replacing Z^{energy} by $Z^{\text{energy\_final}}$, and only minimizing the final energy consumption in (6.21) with respect to (6.14b).

We now discuss the modifications for calculating the regenerative energy in case of an on-board energy storage system. For implementing the utilization rule, i.e., the regenerative energy can only by utilized by the same train that generates it through braking, two equivalent ways can be used. The first way consists in simply requiring $\sum_{f \in F, f' \in F: f \neq f'} u_{f,f',i,j,p,k} = 0$ for preventing the regenerative energy utilization between two different trains. Alternatively, we could re-define the variable $u_{f,f',i,j,p,k}$ as $u_{f,i,j,p,k}$ for all $f \in F$, $(i,j) \in E_f$, and $(p,k) \in E_f$. Then, in (6.17), (6.19), and (6.20), we remove the condition term $f' \in F$ and replace $(p,k) \in E_{f'}$ by $(p,k) \in E_f$. No matter which way is used, a common change in this case is to remove (6.18), as the spatial limitation is not active. The first way is applicable to both the wayside and on-board energy storage systems, and it is easier for switching or integrating the two types of energy storage systems, at the expense of a larger complexity of the optimization problem. In the case with only the on-board energy storage system, using the second way is a better choice, as its problem complexity is largely reduced.

Remark 6.1: Coasting phase

A coasting phase can be included into the proposed optimization problem by assuming a piecewise constant deceleration that depends on the cruising speed. In other words, we could consider a piecewise constant train deceleration; then, the formulation approach stays similar to the approach that includes the piecewise constant train acceleration. As a similar formulation approach can be followed, we do not present the formulations for train coasting in this chapter. Moreover, we can use the arrival and departure times of a train along its route in the solutions (which have no coasting phase) obtained by applying our integrated optimization methods to further generate an accurate train speed profile option by using train trajectory optimization approaches with the aim of minimizing the energy consumption.

6.3 Case study

6.3.1 Setup

We consider the same Dutch railway network and the same 15 trains as in Chapter 5; we refer to Section 5.6.1 for the description of the test case. Also just as in Chapter 5, each train is given a randomly generated primary delay time c_f^{pri} at its origin, and we consider 10 delay cases of the primary delays following a 3-parameter Weibull distribution. Additionally, we consider 6 electric regions, corresponding to 6 station areas, as depicted in Figure 6.1. The 15 trains considered run in the same direction. No energy can be regenerated at Ut (Utrecht) and Hto (Den Bosch Oost) station (as no train brakes here), and no train can utilize the regenerated energy at Ht (Den Bosch) station (as no train departs from here); therefore, in our case study, no electric region is set up in the Ut, Hto, and Ht station areas. The recuperation coefficient η ranges from 70% to 80% in our experiments.

As this chapter focuses on the energy-related extensions based on the integrated optimization approaches proposed in Chapter 5, the complexity of the optimization problems increases with the inclusion of energy consumption and regenerative braking. Due to the worst performance of the P_{PWA} approach evaluated in Section 5.6.2(1) of Chapter 5, we cannot expect this approach to perform better on the extended optimization problems. Therefore, in the experiments of chapter, we neglect the P_{PWA} approach and only test the other two approaches, i.e., the P_{NLP} approach and the P_{TSPO} approach.

In Section 6.3.2(1), we compare the results of the P_{NLP} approach and the P_{TSPO} approach, aiming at both delay recovery and energy efficiency. Section 6.3.2(2) explores the trade-off between train delay and energy consumption, where the possibility of reducing train delay and energy consumption at the same time is shown. Both the weighted-sum formulation and the ε-constraint formulation are used for representing the two-objective optimization problem of delay recovery and energy efficiency. We further show the benefits of regenerative braking by investigating its impact on the energy consumption in Section 6.3.2(3). In order to examine the solution quality of the proposed optimization approach from a train control perspective, Section 6.3.2(4)

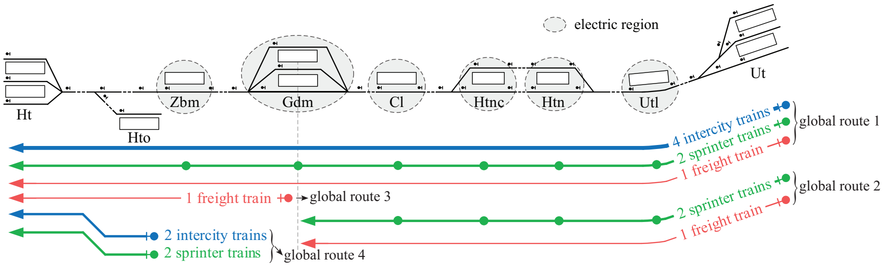


Figure 6.1: Part of the Dutch railway network, with 6 electric regions

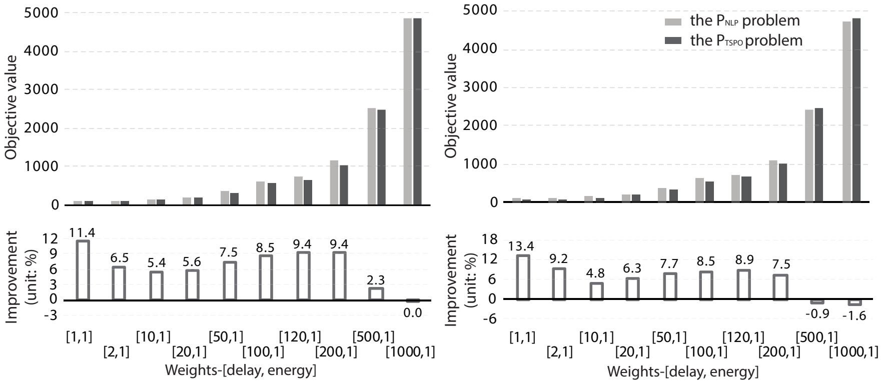
compares the train speed profiles obtained by the proposed integrated optimization approach with those obtained by using the detailed nonlinear train model as proposed by Wang et al. (2013); Liu and Golovitcher (2003); Khmelnitsky (2000). In addition to the constraints caused by the speed limits, maximum acceleration, maximum deceleration, etc., the traffic management problem also presents many operational constraints (i.e., a train should pass a certain place at a certain time, the passing time at a non-stopping station), which should be also considered in the train control problem (Wang et al., 2012). Here we apply a sequential quadratic programming (SQP) approach to solve the resulting nonlinear train control problem. The details for the solution approach will be introduced in Section 6.3.2(4). Note that the solution approaches proposed in Section 5.5 of Chapter 5 are still used to solve the $\mathrm{P}_{\mathrm{NLP}}$ problem and the $\mathrm{P}_{\mathrm{TSPO}}$ problem. Moreover, we additionally report the detailed experimental results of this test case in the online repository (Research Collection ETH Zurich).

We use the SNOPT solver implemented in the MATLAB (R2016a) TOMLAB toolbox to solve the MINLP problem, i.e., the $\mathrm{P}_{\mathrm{NLP}}$ problem. We adopt the IBM ILOG CPLEX optimization studio version 12.6.3 with default settings to solve the MILP problem, i.e., the $\mathrm{P}_{\mathrm{TSPO}}$ problem. The following experiments are all performed on a computer with an Intel® Core$^{\mathrm{TM}}$ i7 @ 2.00 GHz processor and 16GB RAM.

6.3.2 Experimental results

(1) Overall performance of the $\mathrm{P}_{\mathrm{NLP}}$ and $\mathrm{P}_{\mathrm{TSPO}}$ optimization problems

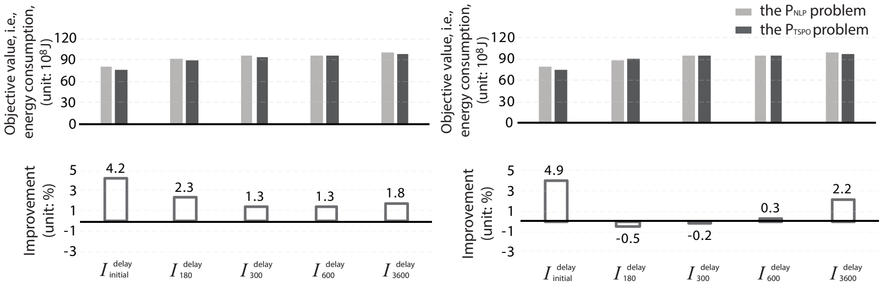
In this section, the overall performance of the $\mathrm{P}_{\mathrm{NLP}}$ and $\mathrm{P}_{\mathrm{TSPO}}$ optimization problems are compared; the results of the weighted-sum formulation and the ε-constraint formulation are presented in Figures 6.2 and 6.3 respectively. For the $\mathrm{P}_{\mathrm{TSPO}}$ optimization problem, the largest set of TSPOs (i.e., Set_1) is considered, which is generated by using the discrete speed values $\{0,40,60,80,90,100,110,120,130\}$ (km/h) for intercity and sprinter trains and $\{0,20,30,40,50,60,70,80\}$ (km/h) for freight trains, as its solution quality is the best among the six sets, discussed in Section 5.6.2(3) of Chapter 5. This set contains 16402 speed profiles per train per block section, which results in 5.70×10^{50} possibilities of combining the speed profiles for all train services. For the weighted-sum formulation, we use 10 weights (indicated in the form of $[\iota^{\mathrm{delay}}, \iota^{\mathrm{energy}}]$, widely ranged, see the X-axis of Figure 6.2) to balance their importance, and for normalization as well. As the weight of the energy consumption ι^{energy} is always set to be 1, we can also use a single weight, denoted as $\iota = \iota^{\mathrm{delay}}$, to describe the multiple choices of weights. An increase of the single weight ι implies that the importance of the train delay increases. For the ε-constraint formulation, we consider 5 upper bounds for the train delay, which stem from the delay time in the initial solution and in the secondary solutions obtained within 180, 300, 600, and 3600 seconds of computation time (refer to Section 5.6.2(3) of Chapter 5), indicated as $I^{\mathrm{delay}}_{\mathrm{initial}}$, I^{delay}_{180}, I^{delay}_{300}, I^{delay}_{600}, and $I^{\mathrm{delay}}_{3600}$ respectively. We consider two computation time limits, i.e., 180 seconds



(a) Computation time of 180 seconds

(b) Computation time of 3600 seconds

Figure 6.2: Comparison of the P$_{\text{NLP}}$ and P$_{\text{TSPO}}$ results, in the case of using the weighted-sum formulation



(a) Computation time of 180 seconds

(b) Computation time of 600 seconds

Figure 6.3: Comparison of the P$_{\text{NLP}}$ and P$_{\text{TSPO}}$ results, in the case of using the ε-constraint formulation

and 3600 seconds, in the case of using the weighted-sum formulation. When using the ε-constraint formulation, the solution is almost never improved after 600 seconds; so we consider 600 seconds as the maximum computation time limit, instead of 3600 seconds.

In Figures 6.2 and 6.3, each bar indicates an average result of 10 delay cases. In the upper portion of each figure, the objective values are given, indicated as gray bars for the P$_{\text{NLP}}$ problem and as black bars for the P$_{\text{TSPO}}$ problem; in the lower portion of each figure, each white bar indicates the average improvement, i.e.,

$$\frac{\text{P}_{\text{NLP}}\,\text{solution} - \text{P}_{\text{TSPO}}\,\text{solution}}{\text{P}_{\text{NLP}}\,\text{solution}} \times 100\%.$$

A positive value means that the P$_{\text{TSPO}}$ solution is better; a negative value implies a better solution quality of the P$_{\text{NLP}}$ problem. Note that in Figure 6.2 we present the objective values, i.e., the real values of the train delay and the energy consumption multiplied by the weights, as we aim at comparing the overall performance of the two approaches; in all other remaining representations of the results (i.e., in Figures 6.3-6.10), we always present the real values of the delay time and the energy consumption.

As illustrated in Figure 6.2, the $\mathrm{P}_{\mathrm{TSPO}}$ problem obtains better solutions in almost all instances, achieving 13.4% improvement in the objective value at most. When the train delay is considered very important, the $\mathrm{P}_{\mathrm{NLP}}$ solution has a quality that is similar to that of the $\mathrm{P}_{\mathrm{TSPO}}$ solution obtained within 180 seconds of computation time in Figure 6.2(a); with a larger computation time of 3600 seconds in Figure 6.2(b), the $\mathrm{P}_{\mathrm{NLP}}$ solution is about 1.6% better than the $\mathrm{P}_{\mathrm{TSPO}}$ solution. From the viewpoints of both solution quality and computational efficiency, we conclude that the $\mathrm{P}_{\mathrm{TSPO}}$ problem performs better in the case of using the weighted-sum formulation.

The results of the two optimization problems by using the ε-constraint formulation are comparatively given in Figure 6.3, which has the same structure as Figure 6.2. When considering the ε-constraint formulation, the performance of the two optimization problems is similar to their performance in the case of using the weighted-sum formulation, but the difference of the two problems in solution quality is smaller. In most instances, the $\mathrm{P}_{\mathrm{TSPO}}$ problem still has a better performance, achieving up to 4.2% improvement in the energy consumption. In a few other instances with 600 seconds of computation time, the $\mathrm{P}_{\mathrm{NLP}}$ problem performs better, but it has only a small (less than 0.5%) improvement in the energy consumption. Overall, the $\mathrm{P}_{\mathrm{TSPO}}$ optimization approach is recognized for having a better performance, by using either the ε-constraint formulation or the weighted-sum formulation.

(2) Exploration of the trade-off between train delay and energy consumption

Due to the good performance of the $\mathrm{P}_{\mathrm{TSPO}}$ approach evaluated in Section 6.3.2(1), we apply this approach to investigate the trade-off between train delay and energy consumption in this section. We present the results of the weighted-sum formulation in Section 6.3.2(2.a). The results of the ε-constraint formulation are analyzed in Section 6.3.2(2.b).

(2.a) The weighted-sum formulation: minimization of both train delay and energy consumption

Figure 6.4(a) and (b) illustrate the deviations of train delay and energy consumption respectively from the initial solution[1], within 180 and 3600 seconds of computation time. The red vertical line (zero line) is the benchmark, representing the initial solution. Each bar indicates an average result of ten delay cases. The gray dashed box in Figure 6.4(a) is a zoom-in, using the interval $[-0.02,\ 1.00]\ \times 10^3$ of the X-axis. The

[1] The initial solution is obtained by considering a fixed full TSPO (train speed profile option) for each train on each block section, which is further improved to generate secondary solutions by considering a larger set of multiple TSPOs. We refer to Section 5.5 for more details.

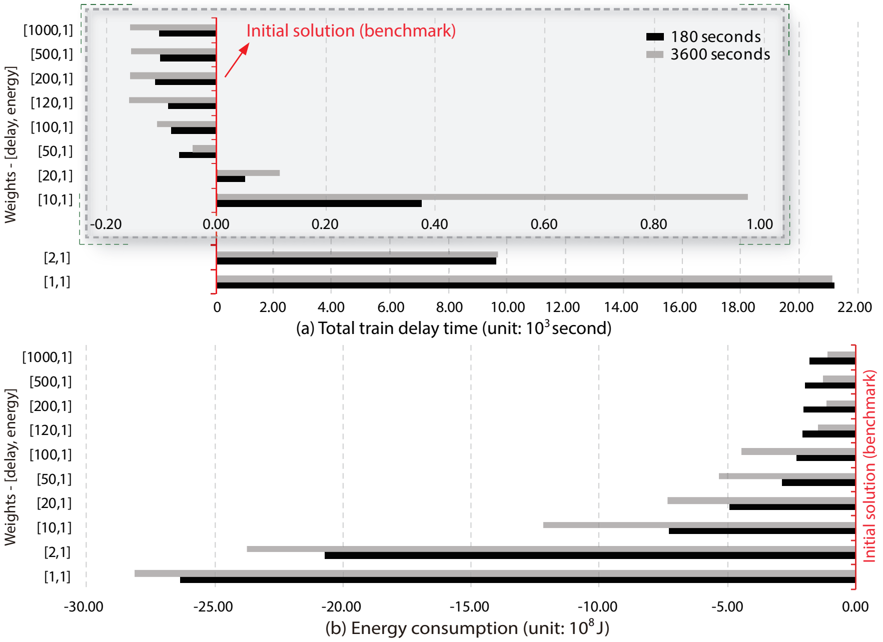


Figure 6.4: Deviations of train delay and energy consumption with respect to the initial solution

Y-axis represents the weights considered. From the bottom to the top of the Y-axis, the importance of the train delay increases. It should be noted that a negative value in Figure 6.4 indicates a reduction (an improvement) from the initial solution, and a positive value means an increase.

As shown, compared with the initial solution, the energy consumption is reduced with a decreasing weight ι, while the total train delay time increases. When the weight ι is not larger than 20, the energy consumption is significantly reduced, at the expense of larger train delay times (corresponding to the positive values in Figure 6.4(a)). In such cases, trains are required to run slowly for saving energy, and train delay is not the determining factor. When the weight ι is not less than 50, the train delay and energy consumption are both reduced with respect to the initial solution. The reduction of the energy consumption becomes smaller with an increasing weight ι, while the reduction of the train delay becomes larger. *The possibility of reducing train delays and saving energy at once by managing train speed is evident, achieving up to a 4.0% and 5.6% reduction of train delay and energy consumption respectively, demonstrating the benefits of the integration of traffic management and train control again.* Moreover, the extension of the computation time to 3600 seconds improves the solution quality, but the improvement is not as significant as that at 180 seconds for most cases.

(2.b) The ε-constraint formulation: energy-saving with respect to an upper bound for train delay

In Figure 6.5, we respectively present the train delay and the energy consumption, obtained by using the ε-constraint formulation (i.e., minimizing the energy consumption with respect to the given upper bound of the train delay), as a function of the computation time. We consider 5 upper bounds for the train delay, indicated as $I^{\text{delay}}_{\text{initial}}$, I^{delay}_{180}, I^{delay}_{300}, I^{delay}_{600}, and I^{delay}_{3600} respectively. We distinguish them by using colors in Figure 6.5. The lighter the color becomes, the stricter the upper bound for the train delay required is, i.e., the requirement of the train delay becomes stricter in a sequence of $I^{\text{delay}}_{\text{initial}}$, ..., I^{delay}_{3600}.

In all cases, a reduction of energy consumption can be always achieved within the first 180 seconds of computation time; however, the energy consumption is almost not reduced anymore after 300 seconds. Since the train delay is considered as a hard constraint, there is little room for its improvement, i.e., the lines of the delay time in Figure 6.5(a) are almost flat. Moreover, the trade-off between the train delay and the energy consumption is clearly shown in Figure 6.5. A stricter upper bound of the train delay leads to less delays (i.e., the lighter lines are lower in Figure 6.5(a)), more energy consumption (i.e., the lighter lines are higher than the darker lines in Figure 6.5(b)), and less saved energy (i.e., the gradient of the darker lines is larger than that of the lighter lines in Figure 6.5(b)).

Overall, the two formulation methods both perform well. However, the ε-constraint formulation requires an appropriate upper bound for the train delay, which is generally hard to determine. On one hand, a tighter upper bound for train delay will lead to a worse performance on the energy consumption, which is reflected in the increase of the energy consumption in Figure 6.5(b), and it may even result in infeasibility of the optimization problem. On the other hand, if we use a looser upper bound, the train delay could be large, even if there is some room for its reduction; therefore, the performance of the train dispatching problem cannot be guaranteed. Moreover, we find solutions where the train delay and the energy consumption are reduced at the same

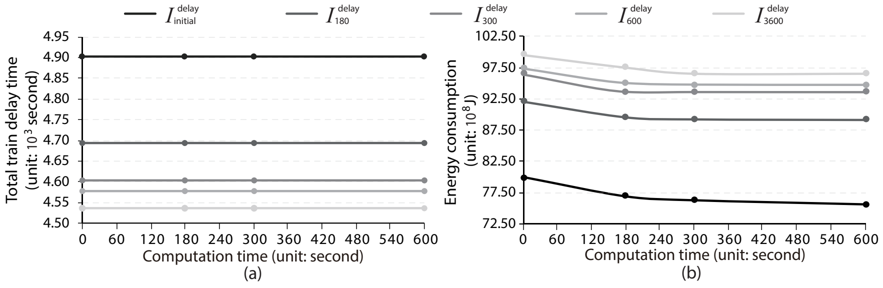


Figure 6.5: Evolution of the train delay and the energy consumption as a function of the computation time

time from the initial solution by using the weighted-sum formulation; however, in the solutions of the ε-constraint formulation, we can only see the reduction of the energy consumption, but no improvement in the train delay. Based on the above reasons, we conclude that the weighted-sum formulation is better and more applicable here than the ε-constraint formulation.

(3) Benefits of regenerative braking

This section compares the results with and without regenerative braking. The composition of the energy consumption in the solutions obtained based on the Dutch test case is illustrated in Figure 6.6. The Y-axis represents the weights considered. From the top to the bottom of the Y-axis, the importance of the train delay increases. The X-axis represents the energy consumption. For each weight, an average result of 10 delay cases is provided. Each black (vertical) bar indicates the total energy consumption without regenerative braking. The light gray and dark gray areas indicate 80% and 60% of this total energy consumption respectively, given as benchmarks. Each dark blue bar indicates the energy used for overcoming the resistance in acceleration and cruising. As a result, the difference between the total energy consumption and the energy consumed for overcoming the resistance in acceleration and cruising is in fact the energy used for train acceleration, which is converted into the train kinetic energy, indicated by a light blue line. A small part of this train kinetic energy is further consumed for overcoming the resistance in deceleration, represented by a light blue bar. By applying regenerative braking, some of this kinetic energy can be stored in energy storage devices, and we use a light green bar to indicate the energy stored during train braking. Then, a part of the energy stored is further re-utilized for train acceleration, which results in a reduction of the total energy consumption from the black bar to the black circle, i.e., the difference between the black bar and the black circle indicates the energy that is re-utilized. The energy loss of the regenerated energy due to system

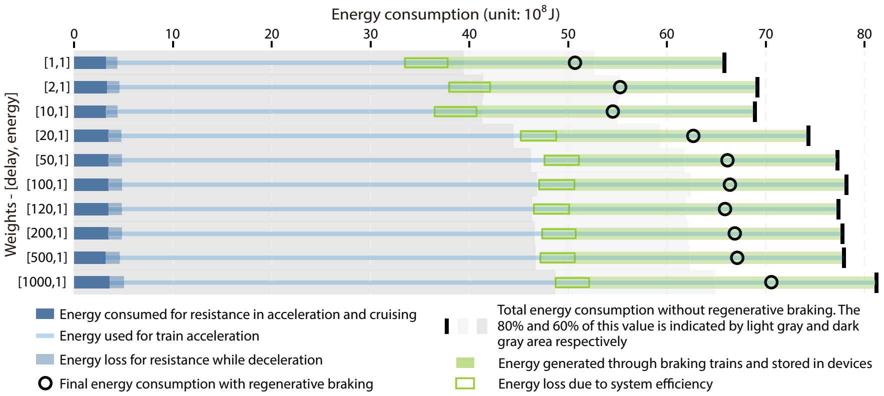


Figure 6.6: Composition of the energy consumption

efficiency (i.e., caused by the recuperation coefficient η) is represented by a bar with a green border.

As illustrated in Figure 6.6, the total energy consumption decreases with the increase of the importance of the energy consumption. The percentage of the energy that is re-utilized becomes larger when considering the energy consumption to be more important. Moreover, in our solutions, there is a large amount (around 40%-50%) of the train kinetic energy that is not stored, indicated by the difference of the lengths between the light blue line and the light green bar. One reason for this unstored energy is due to the configuration of the electric regions, i.e., the Den Bosch (Ht) station area is not considered as an electric region of regenerative braking, so that regenerative braking cannot be applied in this station area, as shown in Figure 6.1. Another reason is that the Den Bosch (Ht) station is the destination for most trains, so that many train braking actions happen in this area. As regenerative braking cannot be used in the Den Bosch (Ht) station, the train kinetic energy in these braking actions is all lost. In fact, the composition of the energy consumption strongly depends on the test case (e.g., electric regions and train routes) and the settings (e.g., the estimated system efficiency). For a certain test case, comprehensive experiments could be done to correct the input parameters (e.g., the recuperation coefficient η for system efficiency) for increasing solution accuracy, and also to find the best option for making maximum use of the regenerated energy (e.g., locations of installing energy storage devices). We evaluate the performance indicators with regards to regenerative braking, including the storage rate of the regenerated energy, the utilization rate of the stored energy, the percentage of the energy loss due to system efficiency, and the reduction of the total energy consumption due to regenerative braking, shown in Figure 6.7(a)-(d) respectively.

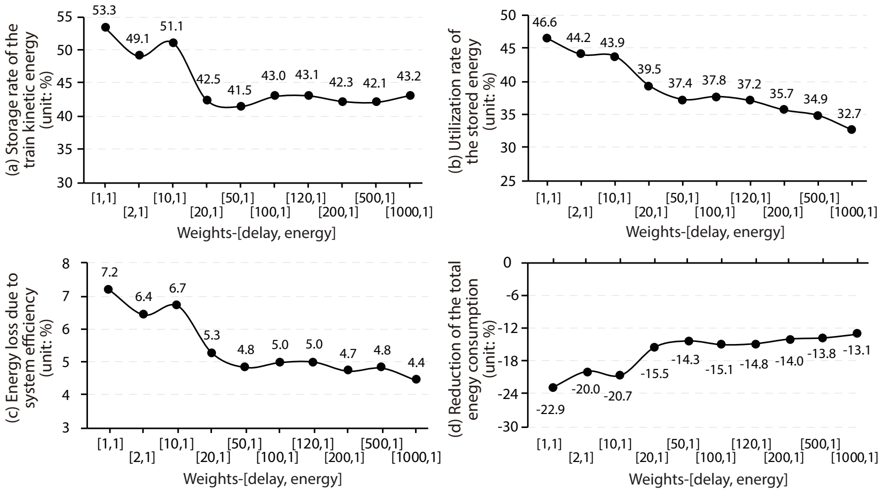


Figure 6.7: Storage rate of the train kinetic energy, utilization rate of the stored energy, energy loss due to system efficiency, and reduction of the total energy consumption

The results of the performance indicators differ for the different weights considered. Basically, an increase of the importance of the energy consumption leads to a larger storage rate of the train kinetic energy and a larger utilization rate of the regenerated energy; as a result, we obtain a larger reduction of the total energy consumption. The energy loss resulting from system efficiency also becomes larger with the increase of the storage rate and the utilization rate. In our case study, there is a surplus of the stored energy, i.e., not all of the stored energy are re-used for train acceleration; therefore, we can save more energy even if the energy loss increases. Overall, based on the Dutch test case, 41.5%-53.3% of the train kinetic energy is stored in energy storage devices. About 4.4%-7.2% of this stored energy is lost due to system inefficiency, and 32.7%-46.6% of this stored energy is re-utilized for train acceleration, which further leads to 13.1%-22.9% reduction of the total energy consumption. According to the experimental results, regenerative braking can significantly reduce the energy consumption of train operations, and it is an effective and practical way to achieve energy-efficient train operation. The results based on the Dutch test case show the effectiveness of the proposed formulations.

(4) Comparison with the train speed profiles generated by a train trajectory optimization approach

In this section, we assess the results of the $\mathrm{P_{TSPO}}$ problem from a train control perspective. We adopt the nonlinear train speed profile optimization approach proposed in Section 2 of Wang et al. (2013) and apply the sequential quadratic programming (SQP) approach to generate a speed profile for each train. The departure times at the origin and arrival times at the destinations of trains are given as time constraints for the train control problem. Moreover, the passing times at intermediate non-stopping stations and critical block sections are given as operational constraints, i.e., the traffic management problem is solved beforehand. For each train, a speed profile is generated with the objective of minimizing the energy consumption. For the train control problem, each block section in the network is divided into 20 subsections and the acceleration or deceleration of trains is assumed to be a constant for each subsection. Since the SQP approach could result in local minima, we use 10 initial points for the calculation of the train speed profile for each train, and we select the best solution among the resulting speed profiles. The SQP approach uses a more accurate and refined model for generating train speed profiles, compared with the $\mathrm{P_{TSPO}}$ approach, so it can be seen as a more accurate approach for obtaining optimal speed profiles. We compare the train speed profiles generated by the $\mathrm{P_{TSPO}}$ approach and the SQP approach, as well as the resulting energy consumption. Note that we consider the $\mathrm{P_{TSPO}}$ approach with the weighted-sum formulation. As we aim at assessing the quality of the speed profiles generated by the $\mathrm{P_{TSPO}}$ problem, we do not consider the option of regenerative braking here for both the $\mathrm{P_{TSPO}}$ approach and the SQP approach.

Figure 6.8 shows the speed-space trajectories obtained by the two approaches. For the sake of compactness, only one representative delay case with 15 trains is provided here. The train speed profiles of the P_{TSPO} solution and the SQP solution are indicated by solid lines and dashed lines respectively.

It can be seen that the speed profiles generated by the SQP approach are smoother than those of the P_{TSPO} approach in acceleration and deceleration modes, as the SQP approach uses the detailed nonlinear train model. However, for most trains that travel sequentially on nodes 27 → 30 → 28, the driving strategy obtained by the P_{TSPO} approach is more efficient for energy consumption. Due to the speed limit requirement for node 27, every train reduces its speed to 60 km/h while passing node 27. In the P_{TSPO} solution, most trains maintain a speed of 60 km/h after passing node 27 to further approach their destination, e.g., the 1B8001 and 1B16001 trains; a few trains accelerate to their maximum speeds after passing node 27, e.g., the 1D8001 train. In the SQP solution, every train that traverses node 27 accelerates after passing node 27 and then decelerates when approaching its destination. This may be caused by the different computational configurations of the two approaches and the sub-optimal solutions found by the two approaches. The train acceleration after passing node 27 needs to ad-

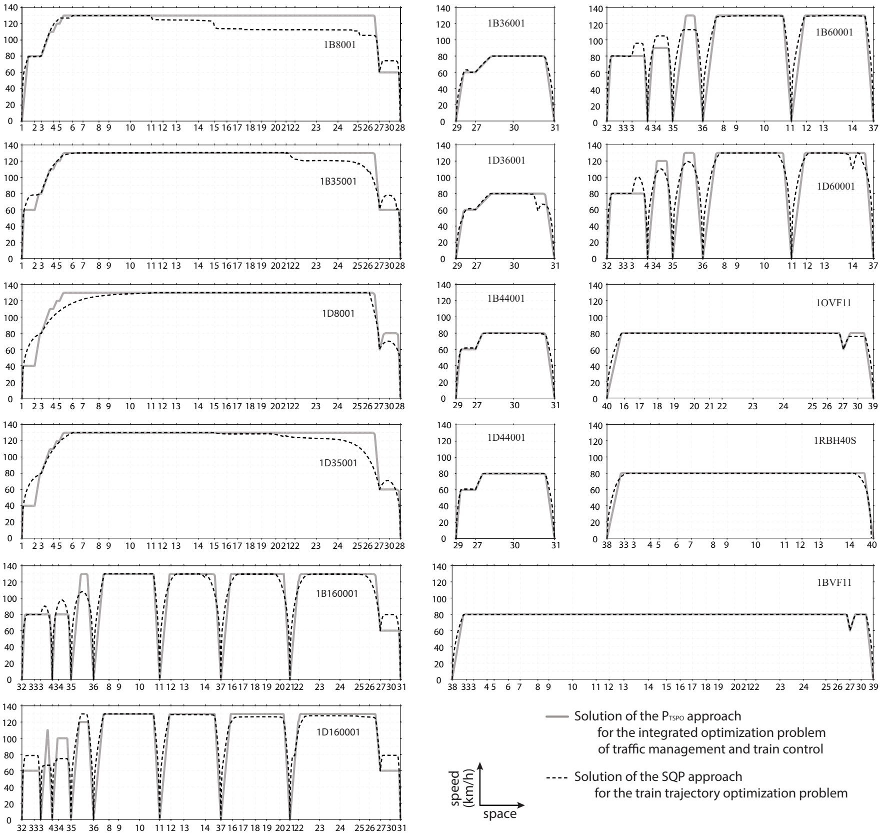


Figure 6.8: Comparison of the train speed profiles

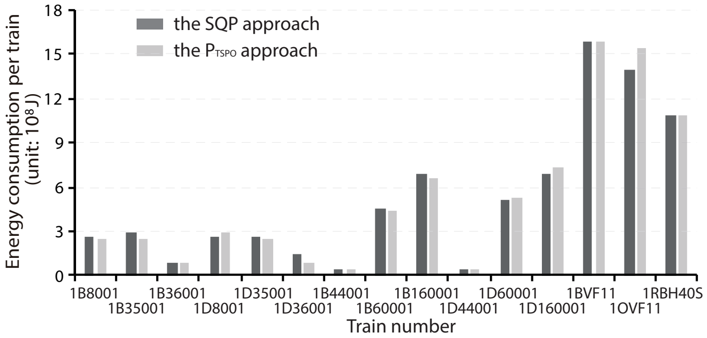


Figure 6.9: Comparison of the energy consumption for one representative delay case

ditionally use energy; therefore, the driving strategy of not accelerating the train after passing node 27, found by the P_{TSPO} approach, is more efficient in use of energy. For some trains, the speed profiles of the two approaches are highly coincident, e.g., the 1B36001 and 1BVF11 trains. The quality of the train speed profiles obtained by the P_{TSPO} approach is overall satisfactory, with regard to the solution found by the SQP approach.

In Figure 6.9, we present the energy consumption of each train, computed by the P_{TSPO} approach (indicated in gray) and the SQP approach (indicated in black). The Y-axis indicates the energy consumption for each train, and the X-axis represents the train number.

When focusing on each single train, the energy consumption computed by the two approaches is different. For some trains, e.g., the 1B8001 and 1B35001 trains, the speed profiles found by the P_{TSPO} approach are better in terms of efficient use of energy, mainly caused by the driving strategy of maintaining a speed of 60 km/h after passing node 27. For some other trains, e.g., the 1D160001 train, better speed profiles for energy efficiency are found by the SQP approach. However, the results of the total energy consumption of the two approaches are very close, i.e., 78.00×10^8(J) and 77.86×10^8(J) for the P_{TSPO} approach and the SQP approach respectively. The SQP approach overall performs better than the P_{TSPO} approach, as it finds better speed profiles in terms of efficient use of energy; however, the relative difference of the total energy consumption obtained by the two approaches is very small, only 0.18%.

According to the comparison of the P_{TSPO} approach and the SQP approach, it is demonstrated that the control performance of the P_{TSPO} approach is appropriate for both the quality of the speed profiles and the calculation of the energy consumption.

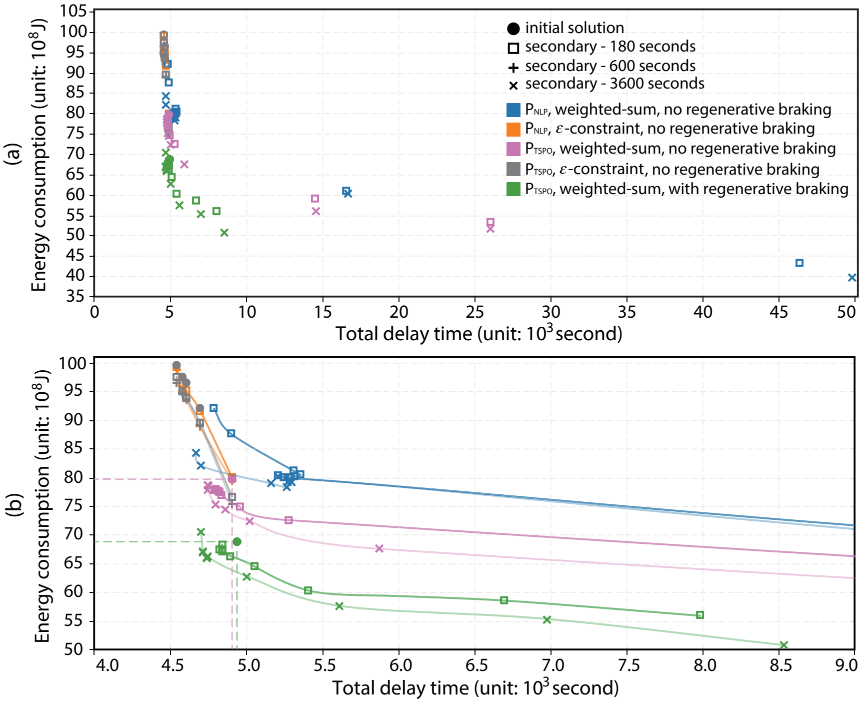


Figure 6.10: Overview of all experimental results, from the viewpoints of train delay and energy consumption

6.3.3 Discussion

We here summarize the main conclusions, sketched quantitatively in Figure 6.10 from the viewpoints of train delay (X-axis) and energy consumption (Y-axis), clarifying the trade-off between them. The experimental results reported in Section 6.3.2 are all included in Figure 6.10. Figure 6.10(b) is a zoom-in of Figure 6.10(a), with some trend lines. We use symbols to distinguish the computation time limits considered for obtaining the solutions, i.e., the dot, square, plus, and cross symbols represent the initial solution, and the secondary solutions obtained within 180, 600, and 3600 seconds of computation time respectively. Each symbol indicates an average result of 10 delay cases, obtained within the given computation time limit. We use colors to indicate the multiple combined choices of the optimization approaches (i.e., the P_{NLP} and P_{TSPO} approaches), the formulation methods (i.e., the weighted-sum formulation and the ε-constraint formulation), and the option of regenerative braking. The blue and orange colors indicate the P_{NLP} solutions by using the weighted-sum formulation and the ε-constraint formulation respectively. The purple and gray colors indicate the P_{TSPO} solutions with the weighted-sum formulation and the ε-constraint formulation respectively. The green color represents the solution considering regenerative braking, obtained by the P_{TSPO} approach with the weighted-sum formulation.

According to the experimental results presented in Section 6.3.2(1), the performance of the P_{TSPO} approach is better than the performance of the P_{NLP} approach. This is reflected in Figure 6.10 by the lower purple line, compared with the blue line in the case of using the weighted-sum formulation, and is also reflected by the gray line, which is mostly lower than the orange line when applying the ε-constraint formulation.

When using the ε-constraint formulation, the gray and orange trend lines in Figure 6.10 go towards a larger energy consumption with a stricter upper bound on the train delay. Required as the input of the ε-constraint formulation, the upper bound for the train delay has to be carefully chosen: an inappropriate upper bound may lead to a bad performance on either train delay or energy consumption, and it may even cause infeasibility of the optimization problem. For the weighted-sum formulation, the results of the train delay and the energy consumption cover a wide range, depending on the weights considered, see the blue, purple, and green trend lines. Overall, for our case, the weighted-sum formulation is more applicable than the ε-constraint formulation, as discussed in Section 6.3.2(2).

In Section 6.3.2(2.a), by using the P_{TSPO} approach with the weighted-sum formulation, we find better solutions. The delay time and the energy consumption are reduced at the same time, compared with the initial solution that is obtained by considering a fixed full speed profile for each train on each block section. This is reflected by the symbols located between the purple dashed lines in Figure 6.10; the vertical and horizontal dashed lines come from the initial solution (indicated by the purple dot symbol), given as benchmarks. Train delay and energy consumption can be improved simultaneously through managing the train speed, by up to 4.0% and 5.6% respectively (see the detailed experimental results provided in the online repository (Research Collection ETH Zurich). The simultaneous reduction of the two objectives also demonstrates the benefit of integrating traffic management and train control and shows great potential for energy efficiency of train operations.

When regenerative braking is applied, the total energy consumption for train operations is significantly reduced, indicated by the green lines in Figure 6.10, which represent a smaller energy consumption in comparison with the purple lines. Applying regenerative braking is thus an effective way to achieve energy-efficient train operation (as discussed in Section 6.3.2(3)).

The good quality of the train speed profiles generated by the P_{TSPO} approach is demonstrated in Section 6.3.2(4), compared with the train speed profiles obtained by the SQP approach, which is a more accurate approach for computing optimal speed profiles. The relative difference of the total energy consumption of the P_{TSPO} solution and the SQP solution is very small (only 0.18%), which also demonstrates the good control performance of the P_{TSPO} approach.

6.4 Conclusions

In this chapter, we have considered extensions towards energy-efficient train operation, based on the integrated optimization approaches proposed in Chapter 5, where the traffic-related properties (i.e., departure times, arrival times, and train orders) and the train-related properties (i.e., train speed trajectory) are optimized simultaneously. We have first introduced energy evaluation into the integrated optimization approaches, calculating the energy used for train acceleration and the energy consumed for overcoming resistance. We have developed a set of linear constraints for the $\mathrm{P}_{\mathrm{TSPO}}$ problem to compute the energy consumption. An approximation of the resistance function with a piecewise constant function has been applied for computing the energy consumption of the $\mathrm{P}_{\mathrm{NLP}}$ and $\mathrm{P}_{\mathrm{PWA}}$ problems. In addition, we have considered the option of regenerative braking and presented linear formulations to calculate the utilization of the energy obtained through regenerative braking. With the inclusion of the energy-related formulations, we could focus on two objectives, i.e., delay recovery and energy efficiency, by using the weighted-sum formulation and the ε-constraint formulation. Experiments have been conducted based on a real-world dataset adapted from the Dutch railway network (the same as Chapter 5). According to the experimental results, the $\mathrm{P}_{\mathrm{TSPO}}$ approach overall performs better than the $\mathrm{P}_{\mathrm{NLP}}$ approach. Aiming at both delay recovery and energy efficiency, the two objectives can be improved at once (e.g., by up to 4.0% and 5.6% for the train delay and the energy consumption in one of the solutions) through managing the train speed. Moreover, for the test case, the application of regenerative braking leads to about 13.1%-22.9% reduction of the total energy consumption. By comparing with the train speed profiles obtained by the SQP approach, which is a more accurate approach for computing speed profiles, the good control performance of the $\mathrm{P}_{\mathrm{TSPO}}$ approach has been demonstrated, as the speed profiles of the $\mathrm{P}_{\mathrm{TSPO}}$ approach are similar to those obtained by the SQP approach.

In future research, comprehensive experiments could be done to correct the input parameters (e.g., the recuperation coefficient η for system efficiency) for increasing solution accuracy, and also to find the best option for making maximum use of the regenerated energy (e.g., locations of installing energy storage devices).

Chapter 7

Distributed optimization of real-time railway traffic management for large-scale networks

This chapter introduces distributed optimization approaches, with the aim of improving the computational efficiency of the integrated optimization problem proposed in Chapters 5 and 6 for large-scale railway networks.

This chapter is organized as follows. Section 7.1 first gives a detailed introduction of the distributed optimization problem of real-time railway traffic management. Section 7.3 introduces three decomposition methods, namely a geography-based, a train-based, and a time-interval-based decomposition, where a number of subproblems are obtained. In Section 7.4, three distributed optimization approaches are developed for handling the couplings among the resulting subproblems. Section 7.6 examines the performance of the proposed algorithms and decomposition methods, through experiments on the Dutch railway network. Finally, the conclusions are given in Section 7.7.

7.1 Introduction

Real-time traffic management is of great importance to limit the negative consequences caused by perturbations occurring in real-time railway operations. The train control problem reflects the traffic control process by defining speed profiles to let the delayed trains reach the stations at the times specified by the traffic management problem. Due to the real-time nature, a solution is required in a very short computation time for dealing with delayed and canceled train services and for evacuating delayed and stranded passengers as quickly as possible.

The real-time traffic management problem has been studied extensively in the literature, and we refer to the literature review in Section 2.2.1. There are many optimization approaches available for the railway traffic management problem, using different formulation methods, e.g., the alternative-graph-based method by D'Ariano et al. (2007a) and the cumulative-flow-variable-based method by Meng and Zhou (2014), and having different focuses, e.g., considering multiple classes of running traffic in Corman et al. (2011a) and integrating train control in Luan et al. (2018a,b) and Chapters 5 and 6. These approaches often lead to large and rather complex optimization problems, especially when considering microscopic details or when integrating traffic management with other problems (e.g., the train control problem). They mostly have excellent performance on small-scale cases, where optimality can be achieved in a short computation time. However, when enlarging the scale of the case, the computation time for finding a solution or for proving the optimality of a solution increases exponentially in general.

Distributed optimization approaches have gained a lot of attention to face the need for fast and efficient solutions for problems arising in the context of large-scale networks, such as utility maximization problems. We refer to Nedic and Ozdaglar (2010) and Meinel et al. (2014) for more details. The main idea is to solve the problems either serially or in parallel to jointly minimize a separable objective function, usually subject to coupling constraints that force the different problems to exchange information during the optimization process. In the literature, these approaches have been widely studied in many fields. In transportation systems, they have been explored for controlling road traffic (Findler and Stapp, 1992), for managing air traffic (Wangermann and Stengel, 1996), and for railway traffic (Kersbergen et al., 2016). Kersbergen et al. (2016) focused on the railway traffic management problem with macroscopic details and considered a geography-based decomposition. Lamorgese et al. (2016) proposed a Benders-like decomposition within a master/slave scheme to address the train dispatching problem. The master and the slave problems here respectively correspond to a macroscopic and microscopic representation of the railway.

Bad computational efficiency is one limitation that (integrated) optimization approaches have for large-scale networks. Overcoming this limitation will promote the application of such optimization approaches in practice. Thus, we aim at improving the computational efficiency of solving such (integrated) optimization problems by using distributed optimization approaches. The optimization problem that we focus on is the mixed-integer linear programming (MILP) problem ($\mathrm{P}_{\mathrm{TSPO}}$, which overall yields a better performance), developed in Chapters 5 and 6, where the traffic-related variables (i.e., a set of times, orders, and routes to be followed by trains) and the train-related variables (i.e., speed trajectories) are optimized simultaneously.

In this chapter, we consider three decomposition methods, namely a geography-based (GEO) decomposition, a train-based (TRA) decomposition, and a time-interval-based (TIN) decomposition. The GEO decomposition consists of first partitioning the whole railway network into many elementary block sections and then clustering these block sections into a given number of regions. An integer linear optimization approach is proposed to cluster the block sections with the objective of minimizing the total number of train service interconnections among the regions and of balancing the region sizes. Consequently, several subproblems are obtained, and each region corresponds to one subproblem. For the TRA decomposition, we decompose an F-train problem into F subproblems, where each subproblem includes one individual train only. The TIN decomposition makes a division of the time horizon into equal-length pieces, and each time interval piece corresponds to one subproblem, which involves all events (i.e., train departures and arrivals) that are estimated to happen in this time interval. No matter which decomposition method is used, couplings always exist among subproblems, and the presence of these couplings leads to a non-separable structure of the whole optimization problem. To handle the issue of the couplings, we introduce three distributed optimization approaches. The first one is an Alternating Direction Method of Multipliers (ADMM) algorithm, where each subproblem is solved through coordination with the other subproblems in an iterative manner. The second one is a priority-rule-based (PR) algorithm, where the subproblems are sequentially and iteratively solved in a priority order with respect to the solutions of the other subproblems that have been solved with a higher priority. The third one is a Cooperative Distributed Robust Safe But Knowledgeable (CDRSBK) algorithm, where four types of couplings are defined and each subproblem is iteratively solved together with its actively coupling subproblems.

Experiments are conducted based on the Dutch railway network to comparatively test the performance of the three proposed algorithms with the three decomposition methods, in terms of feasibility, computational efficiency, solution quality, and estimated optimality.

7.2 Standard mixed-integer linear programming formulation of the $\mathrm{P}_{\mathrm{TSPO}}$ problem

Recall that an MILP approach ($\mathrm{P}_{\mathrm{TSPO}}$) has been developed in Chapters 5 and 6 for addressing the integrated problem of real-time traffic management and train control. The $\mathrm{P}_{\mathrm{TSPO}}$ approach can be expressed by a standard MILP formulation as follows:

$$\min_{\lambda} \quad \mathcal{Z}(\lambda) = c^{\top} \cdot \lambda \tag{7.1a}$$

$$\text{s.t.} \quad A \cdot \lambda \leq b \tag{7.1b}$$

with variable $\lambda \in \mathbb{R}^n$, matrix $A \in \mathbb{R}^{m \times n}$, and vectors $c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$. The objective function $\mathcal{Z}(\lambda)$ in (7.1a) minimizes the weighted sum of the total train delay times at all visited stations and the energy consumption of the train movements. The vector λ contains both the traffic-related variables and train-related variables for describing the train movements on block sections, in particular, the arrival times a, departure times d, train orders θ, incoming speeds v^{in}, cruising speeds v^{cru}, outgoing speeds v^{out}, approach time τ^{approach}, and clear time τ^{clear}. In (7.1b), all constraints (inequalities and equalities) are represented for ensuring the train speed limitations, for enforcing the consistency of train transition times and speeds, for guaranteeing the required dwell times, for determining train blocking times, and for respecting the block section capacities. The MILP problem (7.1a)-(7.1b) can be solved by a standard MILP solver, e.g., CPLEX or Gurobi. Interested readers are referred to Chapters 5 and 6 for a more detailed description of the P_{TSPO} problem.

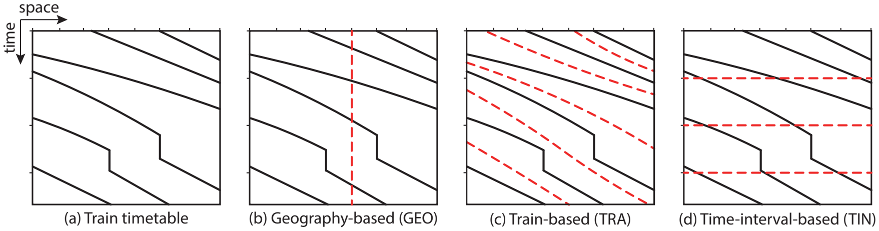
7.3 Problem decomposition

Three decomposition methods, i.e., the geography-based (GEO), the train-based (TRA), and the time-interval-based (TIN) decomposition, are described in Sections 7.3.1 to 7.3.3 respectively. Section 7.3.4 discusses the decomposition result, i.e., subproblems and couplings. Figure 7.1 comparatively illustrates the three decomposition methods in a time-space graph, where black lines indicate train paths and red dashed lines indicate boundaries of subproblems.

7.3.1 Geography-based decomposition

The GEO decomposition partitions the whole railway network into a given number of regions. Consider a railway network composed of a set of block sections E, and consider a set of scheduled trains F traversing this network. We could easily partition the whole network into $|E|$ units, by means of a geography-(i.e., block section)-based decomposition; however, this could result in a large number of subproblems with couplings. In general, a larger number of subproblems implies more couplings among them, which makes coordination difficult and which may affect the overall performance of the system; therefore, we cluster these elementary block sections into a pre-defined number $|R|$ of regions, where $R = \{1, 2, ..., |R|\}$ is the set of regions. Figure 7.1(b) illustrates a 2-region example of the geography-based decomposition; as shown, the timetable is split in the dimension of space.

To distribute $|E|$ different units into $|R|$ groups, there are $|R|^{|E|}$ ways, e.g., up to 10^6 ways for distributing 20 units into 2 groups only. Thus, in our case, a huge number of the GEO decomposition results are available. To obtain the optimal decomposition



(a) Train timetable (b) Geography-based (GEO) (c) Train-based (TRA) (d) Time-interval-based (TIN)

Figure 7.1: Illustration of the three decomposition methods in a time-space graph

result, an integer linear programming (ILP) approach is proposed. We now introduce this ILP approach.

The set E_f contains the sequence of block sections composing the route of train f, and $|E_f|$ represents the number of block sections along the route of train f. The binary vector β_f indicates whether two consecutive block sections along the route of train f belong to different regions, e.g., if $(\beta_f)_j = 1$, then the j^{th} and $(j+1)^{\text{th}}$ block sections in set E_f belong to different regions, otherwise, $(\beta_f)_j = 0$. The binary vector α_r indicates the assignment of all block sections for region r, e.g., if $(\alpha_r)_i = 1$, then the i^{th} block section in set E is assigned to region r, otherwise, $(\alpha_r)_i = 0$. The route matrix $\mathbf{B}_f \in \mathbb{Z}^{(|E_f|-1)\times|E|}$ indicates that train f traverses a sequence of block sections, e.g., if train f traverses from the 1st block section to the 3rd block section in the set E, then $\mathbf{B}_f = \begin{bmatrix} 1 & 0 & -1 & 0 & ... \end{bmatrix}$. Each row of $\mathbf{B}_f$ indicates the transition of a train from one block section to another. The integer vector $\mu \in (\mathbb{Z}^+)^{|E|\times 1}$ indicates the index of regions that each block section $e \in E$ belongs to. We use $\|\cdot\|_1$ to denote the 1-norm. The objective function is formulated as follows:

$$\min_{\alpha,\beta} \left[\zeta \cdot \left(\sum\nolimits_{f\in F} \|\beta_f\|_1\right) + (1-\zeta)\cdot\left(\sum\nolimits_{r=1}^{|R|} \left| \|\alpha_r\|_1 - \frac{|E|}{|R|}\right|\right)\right], \tag{7.2}$$

where the weight $\zeta \in [0,1]$ is used to balance the importance of the two objectives. The first term serves to minimize the train service interconnections among regions, and the second term aims at balancing the region sizes.

We consider four constraints, presented as follows:

$$\frac{\left|\left(\mathbf{B}_f \cdot \mu\right)_j\right|}{|R|-1} \le \left(\beta_f\right)_j, \quad \forall f \in F, j \in \left\{1, ..., |E_f|-1\right\}, \tag{7.3}$$

guarantees that $(\beta_f)_j > 0$ if the two consecutive block sections along the route of train f belong to different regions, i.e., $\left|\left(\mathbf{B}_f \cdot \mu\right)_j\right| > 0$.

$$\mu_i \in \{1, ..., |R|\}, \quad \forall i \in \{1, ..., |E|\}, \tag{7.4}$$

enforces that the indices of the resulting regions cannot exceed the pre-defined number of regions, while

$$(\alpha_r)_i \le 1 - \frac{|\mu_i - r|}{|R|-1}, \quad \forall r \in \{1, ..., |R|\}, i \in \{1, ..., |E|\}, \tag{7.5}$$

and

$$\|\alpha_r\|_1 \ge 1, \quad \forall r \in \{1, ..., |R|\}, \tag{7.6}$$

are used to avoid solution in which no block section is assigned to some region(s). Specifically, in (7.5), if the i^{th} block section in set E is assigned to region r, i.e., $\mu_i = r$, then the binary variable $(\alpha_r)_i = 1$; otherwise, $(\alpha_r)_i = 0$. In (7.6), we ensure that at least one block section is assigned to each region. As a result, (7.5) and (7.6) imply that the number of the resulting regions must equal the given number $|R|$. An illustrative example is provided in Section 7.5 to explain the above formulations.

With a pre-defined number of regions, there are two impact factors of the GEO decomposition result: the network layout and the train routes planned in the original timetable. This implies that the optimal decomposition result is the same for all delay cases. The train routes have impact on the decomposition result, because we minimize the train service interconnections among regions in the objective function (7.2).

When applying the GEO decomposition, some trains may traverse from one region to another region. The time and speed that a train leaves one region should equal the time and speed that the train arrives at the other region. Therefore, the time and speed transition constraints are the complicating constraints for the GEO decomposition, which cause the couplings among regions (i.e., subproblems). The speed and time transition constraints of the MILP problem (7.1) are formulated in (5.2) and (5.9) of Section 5.4.1 respectively.

7.3.2 Train-based decomposition

The TRA decomposition simply splits an $|F|$-train problem into $|F|$ subproblems, and each subproblem corresponds to a 1-train problem, as illustrated in Figure 7.1(c). Thus, for a given instance, only one decomposition result is available. The only impact factor of the TRA decomposition is the involved trains. Such a train-based decomposition was used by Brännlund et al. (1998) for addressing train timetabling problem by using Lagrangian relaxation.

When applying the TRA decomposition, each train is independently scheduled in each subproblem, so that trains may use the same infrastructure at the same time, resulting in conflicts. Therefore, the capacity constraint is the complicating constraint for the TRA decomposition. The capacity constraint is formulated in (5.23)-(5.24) of Section 5.4.1.

7.3.3 Time-interval-based decomposition

The time-interval-based (TIN) decomposition makes a division of a train timetable in the dimension of time, based on a given size of time interval, as illustrated in Figure 7.1. The TIN decomposition is implemented with consideration of disruptions (delays), i.e., taking the impact of disruptions on the train schedule into account while making the decomposition. We independently schedule all trains by taking disruptions into account, generating an infeasible timetable, where train conflicts exist. With this infeasible timetable, we estimate the times at which all events (e.g., train departure and arrival) may occur. Each event is then assigned to one time interval based on its estimated occurrence time. As a result, the subproblem of each time interval includes all events that are estimated to occur in this time interval. The TIN decomposition result mainly depends on the given size of time interval and the estimated train schedule, which can be different in delay cases.

One train service consists of a set of events indicating the departures and arrivals of the train on block sections. When applying the TIN decomposition, these events may be split into more than one time intervals. Thus, similar to the GEO decomposition (where trains may traverse from region to region), the time and speed when a train leaves a time interval should be consistent with those when the train enters the next time interval, i.e., the speed and time transition constraints are complicating constraints, as formulated in (5.2) and (5.9) of Section 5.4.1. Moreover, as the TIN decomposition is based on an estimated infeasible timetable, an event assigned to time interval t maybe further scheduled into the next time interval $t+1$, causing conflicts with the events in time interval $t+1$. Therefore, the capacity constraint in (5.23)-(5.24) of Section 5.4.1 is also a complicating constraint for the TIN decomposition.

7.3.4 Subproblems and couplings

Let us denote with S the set of the $|S|$ resulting subproblems, e.g., $|S| = |R|$ for the GEO decomposition. No matter which decomposition method is used, we can always divide the constraints of the MILP problem (7.1) into two categories, i.e., local constraints and complicating constraints. A local constraint is only related to a single subproblem, so that it leads to a separable structure of the optimization problem. A complicating constraint is associated with at least two subproblems, so that it results in a non-separable structure. We thus rewrite (7.1b) into a general form of the following local and complicating constraints:

$$A^{\text{loc}} \cdot \lambda \leq b^{\text{loc}} \tag{7.7a}$$

$$A^{\text{cpl}} \cdot \lambda \leq b^{\text{cpl}} \tag{7.7b}$$

with matrices $A^{\text{loc}} \in \mathbb{R}^{m_1 \times n}$ and $A^{\text{cpl}} \in \mathbb{R}^{m_2 \times n}$ and vectors $b^{\text{loc}} \in \mathbb{R}^{m_1}$ and $b^{\text{cpl}} \in \mathbb{R}^{m_2}$. The complicating constraint (7.7b) contains the speed transition constraint (5.2) and the time transition constraint (5.9) when the GEO decomposition applies; (7.7b) contains the capacity constraints (5.23)-(5.24) when the TRA decomposition is adopted; and (7.7b) contains all the above constraints (5.2), (5.9), (5.23), and (5.24) when the TIN decomposition is considered.

Let us denote with $Q_p = \{q_1, q_2, ..., q_{m_p}\}$ the set of m_p subproblems that have couplings with subproblem p. The subproblem $p \in S$ of the MILP problem (7.1) is formulated as

$$\min_{\lambda_p} \quad Z_p(\lambda_p) = c_p^\top \cdot \lambda_p \tag{7.8a}$$

$$\text{s.t.} \quad A_p^{\text{loc}} \cdot \lambda_p \leq b_p^{\text{loc}} \tag{7.8b}$$

$$A_{p,q}^{\text{cpl}} \cdot \lambda_p + A_{q,p}^{\text{cpl}} \cdot \lambda_q \leq b_{p,q}^{\text{cpl}}, \forall q \in Q_p \tag{7.8c}$$

where $A_{p,q}^{\text{cpl}}$ and $A_{q,p}^{\text{cpl}}$ are selection matrices for selecting the coupling variables between subproblems p and q. Since each coupling constraint in (7.8c) includes the variables λ_p and λ_q of two subproblems p and q, we cannot explicitly add them to any individual subproblem. Instead we can determine and exchange values of the coupling variables among subproblems in an iterative way. The train(s) of one subproblem p can obtain an agreement through iterations that inform the train(s) of the coupling subproblems $q \in Q_p$ about what subproblem p prefers the values of coupling variables to be. To achieve this agreement, for a single subproblem p, we have to compute the optimal coupling variables (inputs) for its coupling subproblems $q \in Q_p$ as well, rather than only focusing on computing optimal local variables. Moreover, for its coupling subproblems $q \in Q_p$, we need to compute both the optimal local variables and coupling variables (outputs). Through exchanging these desired coupling variables, the values of these outputs and inputs should converge to each other, and a set of local inputs that is overall optimal should be found. Distributed optimization approaches for reaching this agreement are developed in Section 7.4.

7.4 Distributed optimization approaches

This section introduces three distributed optimization approaches to address the issue of couplings among subproblems, namely the Alternating Direction Method of Multipliers (ADMM) algorithm, the priority-rule-based (PR) algorithm, and the Cooperative Distributed Robust Safe But Knowledgeable (CDRSBK) algorithm, presented in Sections 7.4.1 to 7.4.3 respectively. A key challenge in distributed optimization algorithms is to ensure that the solution generated for a single subproblem leads to feasible solutions that satisfy the complicating constraints with other subproblems.

7.4.1 Alternating direction method of multipliers algorithm

The alternating direction method of multipliers (ADMM) algorithm (see e.g., Boyd et al., 2011) solves problems of the following form:

$$\min_{x,z} \quad f(x)+g(z) \tag{7.9a}$$

$$\text{s.t.} \quad A\cdot x+B\cdot z=b, \tag{7.9b}$$

with variables $x\in\mathbb{R}^n$ and $z\in\mathbb{R}^m$, matrices $A\in\mathbb{R}^{p\times n}$ and $B\in\mathbb{R}^{p\times m}$, and vector $b\in\mathbb{R}^p$. Assume that the variables x and z can be split into two parts, with the objective function that is separable across this splitting. We can then form the augmented Lagrangian relaxation as

$$L_\rho(x,z,y)=f(x)+g(z)+y^\top(A\cdot x+B\cdot z-b)+\tfrac{\rho}{2}\cdot\|A\cdot x+B\cdot z-b\|_2^2, \tag{7.10}$$

where y is the dual variable (Lagrangian multiplier), the parameter $\rho>0$ indicates the penalty multiplier, and $\|\cdot\|_2$ denotes the Euclidean norm. The augmented Lagrangian function is optimized by minimizing over x and z sequentially and then evaluating the resulting equality constraint residual. By applying the dual ascent method, the ADMM algorithm consists of the following iterations:

$$x^{i+1}:=\arg\min_x L_\rho(x,z^i,y^i), \tag{7.11a}$$

$$z^{i+1}:=\arg\min_z L_\rho(x^{i+1},z,y^i), \tag{7.11b}$$

$$y^{i+1}:=y^i+\rho(A\cdot x^{i+1}+B\cdot z^{i+1}-b) \tag{7.11c}$$

where i is the iteration counter. In the ADMM algorithm, the variables x and z are updated in a sequential fashion, which accounts for the term alternating direction.

The ADMM algorithm can obviously deal with linear equality constraints, but it can also handle linear inequality constraints. The latter can be reduced to linear equality constraints by replacing constraints of the form $A \cdot x \leq b$ by $A \cdot x + s = b$, adding the slack variable s to the set of optimization variables, and setting $\mathcal{Z}(x,s) = 0$, if $s \geq 0$, otherwise, setting $\mathcal{Z}(x,s) = \infty$. Alternatively, we can also work with an equivalent reformulation of problem (7.8), where we replace the complicating constraint (7.8c) by

$$\mathcal{C}_p(\lambda_p, \lambda_q) = 0 \tag{7.12}$$

where $\mathcal{C}_p(\lambda_p, \lambda_q) = \max\left\{0, A^{\mathrm{cpl}}_{p,q} \cdot \lambda_p + A^{\mathrm{cpl}}_{q,p} \cdot \lambda_q - b^{\mathrm{cpl}}_{p,q}\right\}$ with component-wise maximum. In such a way, we can transform the inequality constraints into equality constraints.

Now we can apply the ADMM algorithm, and the augmented Lagrangian formulation of the MILP problem (7.1) can be described as follows:

$$L_\rho = \sum_{p \in S}\left[\mathcal{Z}_p(\lambda_p) + \sum_{q \in Q_p}\left[y^\top_{p,q} \cdot \mathcal{C}_p(\lambda_p, \lambda_q) + \frac{\rho}{2} \cdot ||\mathcal{C}_p(\lambda_p, \lambda_q)||^2_2\right]\right] \tag{7.13}$$

subject to (7.8b).

The iterations to compute the solution of the MILP problem (7.1) based on the augmented Lagrangian formulation (7.13) include quadratic terms; therefore, the function cannot directly be distributed over subproblems. Inspired by Negenborn et al. (2008), for handling this non-separable issue, the function (7.13) can be approximated by solving $|S|$ separate problems of the form

$$\min_{\lambda_p} \quad \mathcal{Z}_p(\lambda_p) + \sum_{q \in Q_p} \mathcal{J}_p(\lambda_p, \lambda_q, y_{p,q}) \tag{7.14}$$

subject to (7.8b) for the train movements of single subproblem p, where the additional term $\mathcal{J}_p(\cdot)$ deals with coupling variables.

We now define the term $\mathcal{J}_p(\cdot)$ by using a serial implementation. We apply a block coordinate descent approach (Beltran Royoa and Heredia, 2002; Negenborn et al., 2008). The approach minimizes the quadratic term directly in a serial manner. One subproblem after another minimizes its local and coupling variables while the variables of the other subproblems stay fixed. At iteration i, let us use $\widehat{Q^i_p} \subseteq Q_p$ to denote the set of those coupling subproblems (of subproblem p) that have been solved before solving subproblem p.

The serial implementation uses the information from both the current iteration i and the last iteration $i-1$. With the information $\bar{\lambda}_q = \lambda^{(i)}_q$ computed in the current iteration i for subproblems $q \in \widehat{Q^i_p}$ and the information $\bar{\lambda}_q = \lambda^{(i-1)}_q$ obtained in the last iteration $i-1$ for the other subproblems $q \in Q_p \backslash \widehat{Q^i_p}$, we can solve (7.14) for subproblem p by using the following function:

$$\mathcal{J}_p(\lambda_p, \bar{\lambda}, y_{p,q}) = y^\top_{p,q} \cdot \mathcal{C}_p(\lambda_p, \bar{\lambda}_q) + \frac{\rho}{2} \cdot ||\mathcal{C}_p(\lambda_p, \bar{\lambda}_q)||^2_2 \tag{7.15}$$

The second term of (7.15) penalizes the deviation from the coupling variable iterates that were computed for the subproblems before subproblem p in the current iteration i and by the other subproblems during the last iteration $i-1$.

The solution procedure of the ADMM algorithm is described in Algorithm 7.1.

Algorithm 7.1 The solution procedure of the ADMM Algorithm

Input: The penalty multiplier ρ, the maximum number of iterations $I^{\max}$, the expected gap ε, the decomposition results (subproblem set S), and those inputs identical to the $\mathrm{P_{TSPO}}$ problem.

Initialization: Set the Lagrange multipliers $y^{(0)} := 0$ and set all elements in the latest solution set $\mathcal{S}_{\text{sol}} := \{\bar{\lambda}_p | p \in S\}$ to be empty.

```
for iteration $i := 1, 2, ..., I^{\max}$ do
    Randomly generate the orders of subproblems, denoted as $P^{(i)}_{\text{order}}$.
    for subproblem $j := 1, 2, ..., |S|$ do
        Solve subproblem $p := P^{(i)}_{\text{order}}(j)$, consisting of objective function (7.14) and constraint (7.8b), by taking the available solutions in $\mathcal{S}_{\text{sol}}$ for all $q \in \widehat{Q^i_p}$ into account.
        Denote the obtained solution of subproblem $p$ as $\lambda^{(i)}_p$, and update the latest solution set $\mathcal{S}_{\text{sol}}$ by adding or replacing $\bar{\lambda}_p$, where $\bar{\lambda}_p := \lambda^{(i)}_p$.
    end for
    Update the Lagrange multipliers by $y^{(i)}_{p,q} := y^{(i-1)}_{p,q} + \rho \cdot \mathcal{C}_p(\lambda^{(i-1)}_p, \lambda^{(i-1)}_q)$ for all $p \in S$ and $q \in Q_p$.
    Break the iterations if the difference of the coupling variables at the current iteration step $i$ is less than the expected gap $\varepsilon$, i.e., $\|\mathcal{C}\|_\infty \leq \varepsilon$, where $\varepsilon$ is a small positive scalar and $\|\cdot\|_\infty$ denotes the infinity norm.
end for
```

By applying the ADMM algorithm, we solve the subproblems $p \in S$ in an iterative manner, with respect to the local constraint (7.8b) of a single subproblem p and taking the solutions of all coupling subproblems (i.e., the variable $\bar{\lambda}_q$ for $q \in Q_p$ obtained in either the current iteration or the last iteration) into account. In (7.13), only the local objective $\mathcal{Z}_p$ for a single subproblem p is minimized, not the global objective $\sum_{p \in S} \mathcal{Z}_p$ for all subproblems.

In order to further improve the performance of the ADMM algorithm, we can consider a cost-to-go function $\mathcal{Z}^{\text{ctg}}_p(\lambda_p)$ into the objective function of each subproblem, which provides an estimation of the train running to its destination. The cost-to-go function is inspired by Kuwata and How (2011), where a cost-to-go function is used to represent the remainder of the path to the target for addressing an unmanned aerial vehicles trajectory optimization problem. Then, the objective function (7.14) for subproblem $p \in S$ can be rewritten as follows:

$$\min_{\lambda_p} \mathcal{Z}_p(\lambda_p) + \mathcal{Z}^{\text{ctg}}_p(\lambda_p) + \sum_{q \in Q_p} \mathcal{J}_p(\lambda_p, \lambda_q, y_{p,q}) \tag{7.16}$$

For instance, with the GEO decomposition, we can define the cost-to-go function as the deviation between the actual and planned departure time from the block section where a train leaves a region. Thus, an original timetable with more details is then needed, where the departure and arrival times are given not only for stations but also for block sections.

7.4.2 Priority-rule-based algorithm

The ADMM algorithm incorporates the complicating constraint (7.8c) into the objective function and strives to make the information consistent among subproblems (i.e., each subproblem takes the information of the other subproblems into account) in an iterative manner. However, convergence cannot be guaranteed for non-convex optimization problems, so that a feasible solution may not be available. Therefore, we need to explore other distributed optimization approaches. We next introduce a priority-rule-based (PR) algorithm.

The main idea of the PR algorithm is to optimize train schedules of the subproblems in a sequential manner according to problem priorities, with respect to the solutions of the other subproblems that have already been solved in the current iteration. The problem priorities are determined by the train delay times of the subproblems, e.g., we solve the subproblem with the largest delay time first. Note that the result could be different even with the same problem priorities, as multiple optimal solutions may exist for each subproblem. These different optimal solutions with the same objective value for one subproblem could then result in different objective values for the other subproblems.

By applying the PR algorithm, the complicating constraint (7.8c) for the subproblem $p \in S$ can be rewritten as follows:

$$A^{\mathrm{cpl}}_{p,q} \cdot \lambda_p + A^{\mathrm{cpl}}_{q,p} \cdot \bar{\lambda}_q \leq b^{\mathrm{cpl}}_{p,q}, \forall q \in Q_p \tag{7.17}$$

with the solution $\bar{\lambda}_q = \lambda_q^{(i)}$ computed in the current iteration i for all subproblems $q \in \widehat{Q^i_p}$.

The solution procedure of the PR algorithm is described in Algorithm 7.2.

In the priority-rule-based algorithm, we solve each subproblem $p \in S$ in a sequential manner according to the priorities of the subproblems, with respect to the local constraint (7.8b) and the outputs $\bar{\lambda}_q$ of the coupling subproblems $q \in Q_p$ in (7.17). Similar to the ADMM algorithm, only the local objective Z_p is minimized when solving subproblem p, rather than the global objective $\sum_{p \in R} Z_p$ for all subproblems. Constraint (7.17) ensures that the coupling variables of subproblem p satisfy those of its coupling subproblems $q \in Q_p$ obtained in the current iteration. For the first solved subproblem in each iteration, the complicating constraint (7.17) is relaxed.

Algorithm 7.2 The solution procedure of the PR Algorithm

```
Input: The maximum number of iterations $I^{\max}$, the iteration number $\kappa$, the decomposition results (subproblem set $S$), and those inputs identical to the $P_{\text{TSPO}}$ problem.
Initialization: Set the local upper bound $o_{\text{UB}}^{(0)} := M$, and the global upper bound $O_{\text{UB}}^{(0)} := M$, where $M$ is a sufficient large positive number. Initialize the problem priorities $P_{\text{prior}}^{(0)}$ arbitrarily.
for iteration $i := 1,2,...,I^{\max}$ do
    Sort subproblems in set $S$ in a descending order by their problem priorities $P_{\text{prior}}^{(i-1)}$, denoted as $P_{\text{order}}^{(i)}$.
    Set the solution set $\mathcal{S}_{\text{sol}} := \{\bar{\lambda}_p | p \in S\}$ to be empty.
    for subproblem $j := 1,2,...,|S|$ do
        Solve subproblem $p := P_{\text{order}}^{(i)}(j)$, including objective function (7.8a) and constraints (7.8b) and (7.17), with respect to the available solutions in $\mathcal{S}_{\text{sol}}$ for all $q \in \widehat{Q_p^i}$.
        Denote the obtained solution of subproblem $p$ as $\lambda_p^{(i)}$, and update the solution set $\mathcal{S}_{\text{sol}}$ by adding $\bar{\lambda}_p$, where $\bar{\lambda}_p := \lambda_p^{(i)}$.
    end for
    Compute the local upper bound $o_{\text{UB}}^{(i)}$, and update the global upper bound by
        $O_{\text{UB}}^{(i)} := \begin{cases} o_{\text{UB}}^{(i)}, & \text{if } O_{\text{UB}}^{(i-1)} > o_{\text{UB}}^{(i)} \\ O_{\text{UB}}^{(i-1)}, & \text{otherwise} \end{cases}$
    Update the problem priorities $P_{\text{prior}}^{(i)}$ by the train delay times of the subproblems.
    Break the iterations if the global upper bounds are not improved for a given number of iterations $\kappa$, i.e., $O_{\text{UB}}^{(i)} = O_{\text{UB}}^{(i-\kappa)}$.
end for
```

7.4.3 Cooperative Distributed Robust Safe But Knowledgeable algorithm

The third algorithm considered in this research is the Cooperative Distributed Robust Safe But Knowledgeable (CDRSBK) algorithm, introduced by Kuwata and How (2011) to address trajectory planning problems. In the CDRSBK algorithm, four types of couplings among subproblems are defined for a subproblem $p \in S$, as illustrated in Figure 7.2.

Type_1 indicates a non-active coupling between subproblem $p \in S$ and its neighbor; Type_2 indicates an active coupling between subproblem p and its neighbor; Type_3 indicates the coupling between the active coupling neighbors of subproblem p and their neighbors; and Type_4 indicates the coupling between two active coupling neighbors of subproblem p. Let us use Q_p to denote the set of all coupling neighbors of subproblem p and use Q_p^{act} to denote the set of subproblem p's neighbors that have an active coupling with subproblem p. The interpretation of active and non-active couplings can be different for different decomposition methods. We discuss the details regarding

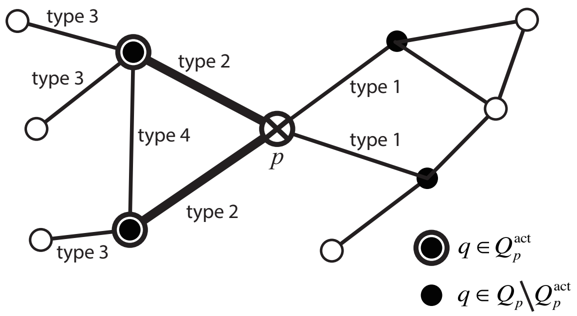


Figure 7.2: Four types of couplings defined in the CDRSBK algorithm

their implementations in Section 7.4.4.

By applying the CDRSBK algorithm, the subproblem $p \in S$ of the MILP problem (7.8a)-(7.8c) can be reformulated as

$$\min_{\lambda_p, \xi_q} \; Z_p(\lambda_p) + \sum_{q \in Q_p^{\text{act}}} Z_q(\bar{\lambda}_q + T_q \cdot \xi_q) \tag{7.18a}$$

$$\text{s.t.} \;\; A_p \cdot \lambda_p \leq b_p^{\text{loc}} \tag{7.18b}$$

$$A_q \cdot (\bar{\lambda}_q + T_q \cdot \xi_q) \leq b_q^{\text{loc}}, \forall q \in Q_p^{\text{act}} \tag{7.18c}$$

$$A_{p,q}^{\text{cpl}} \cdot \lambda_p + A_{q,p}^{\text{cpl}} \cdot \bar{\lambda}_q \leq b_{p,q}^{\text{cpl}}, \forall q \in Q_p \backslash Q_p^{\text{act}} \tag{7.18d}$$

$$A_{p,q}^{\text{cpl}} \cdot \lambda_p + A_{q,p}^{\text{cpl}} \cdot (\bar{\lambda}_q + T_q \cdot \xi_q) \leq b_{p,q}^{\text{cpl}}, \forall q \in Q_p^{\text{act}} \tag{7.18e}$$

$$A_{o,q}^{\text{cpl}} \cdot \bar{\lambda}_o + A_{q,o}^{\text{cpl}} \cdot (\bar{\lambda}_q + T_q \cdot \xi_q) \leq b_{o,q}^{\text{cpl}}, \forall o \in Q_q \backslash Q_p^{\text{act}}, q \in Q_p^{\text{act}} \tag{7.18f}$$

$$\begin{aligned} A_{q_1,q_2}^{\text{cpl}} \cdot (\bar{\lambda}_{q_1} + T_{q_1} \cdot \xi_{q_1}) + A_{q_2,q_1}^{\text{cpl}} \cdot (\bar{\lambda}_{q_2} + T_{q_2} \cdot \xi_{q_2}) \leq b_{q_1,q_2}^{\text{cpl}}, \\ \forall q_1, q_2 \in Q_p^{\text{act}}, q_2 \in Q_{q_1}, q_1 \in Q_{q_2} \end{aligned} \tag{7.18g}$$

In (7.18a), the objective function of both subproblem p and its actively coupled subproblems $q \in Q_p^{\text{act}}$ are included. Constraints (7.18b)-(7.18c) represent the local constraints of subproblem p and its actively coupled subproblems $q \in Q_p^{\text{act}}$ respectively. In (7.18d)-(7.18g), coupling constraints (7.8c) are rewritten for the four types of couplings among subproblems respectively. When solving subproblem p, besides the local variable λ_p, the variable ξ_q is also optimized for its actively coupled subproblems $q \in Q_p^{\text{act}}$ on the communicated solution $\bar{\lambda}_q$, as follows:

$$\lambda_q = \bar{\lambda}_q + T_q \cdot \xi_q \tag{7.19}$$

parameterized with a matrix T_q, which is formed to allow the variable ξ_q to change only the rows that correspond to the active complicating constraints. This can be also interpreted as allowing a change for the constraint that has a non-zero Lagrange multiplier. In (7.18a), the objectives of a single subproblem p and its actively coupled neighbors $q \in Q_p^{\text{act}}$ are both minimized.

The solution procedure of the CDRSBK algorithm is described in Algorithm 7.3.

In each iteration, the CDRSBK algorithm actually solves each subproblem, with additional objectives and coupling constraints that include the changeable (local) variables

Algorithm 7.3 The solution procedure of the CDRSBK Algorithm

Input: The maximum number of iterations $I^{\max}$, the iteration number κ, the decomposition results (subproblem set S), and those inputs identical to the $\mathrm{P_{TSPO}}$ problem.

Initialization: Set the local upper bound $o_{\mathrm{UB}}^{(1)} := M$, and the global upper bound $O_{\mathrm{UB}}^{(1)} := M$, and all elements in the latest solution set $\mathcal{S}_{\mathrm{sol}} := \{\bar{\lambda}_p | p \in S\}$ to be empty.

```
for iteration $i := 1, 2, ..., I^{\max}$ do
    Randomly generate the orders of subproblems, denoted as $P_{\mathrm{order}}^{(i)}$.
    for subproblem $j := 1, 2, ..., |S|$ do
        Solve subproblem $p := P_{\mathrm{order}}^{(i)}(j)$ and its actively coupling subproblems $q \in Q_p^{\mathrm{act}}$, consisting of objective function (7.18a) and constraints (7.18b)-(7.18g), by taking the available solutions in set $\mathcal{S}_{\mathrm{sol}}$ for all $o \in (Q_p \backslash Q_p^{\mathrm{act}}) \cup (Q_q \backslash Q_p^{\mathrm{act}})$ into account.
        Denote the obtained solutions of subproblem $p$ and its actively coupling subproblems $q \in Q_p^{\mathrm{act}}$ as $\lambda_p^{(i)}$ and $\lambda_q^{(i)}$ (which is obtained by (7.19)) respectively, and update the latest solution set $\mathcal{S}_{\mathrm{sol}}$ by adding or replacing $\bar{\lambda}_p$ and $\bar{\lambda}_q$ for all $q \in Q_p^{\mathrm{act}}$, where $\bar{\lambda}_p := \lambda_p^{(i)}$ and $\bar{\lambda}_q := \lambda_q^{(i)}$.
    end for
    Compute the local upper bound $o_{\mathrm{UB}}^{(i)}$, and update the global upper bound by
        $O_{\mathrm{UB}}^{(i)} := \begin{cases} o_{\mathrm{UB}}^{(i)}, & \text{if} \, O_{\mathrm{UB}}^{(i-1)} > o_{\mathrm{UB}}^{(i)} \\ O_{\mathrm{UB}}^{(i-1)}, & \text{otherwise} \end{cases}$
    Break the iterations if the global upper bounds are not improved for a given number of iterations $\kappa$, i.e., $O_{\mathrm{UB}}^{(i)} = O_{\mathrm{UB}}^{(i-\kappa)}$.
end for
```

of its actively coupled subproblems $q \in Q_p^{\mathrm{act}}$. If the variables of its actively coupled subproblems are unchangeable, i.e., $\lambda_q = \bar{\lambda}_q$ when ξ_q has no impact on the variables, the coupling constraints are automatically satisfied and could be omitted.

7.4.4 Remarks on the implementation of the decomposition methods and algorithms

Here we give some remarks for the implementation of the proposed decomposition methods and algorithms, e.g., interpreting the active and non-active couplings in the CDRSBK algorithm for different decomposition methods and giving some tips for achieving feasibility.

Remark 7.1: Train orders in the ADMM algorithm with the GEO decomposition and the TIN decomposition

It is essential to ensure that train orders in subproblems are feasible, in order to avoid unnecessary iterations and to achieve fast convergence. To do this, we keep the consistency of the train orders that are interrelated, e.g., if two trains cannot overtake on a sequence of block sections, then the train orders of these two trains on these block sections are interrelated and must be same.

Remark 7.2: The CDRSBK algorithm & the GEO decomposition
If two regions are connected by tracks, i.e., they are neighbors, then we consider that a coupling exists between the two subproblems of these two regions. A coupling between two subproblems is considered to be active (Type_2) if there is any train traverse between the two regions of the two subproblems; otherwise, the coupling is recognized as non-active coupling (Type_2). For coupling Type_3 and Type_4, we follow their general definitions, i.e., the couplings between an active coupling neighbor and its coupling neighbors are labeled as Type_3 coupling and the coupling between two active coupling neighbor is labeled as Type_4.

Remark 7.3: The CDRSBK algorithm & the TRA decomposition
If two trains use the same infrastructure (block section), then we consider that a coupling exists between the two subproblems of these two trains. If a conflict exists between these two trains, then their coupling is recognized as an active coupling; otherwise, their coupling is considered to be non-active. For coupling Type_3 and Type_4, we follow their general definitions. In the TRA decomposition, we often have many trains that use the same infrastructure; but conflicts may never happen among some of them, e.g., a train scheduled in the early morning has little chance to conflict with another train scheduled in the late afternoon. Thus, to further reduce the problem complexity for large-scale networks, we provide two more options for defining coupling Type_1 and Type_3. We denote the option described above as Opt_1. The difference between Opt_1 and Opt_2 is in the definition of coupling Type_3: in Opt_2, we label the couplings between an active coupling neighbor and its *active* coupling neighbor as Type_3. Based on Opt_2, we discard all Type_1 couplings, which results in Opt_3, i.e., when and only when a conflict happens between two trains, a coupling exists between them and is recognized as active coupling (Type_2). However, we still have Type_3 and Type_4 couplings in Opt_3 by following their general definitions. According to the case study, Opt_3 performs best, in terms of computational efficiency and solution quality. An illustrative example is provided in Section 7.5 to graphically explain these three options.

Remark 7.4: The CDRSBK algorithm & the TIN decomposition
Due to the nature of the TIN decomposition, the relation among subproblems is relatively simple in this case. Couplings exist only between two consecutive subproblems (i.e., two subproblems of two consecutive time intervals t and $t+1$) and are all recognized as active couplings (Type_2). As a result, according to the general definition of the four types of couplings, the couplings between a consecutive subproblem and its consecutive subproblem are considered as Type_3 (e.g., for subproblem t, a Type_3 coupling exists between subproblems $t+1$ and $t+2$), and Type_1 and Type_4 couplings do not exist. Moreover, for guaranteeing a feasible solution in the first iteration, solving subproblems in a time sequence (i.e., for time intervals $t = 1, 2, 3....$ in sequence) is recommended.

7.5 An illustrative example

In this section, we use a small instance to explain the proposed decomposition methods and algorithms. As illustrated in Figure 7.3, the instance includes 4 trains following the pre-defined routes, i.e., train $f_1 : e_1 \rightarrow e_2 \rightarrow e_4$, train f_2 and $f_3 : e_1 \rightarrow e_3 \rightarrow e_5$, and train $f_4 : e_3 \rightarrow e_5$.

We now illustratively explain the formulation of the ILP problem proposed in Section 7.3.1. We can write the set of block sections as $E = \{e_1, e_2, e_3, e_4, e_5\}$. The route matrix B_{f_1} and the variable vector β_{f_1} for train f_1 and the variable vector μ for block sections can be expressed as

$$B_{f_1} = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \end{bmatrix}, \beta_{f_1} = \begin{bmatrix} (\beta_{f_1})_1 \\ (\beta_{f_1})_2 \end{bmatrix}, \text{and } \mu = \begin{bmatrix} \mu_1 & \mu_2 & \mu_3 & \mu_4 & \mu_5 \end{bmatrix}^\top.$$

Consider the consecutive block sections e_1 and e_2 in the route of train f_1; then (7.3) results in the inequality $\frac{|\mu_1 - \mu_2|}{|R|-1} \leq (\beta_{f_1})_1$. If the two block sections belong to the same region, i.e., $\mu_1 = \mu_2$, then we will have $(\beta_{f_1})_1 = 0$ (as we are solving a minimization problem). If block sections e_1 and e_2 belong to different regions, i.e., $\mu_1 \neq \mu_2$, then we will have $(\beta_{f_1})_1 = 1$, as the left-hand side of the inequality is strictly in the range $[0, 1)$ and B_{f_1} is an integer matrix. Constraints (7.5)-(7.6) are used to avoid the solutions like $\mu = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix}^\top$.

We now illustrate the three decomposition methods. Let us assume $|R| = 5$, i.e., we have 5 regions and each region contains only one block section. Let us denote T as the number of subproblems for the TIN decomposition. By applying the three proposed decomposition methods, the resulting subproblems and (primary) couplings can be determined as shown in Figure 7.4. As illustrated, the GEO decomposition results in 5 subproblems, corresponding to the 5 block sections respectively; the TRA decomposition leads to 4 subproblems, corresponding to the 4 trains respectively; and the TIN decomposition gives T subproblems connected in time sequence.

We now illustrate the three options for defining the four types of couplings in the CDRSBK algorithm with the TRA decomposition. Let us assume the infeasible timetable shown in Figure 7.5(a), which can be generated by independently scheduling trains one-by-one without considering their couplings. The three options are illustrated in Figures 7.5(b) to 7.5(d) respectively. Let us now focus on train f_1 (i.e., subproblem f_1). In Opt_1, the coupling between f_1 and f_2 is recognized as an active coupling (Type_2),

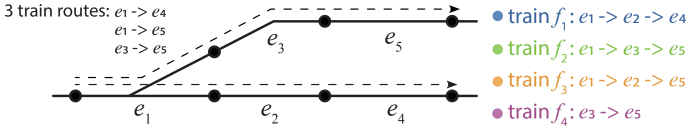

3 train routes: e1 -> e4
e1 -> e5
e3 -> e5
e3
e5
e1
e2
e4
● train f1: e1 -> e2 -> e4
● train f2: e1 -> e3 -> e5
● train f3: e1 -> e2 -> e5
● train f4: e3 -> e5

Figure 7.3: A small instance for illustrating the proposed decomposition methods and algorithms

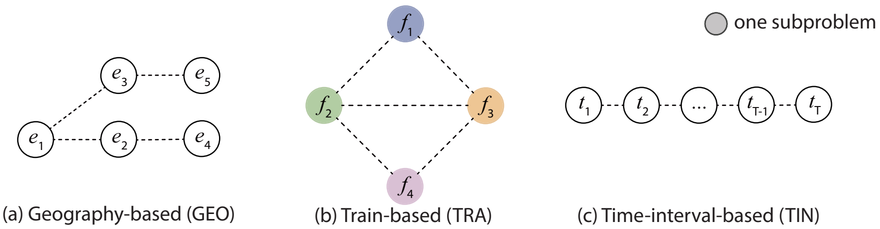
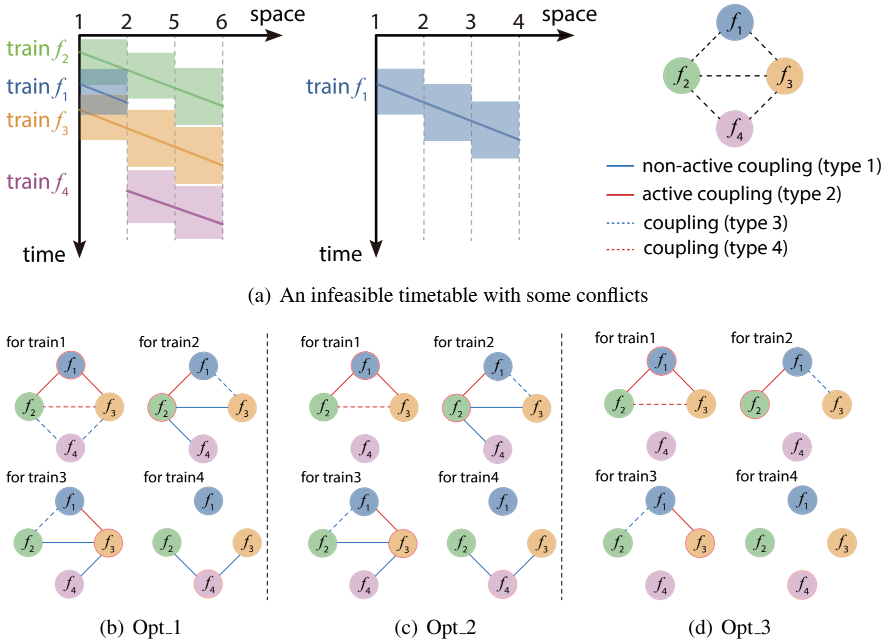


Figure 7.4: Subproblems and couplings

because train f_1 has a conflict with train f_2 in the timetable shown in Figure 7.5(a). Both f_2 and f_3 have an active coupling (Type_2) with f_1; so a Type_3 coupling exists between f_2 and f_3. Train f_1 and train f_4 use completely different block sections. So subproblem (train) f_4 only has couplings with f_2 and f_3, and their couplings are recognized as a Type 3 coupling for subproblem f_1. Train f_2 uses same block sections as all the other trains, but only has a conflict with train f_1; therefore, when we focus on train f_2, the coupling between f_2 and f_1 is considered to be Type_2 and the coupling between f_2 and f_3 (and f_4) is recognized as Type_1. In Opt_2, still focusing on subproblem f_1, as the coupling between f_2 and f_4 is a non-active coupling (Type_1, when focusing on subproblem f_2 or f_4), we consider the Type 3 coupling between f_2 and f_4 (and between f_3 and f_4) do not exist. In Opt_3, we consider no coupling if there is no



(a) An infeasible timetable with some conflicts

(b) Opt_1

(c) Opt_2

(d) Opt_3

Figure 7.5: Three options of the CDRSBK algorithm with the TRA decomposition

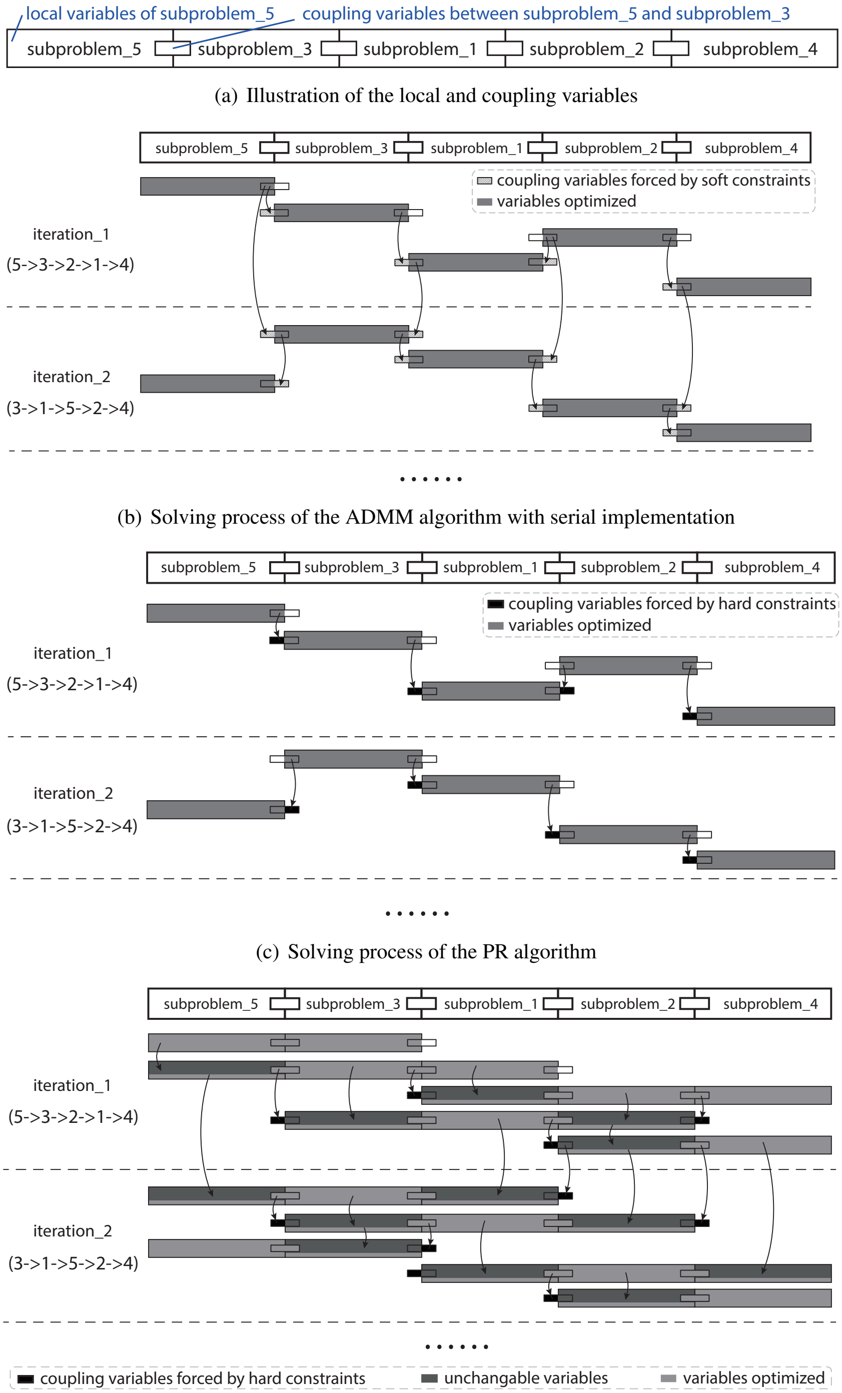
conflict, which can be simply explained as removing all Type_1 couplings based on the coupling architecture of Opt_2; however, Type_3 and Type_4 couplings are still defined as same as Opt_1 (and Opt_2).

We now explain the three proposed algorithms by considering the GEO decomposition as an example. Figure 7.4(a), which shows 5 subproblems and their couplings of the GEO decomposition, is re-drawn as Figure 7.6(a) to express the local and coupling variables. Figure 7.6(a) is used to graphically explain the solving process of the three proposed algorithms.

Figure 7.6(b) illustrates the solving process of the ADMM algorithm based on the small instance in Figure 7.3. For serial implementation, we randomly generate orders of subproblems in each iteration and solve each subproblem according to the orders through coordination with other neighboring subproblems. As shown, we first solve subproblem_5, then the obtained solution of subproblem_5 is given as a soft constraint to solve subproblem_3. Subproblem_1 is solved based on the solutions of both subproblem_2 and subproblem_3, using soft constraints as well. In each iteration, we always consider the latest solution obtained, e.g., when solving subproblem_1 in iteration_2, we use the solution of subproblem_2 obtained in last iteration, as subproblem_2 has not been solved in iteration_2, and we use solution of subproblem_3 obtained in iteration_2, as it has been solved in current iteration_2.

The solving process of the PR algorithm can be illustrated in Figure 7.6(c) for the small instance. As shown, subproblem_5 is first solved at iteration_1. After solving subproblem_5, the solution of subproblem_5 is given as a hard constraint, indicated by a black block, for solving subproblem_3. As we only considered (respect to) the solutions obtained at current iteration, there is no interaction between iterations.

Figure 7.6(d) illustrates the solving process of the CDRSBK algorithm. In this case, all couplings between two neighboring subproblems are considered to be active (Type_2). As shown, in iteration_1, with consideration of subproblem_5, we first solve subproblem_5 and subproblem_3. Then with considering subproblem_3, we solve subproblem_5, subproblem_3, and subproblem_1, but only part of variables in subproblem_5 can be changed. Dark gray indicates unchangeable variables, coming from the lasted solution obtained for the corresponding subproblem, and light gray indicates changeable variables. When addressing subproblem_2, some variables in subproblem_1 are unchangeable, including the coupling variables related to subproblem_3. Therefore, the coupling between subproblem_1 and subproblem_3 will also be satisfied as a hard constraint when solving subproblem_2.


local variables of subproblem_5
coupling variables between subproblem_5 and subproblem_3
subproblem_5
subproblem_3
subproblem_1
subproblem_2
subproblem_4
coupling variables forced by soft constraints
variables optimized
iteration_1
(5->3->2->1->4)
iteration_2
(3->1->5->2->4)
coupling variables forced by hard constraints
variables optimized
unchangable variables
......

(a) Illustration of the local and coupling variables

(b) Solving process of the ADMM algorithm with serial implementation

(c) Solving process of the PR algorithm

(d) Solving process of the CDRSBK algorithm

Figure 7.6: Illustration of the three algorithms, considering the GEO decomposition

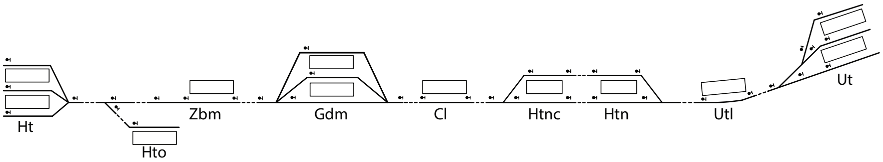


Figure 7.7: A railway network

7.6 Case study

7.6.1 Setup

We consider a line of the Dutch railway network, connecting Utrecht (Ut) to Den Bosch (Ht), of about 50 km length, with 9 stations, as shown in Figure 7.7. The network comprises 42 nodes and 40 cells. We consider one hour of heterogeneous traffic with 15 trains. Moreover, we consider different numbers of regions for the GEO decomposition, ranging from 2 to 6, and we consider 4 time intervals for the TIN decomposition, i.e., 300s, 600s, 900s, and 1200s. We consider 15 delay cases with randomly generated primary delays following a 3-parameter Weibull distribution, as explained in Corman et al. (2011b). We consider the average result of 15 delay cases with randomly generated primary delays. The maximum number of iterations is set to 200, 100, and 30 for the ADMM, PR, and CDRSBK algorithm respectively. A larger number is set for the ADMM algorithm because in general it needs more iterations to converge, and a smaller number is set for the CDRSBK algorithm because it often finds a feasible solution very fast and its solution is updated multiple times in one iteration.

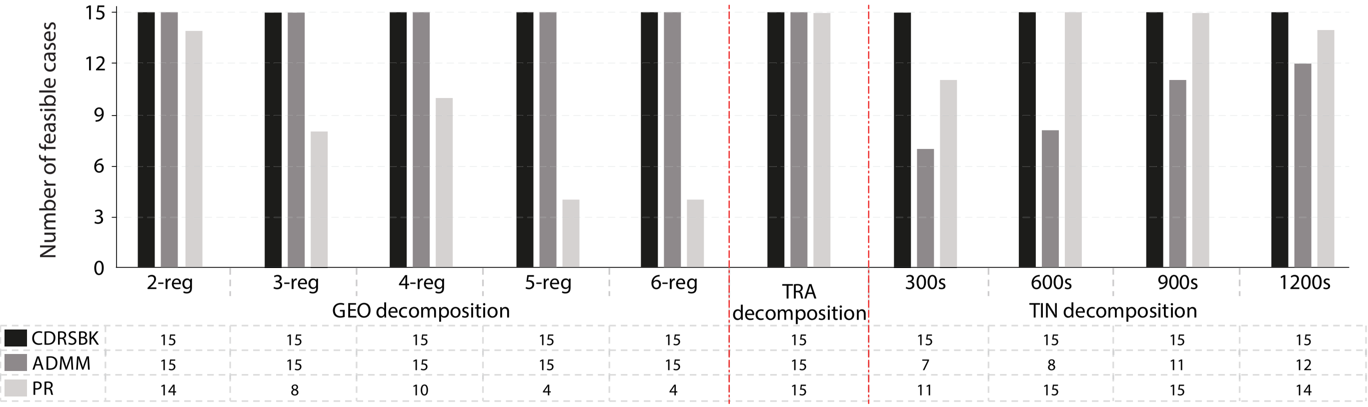
We adopt the CPLEX solver version 12.6.3 implemented in the MATLAB (R2018a) TOMLAB toolbox to solve the MILP problems. The experiments are performed on a computer with an Intel® Core$^{\text{TM}}$ i7 @ 2.00 GHz processor and 16GB RAM.

7.6.2 Experimental results and discussion

This section shows the (average) results of 15 delay cases from the viewpoints of feasibility, estimated optimality, solution quality, and computational efficiency.

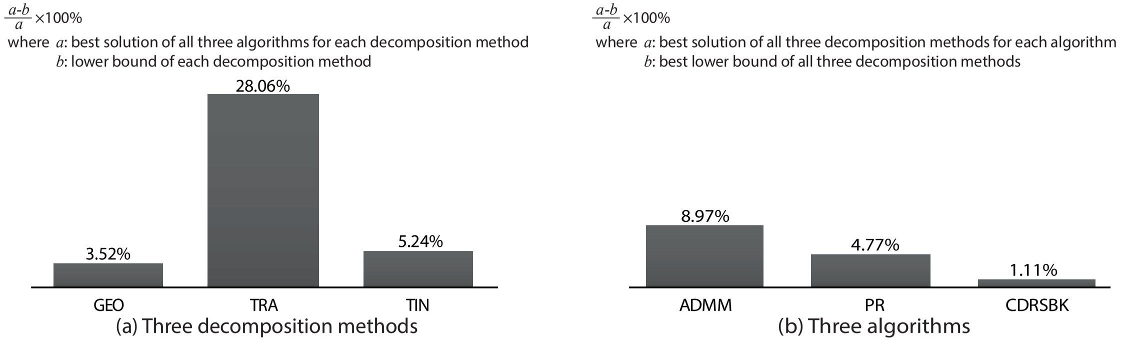
Figure 7.8 presents the number of cases that we can find feasible solutions within the maximum number of iterations. We can conclude that, for achieving feasibility, the TRA decomposition performs best among the three decomposition methods, and the CDRSBK algorithm is the best among the three algorithms. Considering a larger number of regions for the GEO decomposition or considering a smaller time interval for the TIN decomposition can make feasibility difficult to achieve, as they lead to a larger number of couplings among subproblems.

In Figure 7.9, the estimated optimality gap for each decomposition method and each algorithm is given, calculated by $\frac{a-b}{a} \times 100\%$, where a represents the best solution of



	2-reg	3-reg	4-reg	5-reg	6-reg	TRA decomposition	300s	600s	900s	1200s
■ CDRSBK	15	15	15	15	15	15	15	15	15	15
■ ADMM	15	15	15	15	15	15	7	8	11	12
□ PR	14	8	10	4	4	15	11	15	15	14

Figure 7.8: Feasibility of the three decomposition methods and three algorithms


$\frac{a-b}{a} \times 100\%$

where a: best solution of all three algorithms for each decomposition method
b: lower bound of each decomposition method

(a) Three decomposition methods

$\frac{a-b}{a} \times 100\%$

where a: best solution of all three decomposition methods for each algorithm
b: best lower bound of all three decomposition methods

(b) Three algorithms

Figure 7.9: Estimated optimality of the three decomposition methods and three algorithms

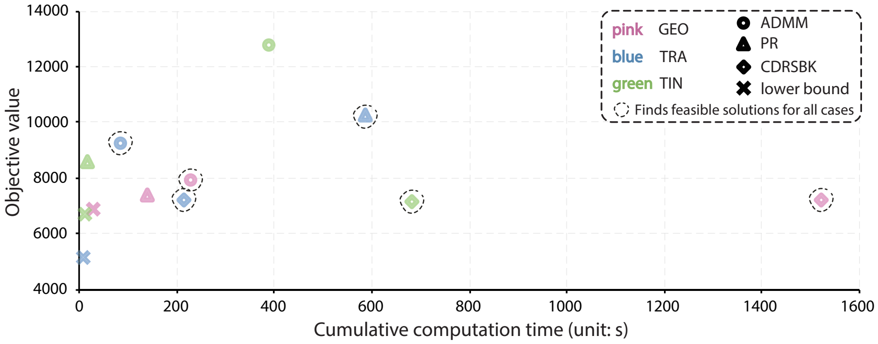


Figure 7.10: Solution quality and computational efficiency

the three algorithms for each decomposition method and b indicates the lower bound obtained by each decomposition, when we focus on the three decomposition methods; and if we focus on the three algorithms, then a represents the best solution of the three decomposition method for each algorithm and b indicates the best lower bound obtained by the three decomposition methods. As shown, the estimated optimality gap of the GEO decomposition is 3.52%, the lowest among the three decomposition methods, and the CDRSBK algorithm has the smallest estimated optimality gap (only 1.11%) among the three algorithms. A large estimated optimality gap does not always reflect a bad solution quality; it may be caused by a loose lower bound, as in the case of the TRA decomposition.

Figure 7.10 shows the cumulative computation time (on the X-axis) and the objective value (on the Y-axis). The cumulative computation time in a serial implementation for solving the subproblems is the sum of the CPU time consumed for finding the best feasible solution. Dashed circles around symbols indicate that feasible solution(s) can be found for all 15 delay cases by using the corresponding decomposition method and algorithm. When focusing on the three decomposition methods (represented by colors), the GEO decomposition (in pink) leads to a large range in computation time and a small range in objective value. This implies that the GEO decomposition results in small differences in the solution quality, but the computational efficiency is quite different for different algorithms. For the TRA decomposition (in blue) and the TIN decomposition (in green), wide ranges still exist in the two dimensions, and they show a general trade-off between solution quality and computational efficiency. Let us now focus on the three algorithms (indicated by symbols). The CDRSBK algorithm (indicated by diamonds) overall yields the best solution quality, and the computational efficiency becomes much better when the TRA decomposition is applied. The performance of the ADMM and PR algorithms is highly variable. For the ADMM algorithm (indicated by circles), the best solution quality is achieved when using the GEO decomposition, and the best computational efficiency is achieved when the TRA decomposition is adopted. The PR algorithm (indicated by triangles) has the best performance with respect to solution quality when the GEO decomposition is used and with respect to computational efficiency when the TIN decomposition is applied. A black dashed circle around a symbol indicates that feasible solution(s) can be found for all 15 delay cases by using the corresponding decomposition method and algorithm. Moreover, the lower bound of the TRA decomposition (indicated by a blue cross symbol) is the loosest, which leads to its large estimated optimality gap in Figure 7.9.

Overall, the CDRSBK algorithm with the TRA decomposition, the ADMM algorithm with the GEO decomposition, and the ADMM algorithm with the TRA decomposition yield a good overall performance. All these three combinations can find feasible solutions for all delay cases. In comparison, the first two combinations yield the best performance with respect to solution quality and a satisfactory performance with respect to computational efficiency. The last combination shows the best computational efficiency (roughly half of the computation time compared with the first two combinations) but at the cost of a relatively bad solution quality.

Moreover, when using the CDRSBK algorithm together with the TRA decomposition, Opt_3 described in Section 7.4.4 yields the best performance with respect to both solution quality and computational efficiency. For Opt_1, Opt_2, and Opt_3, the average objective value for the 15 delay cases is 7934.43, 7334.86, and 7217.08 respectively, and the average cumulative computation time is 255.19 seconds, 224.64 seconds, and 104.75 seconds.

Based on the above findings, to combine the best features of all combinations, a promising approach is to first use the ADMM algorithm with the TRA decomposition to generate a good feasible solution as quickly as possible and then next try the CDRSBK algorithm with the TRA decomposition or the ADMM algorithm with the GEO decomposition to get a potentially better solution at the cost of more CPU time.

7.7 Conclusions

We have introduced distributed optimization approaches, aiming at improving the computational efficiency of the integrated optimization problem for large-scale railway networks. Three decomposition methods have been presented to split the whole optimization problem into several subproblems, and three distributed optimization approaches have been proposed for dealing with the couplings among subproblems.

The performance of the proposed approaches has been examined in terms of feasibility, estimated optimality, solution quality, and computational efficiency. The TRA decomposition and the CDRSBK algorithm yield the best performance from the perspective of feasibility. The GEO decomposition and the CDRSBK algorithm yield the smallest estimated optimality gap. The CDRSBK algorithm with the TRA decomposition and the ADMM algorithm with the GEO decomposition achieve the best performance on solution quality and satisfactory performance on computational efficiency. The ADMM algorithm with the TRA decomposition shows the best computational efficiency but gives a relatively bad solution.

Future research will focus on the practical applications of the distributed optimization approaches. A promising two-step procedure can be used: first generate a feasible solution in short time (e.g., by applying the ADMM algorithm) and then improve the solution quality (by using the CDRSBK algorithm) based on that feasible solution if time permits. The interactions of the ADMM, PR, and CDRSBK algorithms and the GEO, TRA, and TIN decomposition methods could be explored, so that we can exploit their advantages, in order to further achieve a best overall solution.

Chapter 8

Conclusions

This dissertation is motivated by the challenges in improving the performance of railway operations, in terms of punctuality, reliability, non-discrimination, capacity utilization, and energy efficiency, as outlined in Chapter 1.

Several research questions have been stated under the research objectives, which are answered throughout Chapters 2 to 7. This chapter summarizes the answers. Section 8.1 gives the main conclusions, and Section 8.2 recommends for future research directions.

8.1 Main contributions

Several research questions were proposed in Section 1.3, including 1 main question and 6 sub-questions. We now answer the proposed research questions.

Main question: ***Are there benefits of incorporating equity policy, preventive maintenance planning, or train control into railway traffic management by means of optimization approaches?***

The answer of the main research question is positive.

The benefit of incorporating equity policy into railway traffic management is reflected by the improved delay equity among competing train operation companies (TOCs) or trains. Moreover, in comparison with other scheduling algorithms (e.g., First-In-First-Out and First-Scheduled-First-Served), the proposed optimization approach in Chapter 3 can achieve better solutions for both delays and equity.

The benefit of incorporating preventive maintenance (PM) planning into railway traffic management is reflected by the reduction of the total train travel time, which further leads to the release of infrastructure capacity (i.e., more available capacity of the existing infrastructure), as shown in Chapter 4.

The integration of traffic management and train control enables us to assess energy consumption and train delay of train operations simultaneously. As explored in Chapters 5 and 6, aiming at both delay recovery and energy efficiency, the two objectives can be improved at the same time through managing the train speed, which reflects the benefit of their integration.

The main research question is briefly answered above and further detailed by answering the 6 sub-questions item-by-item as follows:

(1) ***How to equitably deal with the conflicting requests of competing train operation companies while dispatching trains?***

As reviewed in Sections 2.2.1 and 2.2.2 of Chapter 2 on real-time traffic management and on equitable capacity allocation (of the train timetabling problem) and equitable control of air traffic and road traffic, the approaches based on auctions and those based on scheduling are two common ways to allocate capacity with some consideration of equity. However, in railway transport system, equitable competition has been mostly considered and addressed during design and strategic planning, and the investigation of equitable (or non-discriminatory) traffic control is absent in the literature.

An optimization approach has been proposed in Chapter 3 for addressing the non-discriminatory railway traffic control problem, where the delay equity among multiple TOCs or trains is explicitly considered, in addition to minimizing the average (consecutive) train delay time. The delay equity is quantified as the degree of homogeneity of the delays faced by different trains or trains of different TOCs, formulated either as an objective or in a constraint. An inequitable (or discriminatory) situation occurs when some trains or some TOCs face much larger delays than other trains or TOCs. The proposed optimization approach can deal with the conflicting requests of competing TOCs (or trains) in an equitable manner. Each solution computed for any input determination has a satisfactory degree of equity, which can be accepted by all interested parties.

(2) ***How to jointly schedule trains and preventive maintenance tasks at the same time?***

As reviewed in Section 2.2.3 of Chapter 2 on the joint scheduling of trains and PM tasks, most existing studies on train scheduling focus on minimizing the total deviation times from an ideal timetable with pre-defined PM plans or without considering maintenance, while studies related to PM mostly concern minimizing total PM costs and delays of PM tasks. Only a few explicit discussions on the integration of these two problems are seen in the literature, and most of them schedule one function by minimizing its impact on the other function. Integrated optimization approaches that simultaneously schedule trains and PM tasks are absent in the literature.

Chapter 4 proposed a virtual-train-based formulation method to describe resources reservation and occupancy of trains and preventive maintenance time slots (PMTSs).

Specifically, the workspace of a PMTS can be described by the route (pre-defined with an origin and a destination) of a virtual train; the working time of a PMTS can be described by the running time and the safety headway of a virtual train; and the shape (rectangle or stairway[1]) of a PMTS can be described by the dwell times of a virtual train at passing stations. In such a way, each PMTS can be represented by a virtual train with a specifically designed safety headway. A capacity constraint is only enforced for the real trains and between the real trains and the virtual trains; it is not imposed for the virtual trains because the PM tasks for different lines can be implemented simultaneously at an interchange station. By applying the virtual-train-based formulation, all trains (including real trains and virtual trains representing PMTSs) can be jointly scheduled at the same time.

(3) ***Can the joint consideration of train scheduling and preventive maintenance planning bring any potential capacity of the existing infrastructure?***

Based on the proposed virtual-train-based formulation, an integrated optimization approach and a Lagrangian-relaxation-based solution approach have been proposed in Chapter 4 for jointly scheduling trains and PM tasks on a general railway network. In comparison with the commonly-used sequential scheduling method, the experimental results showed the benefits of the integrated optimization on train scheduling and PMTSs planning, i.e., the integrated scheduling method is at least as good as the commonly-used sequential scheduling method and up to 25% improvement can be achieved by the integrated scheduling method. Therefore, the answer of this research question is positive, i.e., the joint consideration of train scheduling and PM planning can improve the quality of the train and PMTS schedule and bring potential capacity of the existing infrastructure.

(4) ***How to incorporate driving actions (train control) into traffic management?***

As reviewed in Section 2.2.4 of Chapter 2 on the interaction of traffic management and train control, the vast majority of the optimization-based train rescheduling approaches has a common assumption that a fixed speed profile is used for each train, i.e., a pre-determined (constant) minimum running time is considered for each train, and the studies on train control mostly focus on trajectory optimization with a given running time, i.e., determining the driving regimes and the switching points, with the aim of minimizing energy consumption. In the literature, the available studies try to address their interaction and integration in a decomposed, iterative, or non-optimized manner; however, few authors deal with the integrated problem by employing mathematical optimization methods.

An integrated modeling approach has been presented in Chapter 5, which incorporates the representation of microscopic traffic regulations and speed trajectories

[1] If the starting times of PM tasks on a sequence of block sections are same, as well as the end times, then the blockage of the PM tasks on a time-space graph will result in a rectangle shape. If there are spaced starting times for the PM tasks on a sequence of block sections, then it will show a stairway shape on a time-space graph.

into a single optimization problem. The proposed modeling approach divides the speed of a train on a cell into three phases, i.e., incoming, cruising, and outgoing phases, while considering 4 time variables (e.g., the time point that a train starts or ends cruising, in addition to the departure and arrival times) for describing the state transition of a train on a cell. Train acceleration is considered as a piecewise constant function by giving a fixed switching point (breakpoint) of speed (e.g., 60 km/h) for each train category, while train deceleration is considered constant for a certain train category and different among train categories.

Based on the modeling approach, three integrated optimization approaches for real-time traffic management, while explicitly including train control, have been developed to deliver both a train dispatching solution (including train routes, orders, departure and arrival times at passing stations) and a train control solution (i.e., train speed trajectories). In these optimization approaches, train speed is considered variable, and the blocking time of a train on a cell dynamically depends on its real operating speed.

(5) ***Is an improvement in energy efficiency of train operations possible by means of integrating traffic management and train control?***

Two approaches have been developed in Chapter 6 for including the minimization of energy consumption into the integrated optimization problems of traffic management and train control (proposed in Chapter 5), with either nonlinear constraints or linearized constraints. These enable us to assess and optimize energy consumption and train delay of train operations simultaneously. The energy consumed for accelerating trains and for overcoming resistances is evaluated. Moreover, we consider the option of regenerative braking and present linear formulations to calculate the utilization of the energy obtained through regenerative braking.

According to the experimental results, the two objectives of delay recovery and energy efficiency can be improved at the same time (e.g., by up to 4.0% and 5.6% for the train delay and the energy consumption in one of the solutions) through managing the train speed. For the test case, the application of regenerative braking leads to about 13.1%-22.9% reduction of the total energy consumption. Those experimental results answer the research question, i.e., the improvement in energy efficiency of train operation can be achieved by the integration of traffic management and train control.

(6) ***Which distributed optimization approaches can be used to reduce the computation time of the integrated problem of traffic management and train control for large railway networks?***

In Chapter 7, three decomposition methods (i.e., a geography-based, a train-based, and a time-interval-based decomposition, abbreviated GEO, TRA, and TIN decomposition respectively) have been presented to split the whole optimization problem into several subproblems, and three distributed optimization approaches have been proposed for dealing with the couplings among subproblems. The three distributed optimization approaches under consideration are an Alternating Direction Method of Multipliers (ADMM) algorithm, a priority-rule-based (PR) algorithm, and a Cooperative Distributed Robust Safe But Knowledgeable (CDRSBK) algorithm.

In the experiments, the performance of the proposed approaches has been examined in terms of feasibility, estimated optimality, solution quality, and computational efficiency. Overall, the CDRSBK algorithm with the TRA decomposition and the ADMM algorithm with the GEO decomposition achieve the best performance on solution quality and satisfactory performance on computational efficiency. The ADMM algorithm with the TRA decomposition shows the best computational efficiency but gives a relatively bad solution.

8.2 Recommendations for future research

In this section, we recommend several directions for future research. To extend this dissertation from a theoretical perspective, the following directions are given:

- Complex interlocking systems can be incorporated in the optimization problems, by refining the concept of cells. This would allow to enlarge the set of routes in station areas, as well as including more processes at stations, like turn-around or shunting.

- A direction goes towards studying how to best structure the original timetable, with the objective of ensuring equitable traffic control in operations. This would describe the impact of the timetable beyond robustness and resilience against small delays in operations (see Bešinović et al., 2016). Moreover, a comprehensive framework can be defined where equitable planning (equitable capacity allocation) and equitable control can be considered at the same time, to reach non-discriminatory operations at a system level.

- The relation between maintenance plans and reliability of train services can be defined by considering the risk associated with delaying maintenance, for being able to (re-)schedule traffic in a closed-loop perspective (Corman and Quaglietta, 2015) or within a robust optimization framework (Meng et al., 2016), in order to further reduce the system cost and achieve the largest economic benefits.

- With all kinds of uncertainties, e.g., the unexpected longer duration of maintenance and the unexpected extra time for boarding and alighting of passengers or loading and unloading of goods, that often occur in real operations, stochastic optimization approaches should be sought for the real-time traffic management problem, in order to generate a robust train dispatching solution, which can be less sensitive to uncertainties as much as possible. This leads to the exploration of the trade-off between the quality and the robustness of a dispatching solution.

The following two extensions are made from a practical perspective:

- Embedded with the proposed optimization approaches for the integration of traffic management and train control, a comprehensive system could be developed to integrate the multiple steps in the solving procedure, e.g., the preprocessing step for generating a set (or an efficient subset) of the possible train speed profile options, the solving step to solve the optimization problem, and the displaying step to show train timetables and speed-space graphs. This system can be used as a decision support tool for both train dispatcher and train driver, in order to bring the research presented in this dissertation into practice.
- For practical applications of the distributed optimization approaches, a promising two-step procedure can be used: first generate a feasible solution in short time (e.g., by applying the ADMM algorithm) and then improve the solution quality (by using the CDRSBK algorithm) based on that feasible solution if time permits. This leads to one direction of the future research on exploring the interactions of the ADMM, PR, and CDRSBK algorithms and the GEO, TRA, and TIN decomposition methods, so that we can exploit their advantages, in order to further achieve a best overall solution.

Bibliography

Ahmed, Q. I., Lu, H., and Ye, S. (2008). Urban transportation and equity: A case study of Beijing and Karachi. *Transportation Research Part A: Policy and Practice*, 42(1):125–139.

Albrecht, A., Howlett, P., Pudney, P., Vu, X., and Zhou, P. (2013a). Using timing windows to allow energy-efficient driving. In *Proceedings of the 10th World Congress on Rail Research. Sydney, Australia*.

Albrecht, A. R., Howlett, P. G., Pudney, P. J., and Vu, X. (2013b). Energy-efficient train control: from local convexity to global optimization and uniqueness. *Automatica*, 49(10):3072–3078.

Albrecht, A. R., Panton, D. M., and Lee, D. H. (2013c). Rescheduling rail networks with maintenance disruptions using problem space search. *Computers & Operations Research*, 40(3):703–712.

Albrecht, T. (2009). The influence of anticipating train driving on the dispatching process in railway conflict situations. *Networks and Spatial Economics*, 9(1):85–101.

Albrecht, T., Binder, A., and Gassel, C. (2011). An overview on real-time speed control in rail-bound public transportation systems. In *Proceedings of the 2nd International Conference on Models and Technologies for Intelligent Transportation Systems. Leuven, Belgium*, pages 1–4.

Banedanmark (2017). Report on train punctuality. `http://rigsrevisionen.dk/publications/2017/32017/`.

Beltran Royoa, C. and Heredia, F. J. (2002). Unit commitment by augmented Lagrangian relaxation: Testing two decomposition approaches. *Journal of Optimization Theory and Applications*, 112(2):295–314.

Bemporad, A. and Morari, M. (1999). Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427.

Bešinović, N., Goverde, R. M., Quaglietta, E., and Roberti, R. (2016). An integrated micro–macro approach to robust railway timetabling. *Transportation Research Part B: Methodological*, 87:14–32.

Boizumeau, J., Leguay, P., and Navarro, E. (2011). Braking energy recovery at the Rennes metro. In *Proceedings of the Workshop on Braking Energy Recovery Systems–Ticket to Kyoto. Bielefeld, Germany.*

Boland, N., Kalinowski, T., Waterer, H., and Zheng, L. (2013). Mixed integer programming based maintenance scheduling for the hunter valley coal chain. *Journal of Scheduling*, 16(6):649–659.

Borndorfer, R., Grotschel, M., Lukac, S., and Mitusch, K. (2006). An auctioning approach to railway slot allocation. *Competition and Regulation in Network Industries*, 7:163.

Boston Consulting Group (2017). The 2017 European Railway Performance Index.

Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122.

Brännlund, U., Lindberg, P. O., Nou, A., and Nilsson, J.-E. (1998). Railway timetabling using Lagrangian relaxation. *Transportation Science*, 32(4):358–369.

Brünger, O. and Dahlhaus, E. (2008). *Railway Timetable & Traffic-Analysis, Modelling, Simulation (Chapter Running Time Estimation, pp. 58-82).* Eurail Press.

Budai, G., Dekker, R., and Nicolai, R. P. (2008). Maintenance and production: a review of planning models. In *Complex System Maintenance Handbook*, pages 321–344. Springer.

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

Caimi, G., 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.

Caimi, G., Laumanns, M., Schüpbach, K., Wörner, S., and Fuchsberger, M. (2011). The periodic service intention as a conceptual framework for generating timetables with partial periodicity. *Transportation Planning and Technology*, 34(4):323–339.

Caprara, A., Fischetti, M., and Toth, P. (2002). Modeling and solving the train timetabling problem. *Operations Research*, 50(5):851–861.

Caprara, A., Monaci, M., Toth, P., and Guida, P. L. (2006). A Lagrangian heuristic algorithm for a real-world train timetabling problem. *Discrete Applied Mathematics*, 154(5):738–753.

Chevrier, R., Pellegrini, P., and Rodriguez, J. (2013). Energy saving in railway timetabling: A bi-objective evolutionary approach for computing alternative running times. *Transportation Research Part C: Emerging Technologies*, 37:20–41.

Corman, F., D'Ariano, A., Hansen, I. A., and Pacciarelli, D. (2011a). Optimal multi-class rescheduling of railway traffic. *Journal of Rail Transport Planning & Management*, 1(1):14–24.

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. (2010). 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., Pranzo, M., and Hansen, I. A. (2011b). Effectiveness of dynamic reordering and rerouting of trains in a complicated and densely occupied station area. *Transportation Planning and Technology*, 34(4):341–362.

Corman, F. and Meng, L. (2015). A review of online dynamic models and algorithms for railway traffic management. *IEEE Transactions on Intelligent Transportation Systems*, 16(3):1274–1284.

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.

D'Ariano, A. and Albrecht, T. (2010). Running time re-optimization during real-time timetable perturbations. *Part C of Timetable Planning and Information Quality*, WIT Press:147–156.

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

D'Ariano, A., Pranzo, M., and Hansen, I. A. (2007b). Conflict resolution and train speed coordination for solving real-time timetable perturbations. *IEEE Transactions on Intelligent Transportation Systems*, 8(2):208–222.

Davis, W. J. (1926). The tractive resistance of electric locomotives and cars. *General Electric Review*, 29:685–708.

De Poza, I. d. P. y., Ruiz, M. A. V., and Goodchild, C. (2009). Assessing fairness and equity in trajectory based operations. In *Proceedings of the 9th AIAA Aviation Technology, Integration, and Operations Conference (ATIO), South Carolina, USA*.

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

European Commission (1991). Council Directive 91/440/EEC on the Development of the Community's Railways.

European Commission (2001). Council Directive 2001/14/EC on the Allocation of Railway Infrastructure Capacity and the Levying of Charges for the Use of Railway Infrastructure and Safety Certification.

European Commission (2015). Study on the Cost and Contribution of the Rail Sector.

European Commission (2018). Europeans' Satisfaction with Passenger Rail Services.

Fang, W., Yang, S., and Yao, X. (2015). A survey on problem models and solution approaches to rescheduling in railway networks. *IEEE Transactions on Intelligent Transportation Systems*, 16(6):2997–3016.

Findler, N. V. and Stapp, J. (1992). Distributed approach to optimized control of street traffic signals. *Journal of Transportation Engineering*, 118(1):99–110.

Forsgren, M., Aronsson, M., and Gestrelius, S. (2013). Maintaining tracks and traffic flow at the same time. *Journal of Rail Transport Planning & Management*, 3(3):111–123.

Garcia-Tabares, L., Iglesias, J., Lafoz, M., Martinez, J., Vazquez, C., Tobajas, C., Gomez-Alors, A., Echeandia, A., Lucas, J., Zuazo, C., et al. (2011). Development and testing of a 200 MJ/350 kW kinetic energy storage system for railways applications. In *Proceedings of the 9th World Congress on Railway Research, Lille, France*, pages 22–26.

Ghaviha, N., Campillo, J., Bohlin, M., and Dahlquist, E. (2017). Review of application of energy storage devices in railway transportation. *Energy Procedia*, 105:4561–4568.

Ginkel, A. and Schöbel, A. (2007). To wait or not to wait? The bicriteria delay management problem in public transportation. *Transportation Science*, 41(4):527–538.

Glover, C. N. and Ball, M. O. (2013). Stochastic optimization models for ground delay program planning with equity-efficiency tradeoffs. *Transportation Research Part C: Emerging Technologies*, 33:196–202.

González-Gil, A., Palacin, R., and Batty, P. (2013). Sustainable urban rail systems: Strategies and technologies for optimal management of regenerative braking energy. *Energy Conversion and Management*, 75:374–388.

Hadidi, L. A., Al-Turki, U. M., Rahim, A., et al. (2012). Integrated models in production planning and scheduling, maintenance and quality: a review. *International Journal of Industrial and Systems Engineering*, 10(1):21.

Hansen, H. S., Nawaz, M. U., and Olsson, N. (2017). Using operational data to estimate the running resistance of trains. estimation of the resistance in a set of Norwegian tunnels. *Journal of Rail Transport Planning & Management*, 7(1–2):62–76.

Hansen, I. A. and Pachl, J. (2014). *Railway Timetabling & Operations: Analysis, Modelling, Optimisation, Simulation, Performance Evaluation*. Eurailpress, Hamburg, Germany.

Harrod, S. (2011). Modeling network transition constraints with hypergraphs. *Transportation Science*, 45(1):81–97.

Harrod, S. (2013). Auction pricing of network access for North American railways. *Transportation Research Part E: Logistics and Transportation Review*, 49(1):176–189.

Hoffman, R. L. and Davidson, G. (2003). Equitable allocation of limited resources(EALR) - defining, measuring, and implementing equity. In *FAA Free Flight Program, Metro Aviation, Virginia, USA*.

Howlett, P. (2000). The optimal control of a train. *Annals of Operations Research*, 98(1-4):65–87.

Howlett, P. G. and Pudney, P. J. (2012). *Energy-Efficient Train Control*. Springer Science & Business Media.

INFORMS RAS (2012). Institute for Operations Research and Management Sciences (INFORMS) Railroad Application Section (RAS) problem solving competition. http://connect.informs.org/railway-applications/awards/problem-solving-competition/2012.

Karsu, Ö. and Morton, A. (2015). Inequity averse optimization in operational research. *European Journal of Operational Research*, 245(2):343–359.

Kecman, P., Corman, F., D'Ariano, A., and Goverde, R. M. (2013). Rescheduling models for railway traffic management in large-scale networks. *Public Transport*, 5(1-2):95–123.

Kersbergen, B., van den Boom, T., and De Schutter, B. (2016). Distributed model predictive control for railway traffic management. *Transportation Research Part C: Emerging Technologies*, 68:462–489.

Khaligh, A. and Li, Z. (2010). Battery, ultracapacitor, fuel cell, and hybrid energy storage systems for electric, hybrid electric, fuel cell, and plug-in hybrid electric vehicles: State of the art. *IEEE transactions on Vehicular Technology*, 59(6):2806–2814.

Khmelnitsky, E. (2000). On an optimal control problem of train operation. *IEEE Transactions on Automatic Control*, 45(7):1257–1266.

Kim, A. and Hansen, M. (2013). A framework for the assessment of collaborative en route resource allocation strategies. *Transportation Research Part C: Emerging Technologies*, 33:324–339.

Kuhn, K. D. (2013). Ground delay program planning: Delay, equity, and computational complexity. *Transportation Research Part C: Emerging Technologies*, 35:193–203.

Kurosaki, F. (2008). *An Analysis of Vertical Separation of Railways*. PhD thesis, University of Leeds.

Kuwata, Y. and How, J. P. (2011). Cooperative distributed robust trajectory optimization using receding horizon milp. *IEEE Transactions on Control Systems Technology*, 19(2):423–431.

Lamorgese, L., Mannino, C., and Piacentini, M. (2016). Optimal train dispatching by benders-like reformulation. *Transportation Science*, 50(3):910–925.

Li, X. and Lo, H. K. (2014a). An energy-efficient scheduling and speed control approach for metro rail operations. *Transportation Research Part B: Methodological*, 64:73–89.

Li, X. and Lo, H. K. (2014b). Energy minimization in dynamic train scheduling and control for metro rail operations. *Transportation Research Part B: Methodological*, 70:269–284.

Lidén, T. and Joborn, M. (2016). Dimensioning windows for railway infrastructure maintenance: Cost efficiency versus traffic impact. *Journal of Rail Transport Planning & Management*, 6(1):32–47.

Lidén, T. and Joborn, M. (2017). An optimization model for integrated planning of railway traffic and network maintenance. *Transportation Research Part C: Emerging Technologies*, 74:327–347.

Litman, T. (2002). Evaluating transportation equity. *World Transport Policy and Practice*, 8(2):50–65.

Liu, R. R. and Golovitcher, I. M. (2003). Energy-efficient operation of rail vehicles. *Transportation Research Part A: Policy and Practice*, 37(10):917–932.

Lu, Q. and Feng, X. (2011). Optimal control strategy for energy saving in trains under the four-aspect fixed autoblock system. *Journal of Modern Transportation*, 19(2):82–87.

Luan, X., Corman, F., and Meng, L. (2017a). Non-discriminatory train dispatching in a rail transport market with multiple competing and collaborative train operating companies. *Transportation Research Part C: Emerging Technologies*, 80:148–174.

Luan, X., Miao, J., Meng, L., Corman, F., and Lodewijks, G. (2017b). Integrated optimization on train scheduling and preventive maintenance time slots planning. *Transportation Research Part C: Emerging Technologies*, 80:329–359.

Luan, X., Wang, Y., De Schutter, B., Meng, L., Lodewijks, G., and Corman, F. (2018a). Integration of real-time traffic management and train control for rail networks-part 1: Optimization problems and solution approaches. *Transportation Research Part B: Methodological*, 115:41–71.

Luan, X., Wang, Y., De Schutter, B., Meng, L., Lodewijks, G., and Corman, F. (2018b). Integration of real-time traffic management and train control for rail networks-part 2: Extensions towards energy-efficient train operations. *Transportation Research Part B: Methodological*, 115:72–94.

Lüthi, M. (2009). *Improving the Efficiency of Heavily Used Railway Networks through Integrated Real-time Rescheduling*. PhD thesis, ETH Zürich.

Manley, B. and Sherry, L. (2010). Analysis of performance and equity in ground delay programs. *Transportation Research Part C: Emerging Technologies*, 18(6):910–920.

Mazzarello, M. and Ottaviani, E. (2007). A traffic management system for real-time traffic optimisation in railways. *Transportation Research Part B: Methodological*, 41(2):246–274.

Meinel, M., Ulbrich, M., and Albrecht, S. (2014). A class of distributed optimization methods with event-triggered communication. *Computational Optimization and Applications*, 57(3):517–553.

Meinert, M. (2009). New mobile energy storage system for rolling stock. In *Proceedings of the 13th European Conference on Power Electronics and Applications*, pages 1–10. IEEE.

Meng, L., Luan, X., and Zhou, X. (2016). A train dispatching model under a stochastic environment: stable train routing constraints and reformulation. *Networks and Spatial Economics*, 16(3):791–820.

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.

Montigel, M. (2009). Operations control system in the Lötschberg base tunnel. *Railway Technical Review*, 2:43–44.

Mu, S. and Dessouky, M. (2011). Scheduling freight trains traveling on complex networks. *Transportation Research Part B: Methodological*, 45(7):1103–1123.

Narayanaswami, S. and Rangaraj, N. (2011). Scheduling and Rescheduling of Railway Operations: A Review and Expository Analysis. *Technology Operation Management*, 2(2):102–122.

Nash, C. and Rivera-Trujillo, C. (2004). Rail regulatory reform in europe–principles and practice. In *STELLA Focus Group 5 synthesis meeting, Athens*.

Nederlandse Spoorwegen (2017). NS Annual Report 2017. `https://www.nsjaarverslag.nl/FbContent.ashx/pub_1000/downloads/v180419111054/NS_annualreport_2017.pdf`.

Nedic, A. and Ozdaglar, A. (2010). Cooperative distributed multi-agent optimization. In *Convex Optimization in Signal Processing and Communications*, pages 340–386. Cambridge University Press.

Negenborn, R. R., De Schutter, B., and Hellendoorn, J. (2008). Multi-agent model predictive control for transportation networks: Serial versus parallel schemes. *Engineering Applications of Artificial Intelligence*, 21(3):353–366.

Network Rail (2017). Punctuality on the national rail network. `https://www.networkrail.co.uk/who-we-are/how-we-work/performance/public-performance-measure/punctuality-national-rail-network/`.

Ogasa, M. (2010). Application of energy storage technologies for electric railway vehiclesexamples with hybrid electric railway vehicles. *IEEJ Transactions on Electrical and Electronic Engineering*, 5(3):304–311.

On-Time (2014). Optimal networks for train integration management across Europe. `http://www.ontime-project.eu/aboutproject.aspx`.

Pachl, J. (2009). *Railway Operation and Control (2nd edn)*. VTD Rail Publishing, Mountlake Terrace.

Pellegrini, P., Marlière, G., Pesenti, R., and Rodriguez, J. (2015). RECIFE-MILP: An effective MILP-based heuristic for the real-time railway traffic management problem. *IEEE Transactions on Intelligent Transportation Systems*, 16(5):2609–2619.

Pellegrini, P., Marlière, G., and Rodriguez, J. (2014). Optimal train routing and scheduling for managing traffic perturbations in complex junctions. *Transportation Research Part B: Methodological*, 59:58–80.

Pellegrini, P. and Rodriguez, J. (2013). Single European sky and single European railway area: A system level analysis of air and rail transportation. *Transportation Research Part A: Policy and Practice*, 57:64–86.

Peng, F., Kang, S., Li, X., Ouyang, Y., Somani, K., and Acharya, D. (2011). A heuristic approach to the railroad track maintenance scheduling problem. *Computer-Aided Civil and Infrastructure Engineering*, 26(2):129–145.

Pudney, P. and Wardrop, A. (2008). Generating train plans with problem space search. In *Computer-aided systems in public transport*, pages 195–207. Springer.

Quaglietta, E., Corman, F., and Goverde, R. M. (2013). Stability analysis of railway dispatching plans in a stochastic and dynamic environment. *Journal of Rail Transport Planning & Management*, 3(4):137–149.

Quaglietta, E., Pellegrini, P., Goverde, R. M., Albrecht, T., Jaekel, B., Marlière, G., Rodriguez, J., Dollevoet, T., Ambrogio, B., Carcasole, D., et al. (2016). The on-time real-time railway traffic management framework: A proof-of-concept using a scalable standardised data communication architecture. *Transportation Research Part C: Emerging Technologies*, 63:23–50.

Rao, X., Montigel, M., and Weidmann, U. (2016). A new rail optimisation model by integration of traffic management and train automation. *Transportation Research Part C: Emerging Technologies*, 71:382–405.

Research Collection ETH Zurich (2018). Detailed experimental results of Chapter 5. `https://www.research-collection.ethz.ch/handle/20.500.11850/256447`.

Rios, J. and Ross, K. (2007). Delay Optimization for Airspace Capacity Management with Runtime and Equity Considerations. In *In AIAA Guidance, Navigation and Control Conference and Exhibit, Hilton Head, SC*.

Rodrigo, E., Tapia, S., Mera, J., and Soler, M. (2013). Optimizing electric rail energy consumption using the lagrange multiplier technique. *Journal of Transportation Engineering*, 139(3):321–329.

Rodriguez, J. (2007). A constraint programming model for real-time train scheduling at junctions. *Transportation Research Part B: Methodological*, 41(2):231–245.

Samà, M., Pellegrini, P., D'Ariano, A., Rodriguez, J., and Pacciarelli, D. (2016). Ant colony optimization for the real-time train routing selection problem. *Transportation Research Part B: Methodological*, 85:89–108.

Santos, B., Antunes, A., and Miller, E. J. (2008). Integrating equity objectives in a road network design model. *Transportation Research Record: Journal of the Transportation Research Board*, (2089):35–42.

Schachtebeck, M. and Schöbel, A. (2010). To wait or not to wait-and who goes first? Delay management with priority decisions. *Transportation Science*, 44(3):307–321.

Scheepmaker, G. M. and Goverde, R. M. (2016). Energy-efficient train control including regenerative braking with catenary efficiency. In *Proceedings of the IEEE International Conference on Intelligent Rail Transportation (ICIRT)*, pages 116–122. IEEE.

Scheepmaker, G. M., Goverde, R. M., and Kroon, L. G. (2017). Review of energy-efficient train control and timetabling. *European Journal of Operational Research*, 257(2):355–376.

Schlechte, T. (2011). *Railway Track Allocation: Models and Algorithms*. PhD thesis, Technischen Universität Berlin, Germay.

Schöbel, A. (2007). Integer programming approaches for solving the delay management problem. In *Algorithmic Methods for Railway Optimization*, pages 145–170. Springer.

Shimada, M., Oishi, R., Araki, D., and Nakamura, Y. (2010). Energy storage system for effective use of regenerative energy in electrified railways. *Hitachi Review*, 59(1):33–38.

Siemens (2011). Increasing energy efficiency optimized traction power supply in mass transit systems. `https://w3.usa.siemens.com/mobility/us/Documents/en/rail-solutions/railway-electrification/dc-traction-power-supply/increasing-energy-efficiency-en.pdf`.

Steiner, M., Klohr, M., and Pagiela, S. (2007). Energy storage system with ultracaps on board of railway vehicles. In *Proceedings of the European Conference on Power Electronics and Applications*, pages 1–10.

Su, S., Tang, T., and Wang, Y. (2016). Evaluation of strategies to reducing traction energy consumption of metro systems using an optimal train control simulation model. *Energies*, 9(2):105.

Törnquist, J. (2012). Design of an effective algorithm for fast response to the rescheduling of railway traffic during disturbances. *Transportation research Part C: Emerging technologies*, 20(1):62–78.

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

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

Tuyttens, D., Fei, H., Mezmaz, M., and Jalwan, J. (2013). Simulation-based genetic algorithm towards an energy-efficient railway traffic control. *Mathematical Problems in Engineering*, 2013.

UIC (2002). Regenerative braking in DC systems. `http://www.railway-energy.org/tfee/index.php?ID=220&TECHNOLOGYID=103&SEL=210&EXPANDALL=3`.

UIC (2012). Moving towards sustainable mobility: a strategy for 2030 and beyond for the European railway sector. `http://www.cer.be/sites/default/files/publication/CER-UIC_Sustainable_Mobility_Strategy_-_SUMMARY.pdf`.

Vansteenwegen, P., Dewilde, T., Burggraeve, S., and Cattrysse, D. (2016). An iterative approach for reducing the impact of infrastructure maintenance on the performance of railway systems. *European Journal of Operational Research*, 252(1):39–53.

V/Line (2017). Performance and capacity. `https://www.vline.com.au/About-V-Line/Performance`.

Vossen, T., Ball, M., Hoffman, R., and Wambsganss, M. (2003). A general approach to equity in traffic flow management and its application to mitigating exemption bias in ground delay programs. *Air Traffic Control Quarterly*, 11(4):277–292.

Wang, P. and Goverde, R. M. (2016). Multiple-phase train trajectory optimization with signalling and operational constraints. *Transportation Research Part C: Emerging Technologies*, 69:255–275.

Wang, P. and Goverde, R. M. (2017). Multi-train trajectory optimization for energy efficiency and delay recovery on single-track railway lines. *Transportation Research Part B: Methodological*, 105:340–361.

Wang, P. and Goverde, R. M. (2019). Multi-train trajectory optimization for energy-efficient timetabling. *European Journal of Operational Research*, 272(2):621–635.

Wang, Y., De Schutter, B., van den Boom, T. J., and Ning, B. (2013). Optimal trajectory planning for trains–a pseudospectral method and a mixed integer linear programming approach. *Transportation Research Part C: Emerging Technologies*, 29:97–114.

Wang, Y., De Schutter, B., van den Boom, T. J. J., and Ning, B. (2012). Optimal trajectory planning for trains under operational constraints using mixed integer linear programming. In *Proceedings of the 13th IFAC Symposium on Control in Transportation Systems (CTS2012). Sofia, Bulgaria*, pages 158–163.

Wang, Y., Ning, B., Cao, F., De Schutter, B., and van den Boom, T. J. (2011). A survey on optimal trajectory planning for train operations. In *Proceedings of the 2011 IEEE International Conference on Service Operations, Logistics, and Informatics (SOLI). Beijing, China*, pages 589–594.

Wang, Y., Ning, B., van den Boom, T., and De Schutter, B. (2016). *Optimal Trajectory Planning and Train Scheduling for Urban Rail Transit Systems*. Springer.

Wangermann, J. P. and Stengel, R. F. (1996). Distributed optimization and principled negotiation for advanced air traffic management. In *Proceedings of the IEEE International Symposium on Intelligent Control*, pages 156–161.

Williams, H. P. (2013). *Model Building in Mathematical Programming*. John Wiley & Sons.

Wu, D., Yin, Y., Lawphongpanich, S., and Yang, H. (2012). Design of more equitable congestion pricing and tradable credit schemes for multimodal transportation networks. *Transportation Research Part B: Methodological*, 46(9):1273–1287.

Wu, J., Liu, M., Sun, H., Li, T., Gao, Z., and Wang, D. Z. (2015). Equity-based timetable synchronization optimization in urban subway network. *Transportation Research Part C: Emerging Technologies*, 51:1–18.

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*, 104:638 – 666.

Xu, X., Li, K., Yang, L., and Ye, J. (2014). Balanced train timetabling on a single-line railway with optimized velocity. *Applied Mathematical Modelling*, 38(3):894–909.

Yang, H. and Zhang, X. (2002). Multiclass network toll design problem with social and spatial equity constraints. *Journal of Tranportation Engineering*, 128(5):420–428.

Yang, X., Li, X., Ning, B., and Tang, T. (2016). A survey on energy-efficient train operation for urban rail transit. *IEEE Transactions on Intelligent Transportation Systems*, 17(1):2–13.

Yang, X., Ning, B., Li, X., and Tang, T. (2014). A two-objective timetable optimization model in subway systems. *IEEE Transactions on Intelligent Transportation Systems*, 15(5):1913–1921.

Yin, Y. and Yang, H. (2004). Optimal tolls with a multiclass, bicriterion traffic network equilibrium. *Transportation Research Record: Journal of the Transportation Research Board*, (1882):45–52.

Zhan, S., Kroon, L. G., 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.

Zhong, M. (2012). *Models and Solution Algorithms for Equitable Resource Allocation in Air Traffic Flow Management*. PhD thesis, University of Maryland, USA.

Zhou, L., Tong, L., Chen, J., Tang, J., and Zhou, X. (2017). Joint optimization of high-speed train timetables and speed profiles: A unified modeling approach using space-time-speed grid networks. *Transportation Research Part B: Methodological*, 97:157 – 181.

Appendix A

A.1 Additional explanations of the formulations in Section 5.4

In Section 5.4, we have introduced six logical speed indicators $\zeta_{1,f,i,j}$, ..., $\zeta_{6,f,i,j}$ to indicate the actions taken by train f on cell (i,j), i.e., the train trajectory. Some constraints, e.g., (5.26) and (5.28), further employ these indicators to perform their functions. For assisting the readers to understand our formulations, we here describe the six logical speed indicators in detail, and then we explain how these indicators play a role in other constraints. In the remainder of this section, we omit the subscripts f,i,j of the parameters and variables to improve the readability, e.g., the incoming speed is denoted as v^{in}, and the acceleration is indicated as α_1 when the train speed is less than the switching speed v^{turn} (the speed point for switching the train acceleration) and as α_2 when the train speed is larger than the switching speed v^{turn}.

A.1.1 Explanation of the six logical speed indicators $\zeta_{1,f,i,j},\ldots,\zeta_{6,f,i,j}$ in Table 5.2

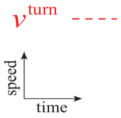
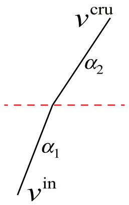
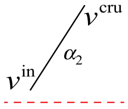
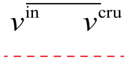
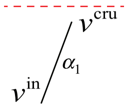
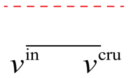
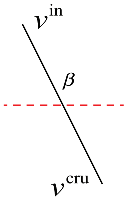
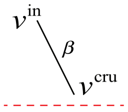
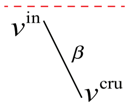
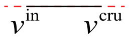
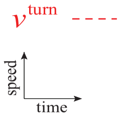
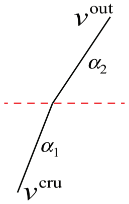
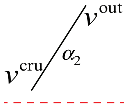
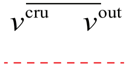
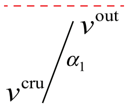
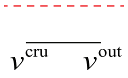
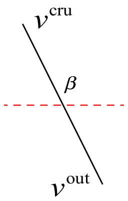
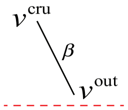
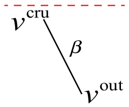
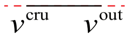
Table A.1 summarizes all possible train trajectories, i.e., the action(s) that a train may take, in the incoming and outgoing phases respectively.

As presented, there are 9 possible trajectories for each phase. Each scenario can be represented by the speed indicators ζ_1,ζ_3, and ζ_4 for the incoming phase or by the speed indicators ζ_2,ζ_5, and ζ_6 for the outgoing phase. Regarding the cruising phase, the train speed is constant, so only one train trajectory is possible, like “Trajectory_3”, “Trajectory_5”, and “Trajectory_9”.

A.1.2 Explanation of (5.26)

Constraints (5.26a)-(5.26e) are proposed for the incoming phase by employing the speed indicators and by satisfying the formula of the uniformly accelerating and decelerating motions, i.e., for such a motion with an initial speed v_0, a final speed v_t, and

Table A.1: Summary of the possible train trajectories and the corresponding value of the speed indicators

<table>
<tr><th></th><th>Trajectory ID</th><th>Trajectory_1</th><th>Trajectory_2</th><th>Trajectory_3</th><th>Trajectory_4</th><th>Trajectory_5</th><th>Trajectory_6</th><th>Trajectory_7</th><th>Trajectory_8</th><th>Trajectory_9</th></tr>
<tr><td rowspan="4">Incoming phase</td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr>
<tr><td>ζ_1</td><td>1</td><td colspan="2">1</td><td colspan="2">1</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>ζ_3</td><td>0</td><td colspan="2">1</td><td colspan="2">0</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>ζ_4</td><td>0</td><td colspan="2">0</td><td colspan="2">1</td><td>1</td><td>0</td><td>1</td><td>1</td></tr>
<tr><th></th><th>Trajectory ID</th><th>Trajectory_10</th><th>Trajectory_11</th><th>Trajectory_12</th><th>Trajectory_13</th><th>Trajectory_14</th><th>Trajectory_15</th><th>Trajectory_16</th><th>Trajectory_17</th><th>Trajectory_18</th></tr>
<tr><td rowspan="4">Outgoing phase</td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr>
<tr><td>ζ_2</td><td>1</td><td colspan="2">1</td><td colspan="2">1</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>ζ_5</td><td>0</td><td colspan="2">1</td><td colspan="2">0</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>ζ_6</td><td>0</td><td colspan="2">0</td><td colspan="2">1</td><td>1</td><td>0</td><td>1</td><td>1</td></tr>
</table>

a steady acceleration α, the elapsed time for accelerating from speed v_0 to speed v_t is $\Delta t = \frac{v_t - v_0}{\alpha}$. As shown in Table A.2, constraints (5.26a)-(5.26e) represent the 9 possible trajectories for the incoming phase in Table A.1.

Table A.2: Overview of the details of (5.26)

Constraint	Corresponding trajectory	Value of the speed indicators			Reduced equation
	ID	ζ_1	ζ_3	ζ_4	
(5.26a)	Trajectory_6, Trajectory_7, and Trajectory_8	0	0 or 1	0 or 1	$a^{\text{cru}} - a = -\frac{v^{\text{cru}} - v^{\text{in}}}{\beta}$
(5.26b)	Trajectory_2, Trajectory_3, and Trajectory_9	1	1	0 or 1	$a^{\text{cru}} - a = \frac{v^{\text{cru}} - v^{\text{in}}}{\alpha_2}$
(5.26c)	Trajectory_4, Trajectory_5, and Trajectory_9	1	0 or 1	1	$a^{\text{cru}} - a = \frac{v^{\text{cru}} - v^{\text{in}}}{\alpha_1}$
(5.26d)	Trajectory_1	1	0	0	$a^{\text{turn}} - a = \frac{v^{\text{turn}} - v^{\text{in}}}{\alpha_1}$
(5.26e)	Trajectory_1	1	0	0	$a^{\text{cru}} - a^{\text{turn}} = \frac{v^{\text{cru}} - v^{\text{turn}}}{\alpha_2}$

Regarding the cases of “Trajectory_3”, “Trajectory_5”, and “Trajectory_9”, as the incoming speed v^{in} equals the cruising speed v^{cru}, the incoming phase does not exist anymore, and the condition $a^{\text{cru}} = a$ is required by (5.26b) and (5.26c). Note that similar constraints can be constructed to represent the “Trajectory_10”, ..., “Trajectory_18” for the outgoing phase in Table A.1. We do not present those details here.

A.1.3 Explanation of (5.28)

Constraints (5.28a)-(5.28d) are proposed for calculating the distance L^{in} that a train travels within a cell in the incoming phase. These constraints also satisfy the formula of the uniformly accelerating and decelerating motions, i.e., for such a motion with an initial speed v_0, a final speed v_t, and a steady acceleration α, the distance traveled for accelerating from speed v_0 to speed v_t is $L = \frac{{v_t}^2 - {v_0}^2}{2 \cdot \alpha}$. As shown in Table A.3, constraints (5.28a)-(5.28d) represent the 9 possible trajectories for the incoming phase in Table A.1.

Table A.3: Overview of the details of (5.28)

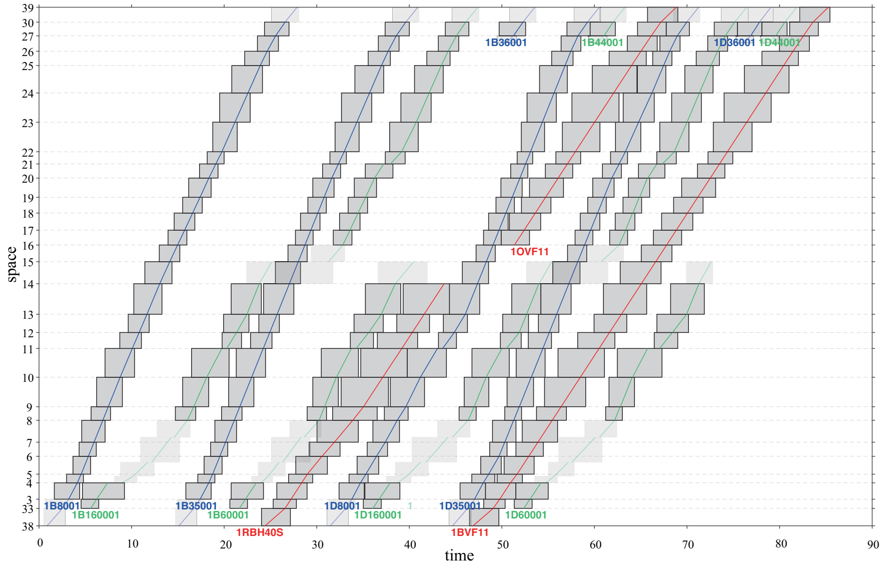
Constraint	Corresponding trajectory	Value of the speed indicators			Reduced equation
	ID	ζ_1	ζ_3	ζ_4	
(5.28a)	Trajectory_6, Trajectory_7, and Trajectory_8	0	0 or 1	0 or 1	$L^{\text{in}} = -\frac{(v^{\text{cru}})^2 - (v^{\text{in}})^2}{2 \cdot \beta}$
(5.28b)	Trajectory_2, Trajectory_3, and Trajectory_9	1	1	0 or 1	$L^{\text{in}} = \frac{(v^{\text{cru}})^2 - (v^{\text{in}})^2}{2 \cdot \alpha_2}$
(5.28c)	Trajectory_4, Trajectory_5, and Trajectory_9	1	0 or 1	1	$L^{\text{in}} = \frac{(v^{\text{cru}})^2 - (v^{\text{in}})^2}{2 \cdot \alpha_1}$
(5.28d)	Trajectory_1	1	0	0	$L^{\text{in}} = \frac{(v^{\text{turn}})^2 - (v^{\text{in}})^2}{2 \cdot \alpha_1} + \frac{(v^{\text{cru}})^2 - (v^{\text{turn}})^2}{2 \cdot \alpha_2}$

Regarding the “Trajectory_3”, “Trajectory_5”, and “Trajectory_9”, as the incoming speed v^{in} equals the cruising speed v^{cru}, the incoming phase does not exist anymore, and then the distance L^{in} equals zero according to (5.28b) and (5.28c). Note that similar constraints can be constructed to represent the “Trajectory_10”, ..., “Trajectory_18” in Table A.1, for calculating the distance L^{out} that a train runs over on a cell in the outgoing phase. We do not present those details here.

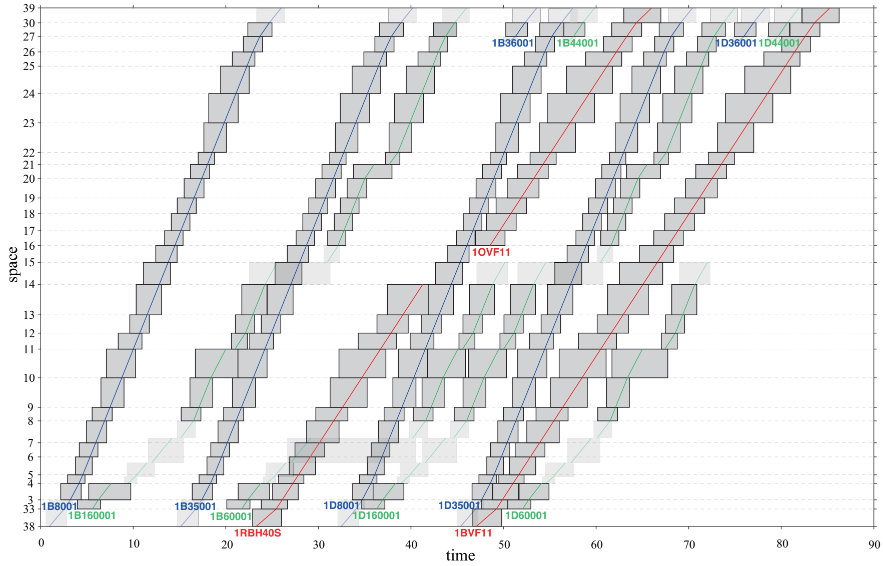
A.2 Illustration of the train timetables

We report here the train timetables of a representative case for the Dutch test case (regarding the experiments in Section 5.6.2(1)), obtained by the $\mathrm{P_{NLP}}$ problem (in Figure A.1(a)) and the $\mathrm{P_{TSPO}}$ problem (an initial solution in Figure A.1(b) and a secondary solution in Figure A.1(c)) respectively. Figure A.1(d) then provides the speed-space graphs for all trains, corresponding to the train timetables given in Figures A.1(a) to A.1(c). As there are siding tracks in some station areas, it is hard to draw every train path in a single timetable. In order to present all train paths completely, we draw the train blocking times on the main tracks by using dark gray blocks, and we use light gray blocks to show the train blocking times on the siding tracks. Therefore, an overlap of the dark and light gray blocks does not indicate a train conflict, but it means that the two trains are running on different siding tracks in the same station area.

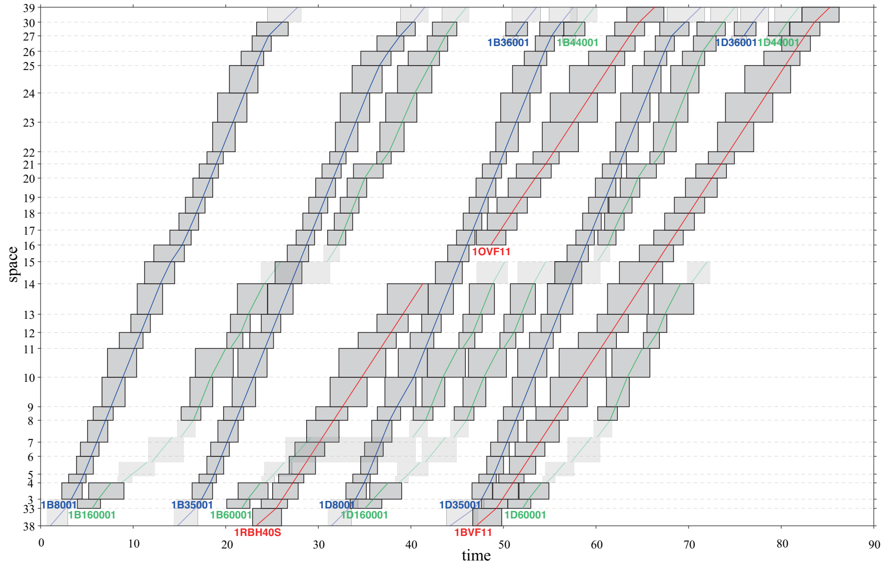
The total train delay time of the train timetables in Figures A.1(a) to A.1(c) is 3993 seconds, 3793 seconds, and 3426 seconds respectively. As we can see in the train timetables of Figures A.1(a) to A.1(c), the orders of the sprinter train 1B60001 and the intercity train 1D8001 (and the freight train 1RBH40S as well) change on some cells, e.g., cell (8, 9). As a result, in Figure A.1(b), the sprinter train 1B60001 has more delays (916 seconds), and the sum of the delays of the other affected trains (including train 1RBH40S, 1D8001, and 1OVF11) decreases by 1219 seconds; in Figure A.1(c), the delay of train 1B60001 increases by 927 seconds, and the total delay of the other affected trains decreases by 1302 seconds.



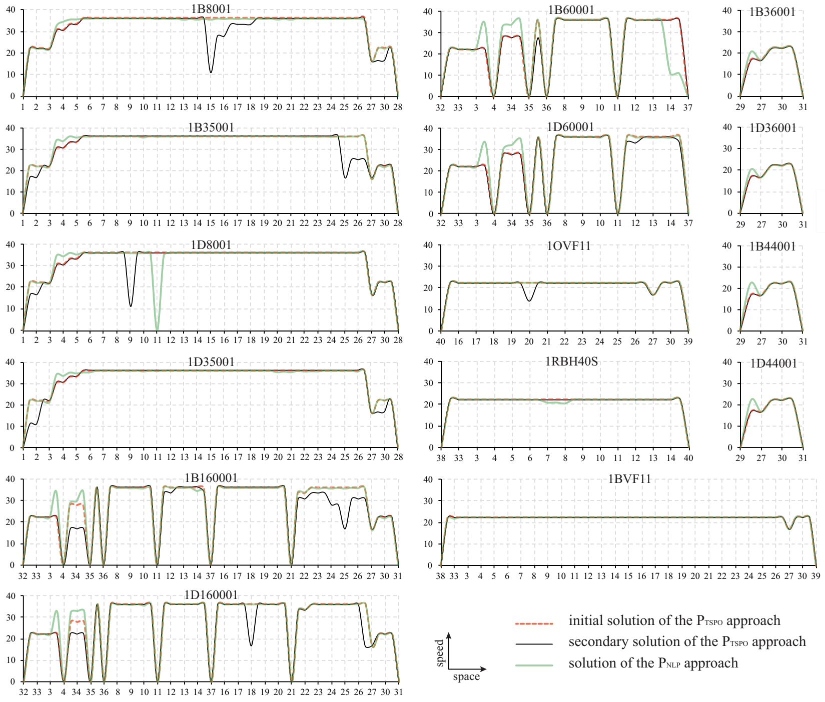
(a) Train timetable corresponding to the solution obtained by the $\mathrm{P_{NLP}}$ problem



(b) Train timetable corresponding to the initial solution of the P_{TSPO} problem



(c) Train timetable corresponding to the secondary solution of the P_{TSPO} problem



(d) Speed-space graphs for all trains, corresponding to the train timetables in (a)-(c)

Figure A.1: Train timetables and train speed-space graphs for the Dutch railway network

A.3 Case study based on the railway network from the INFORMS RAS problem solving competition 2012

A.3.1 Description of the railway network

To further assess the model performance on larger-scale instances, we adapt the railway network from the INFORMS RAS problem solving competition 2012 (INFORMS RAS, 2012), with both single-track segments and double-track segments, consisting of 67 nodes and 76 cells, as sketched in Figure A.2(a).

The train data (e.g., acceleration/deceleration rate, category, and length) and the stop pattern same to the Dutch railway network are used here; we refer to Section 5.6.1 for more information. We consider 2.5 hours of traffic with 25 trains, including 10 intercity, 10 sprinter, and 5 freight trains, and six global (bi-)directional train routes, as illustrated in Figure A.2(b). Each route has a mark in the form of (x, y, z) at its origin; the mark indicates the numbers of intercity (x), sprinter (y), and freight (z)

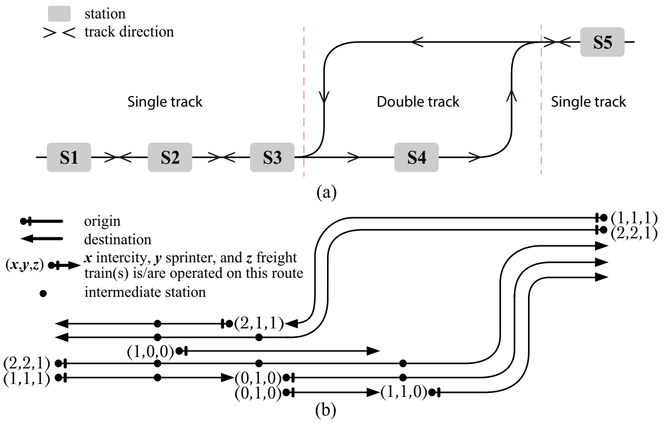


Figure A.2: A rail network adapted from INFORMS RAS (2012)

trains respectively that are operated on this route.

A.3.2 Performance of the P_{TSPO} model on a larger-scale instance

As evaluated in Section 5.6.2(1), the P_{TSPO} model yields the best performance, and the other two models already have a computation time in the experiments based on the Dutch railway network, either obtaining no feasible solution or taking a much longer computation time. Therefore, in this section, we only examine the P_{TSPO} model performance on larger-scale instances, by using the INFORMS RAS railway network described in Section A.3.1. We use the larger set of TSPOs (i.e., Set_1 in Table 5.4), due to its good solution quality, as discussed in Section 5.6.2(3). The average results of the 10 delay cases with randomly generated primary delays are illustrated in Figure A.3, including the initial solution, the secondary solutions as a function of the computation time, and the improvement in the objective value.

Similar to the results of the Dutch railway network, the initial solution is still obtained

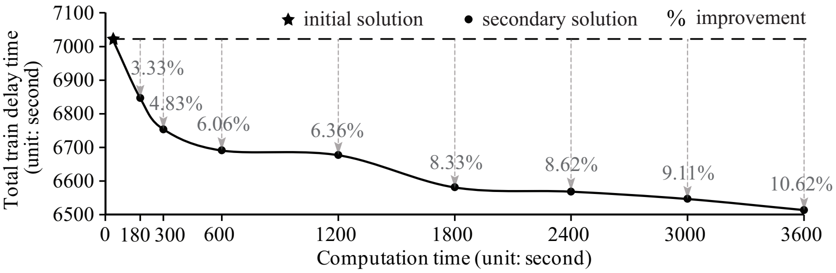
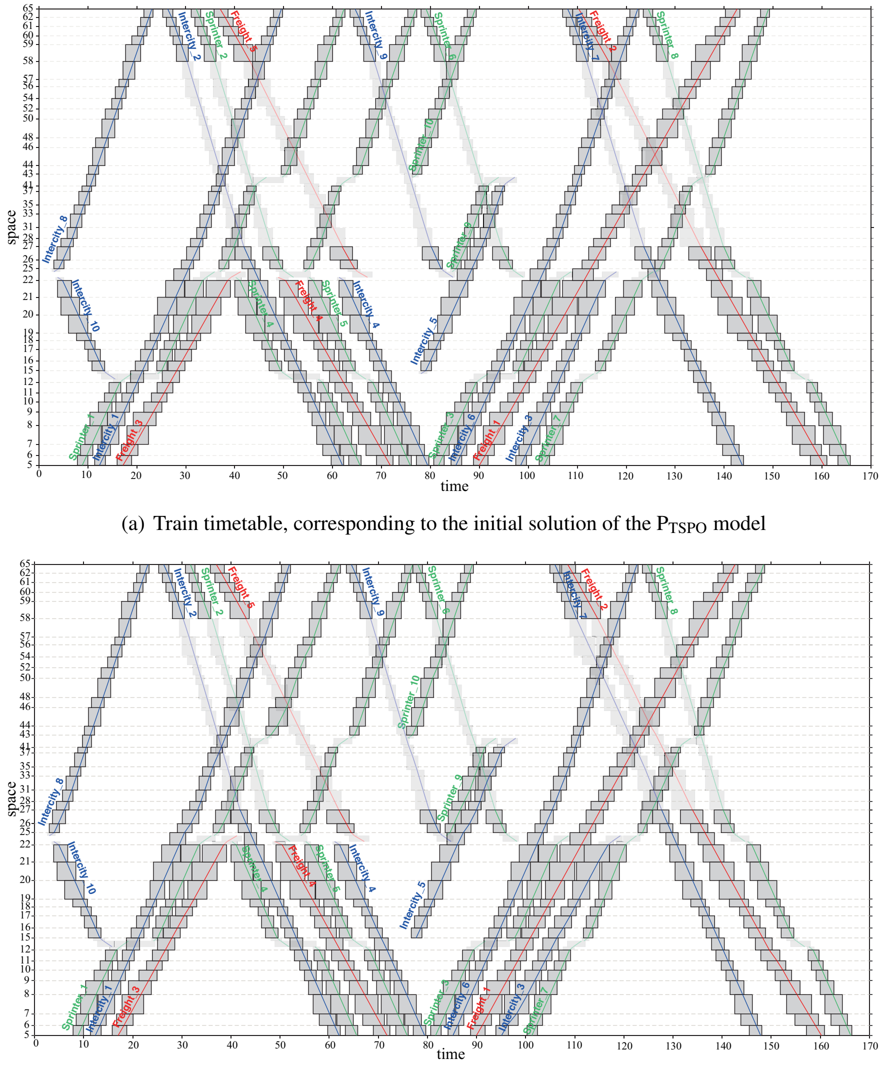


Figure A.3: Total train delay time as a function of computation time, results for the INFORMS RAS railway network

very quickly, and the total train delay time decreases as a function of the computation time in the secondary solutions. Considering multiple TSPOs achieves 3.33% improvement in the train delay time within 180 seconds, and this improvement increases to 10.62% when the computation time is extended to 3600 seconds.

Figure A.4 reports the train timetables of a representative case for the INFORMS RAS railway network, obtained by the P_{TSPO} model. An initial solution and a secondary solution are provided in Figure A.4(a) and A.4(b) respectively.



(a) Train timetable, corresponding to the initial solution of the P_{TSPO} model

(b) Train timetable, corresponding to the secondary solution of the P_{TSPO} model

Figure A.4: Train timetables for the INFORMS RAS railway network

Samenvatting

In dit proefschrift worden optimaliseringmethoden ontwikkeld voor verkeersmanagement in spoornetwerken. Het doel is om treinoperaties te verbeteren, in termen van punctualiteit, betrouwbaarheid, niet-discriminatie, benutting van de capaciteit en efficiëntie van het energiegebruik. Er wordt in het bijzonder aandacht besteed aan de volgende vier aspecten:

- Niet-discrimerende verkeersregeling

  Dit onderwerp betreft niet-discrimerende en gelijkwaardige behandeling van conflicterende wensen van concurrerende vervoerders. Er wordt een methode voorgesteld gebaseerd op gemengd-geheeltallige lineaire programmering (in het Engels: *mixed-integer linear programming, MILP*). Deze methode geeft een aanvaardbaar niveau van gelijkwaardige behandeling bij optimalisering van de vertrek- en aankomsttijden van treinen, volgordes en routes. Met behulp van experimenten en uitgebreide gevoeligheidsanalyse wordt de relatie onderzocht tussen de mate van gelijkwaardigheid en de prestatie van het systeem. Het blijkt dat minimalisering van treinvertragingen en gelijkwaardige verdeling van vertragingen conflicterende doelstellingen zijn; de gelijkwaardigheid van verkeersbewegingen kan worden verhoogd ten koste van grotere vertragingen.

- Verkeersregeling in samenhang met preventief onderhoud

  In dit onderdeel worden de mogelijkheden van de bestaande infrastructuur beter benut door een betere coördinatie van de verkeersregeling en preventief onderhoud. Hiertoe worden preventieve onderhoudsacties geformuleerd als bewegingen van virtuele treinen. Vervolgens worden in een geïntegreerde optimalisatie treinroutes, volgordes, vertrek- en aankomsttijden op tussenstations bepaald, alsook tijdvensters voor onderhoud op de betreffende segmenten en stations. Met behulp van experimenten wordt de effectiviteit van de geïntegreerde optimalisering nagegaan, en wordt aangetoond welke de voordelen zijn van simultane planning van treinbewegingen en preventief onderhoud ten opzichte van de gebruikelijke sequentiële planning.

- Integratie van verkeersregeling en treinbesturing

  In dit onderdeel wordt onderzocht welke de mogelijkheden zijn voor optimalisering van het energiegebruik in treinoperaties door integratie van strategieën

voor treinbesturing in de verkeersregeling. De details voor de verkeersregels en snelheidsprofielen worden opgenomen in één model. Er worden drie optimaliseringsmethoden ontwikkeld om tegelijk een oplossing te geven voor de inzet van treinen en voor de besturing van treinen. Het probleem wordt eerst geformuleerd als een gemengd-geheeltallig niet-lineair programmeringsprobleem (in het Engels: *mixed-integer nonlinear programming, MINLP*). Dit MINLP-probleem wordt geherformuleerd door de niet-lineaire termen te benaderen met stuksgewijs affiene functies. Dit geeft een MILP-probleem. In een voorbewerkingsstap worden mogelijke snelheidsprofielen op elk blok gegenereerd. Daarvan wordt er één gekozen door het oplossen van een MILP-probleem (de derde optimaliseringsmethode), rekening houdend met veiligheid, capaciteit en snelheidsregels. In deze optimaliseringsbenadering wordt de snelheid van de trein gezien als een variabele; de bloktijd van een trein wordt bepaald door de werkelijke snelheid. Uit de resultaten van experimenten blijkt dat de derde optimaliseringsmethode globaal gezien de beste resultaten geeft binnen de gewenste rekentijd. De resultaten tonen de voordelen van integratie, d.w.z. vertragingen kunnen worden verminderd door aanpassing van de snelheid.

- Gedistribueerde optimalisatie van verkeersregeling voor grote netwerken

  Dit deel is gericht op verbetering van de efficiëntie van de berekeningen aan het optimaliseringsprobleem met integratie van verkeersregeling en treinbesturing. Er worden drie methoden afgeleid om het probleem op te delen in deelproblemen: decompositie op basis van geografie, trein-gerelateerde decompositie en tijd-gerelateerde decomposite. Er worden drie gedistribueerde optimaliseringsbenaderingen ontwikkeld waarmee sequentieel en interactief elk deelprobleem wordt opgelost, samen met andere deelproblemen of rekening houdend met de oplossingen van andere deelproblemen. De drie beschouwde algorithmes zijn een '*alternating direction method of multipliers*' (ADMM) algorithme, een '*priority-rule-based*' (PR) algorithme en een '*cooperative distributed robust safe but knowledgeable*' (CDRSBK) algorithme. Er worden experimenten gedaan om de prestaties van de voorgestelde decompositiemethoden en algorithmes te vergelijken ten aanzien van realiseerbaarheid, rekenefficiëntie, kwaliteit van de oplossing en de geschatte afstand tot de optimale oplossing.

Summary

This thesis adopts optimization approaches to tackle the traffic management problem for railway networks, aiming at achieving better performance of railway operations, in terms of punctuality, reliability, non-discrimination, capacity utilization, and energy efficiency. Specifically, the following four aspects are considered:

- Non-discriminatory traffic control

  This topic deals with conflicting requests of competing train operators in a non-discriminatory manner by considering equity in the decision process. A mixed-integer linear programming approach is proposed, which enables us to achieve a satisfactory degree of equity while optimizing the train departure and arrival times, orders, and routes. In experiments, we study and quantify the trade-off between equity and system performance, based on an extended sensitivity analysis. We demonstrate that the minimization of train delays and delay inequity are two conflicting objectives; generally, equity of running traffic is improved at the expense of larger delays.

- Traffic control cooperating with a preventive maintenance plan

  This topic exploits the potential of existing infrastructure by better coordination between the decisions on traffic management and preventive maintenance plan. A virtual-train-based formulation method is introduced to describe preventive maintenance tasks as virtual trains in train schedules. Next, an integrated optimization approach is developed to simultaneously determine train routes, orders, departure and arrival times at passing stations, as well as preventive maintenance time slots on relevant segments and stations. In experiments, the effectiveness of the integrated optimization approach is verified, and the benefits of simultaneously scheduling trains and planning preventive maintenance tasks are demonstrated, compared with a commonly-used sequential scheduling method.

- Traffic control integrating with train control

  This topic investigates the optimization of energy efficiency in train operations by incorporating driving strategies into traffic control. The representation of microscopic traffic regulations and speed trajectories are incorporated into a single optimization problem. Three optimization approaches are developed to deliver a

train dispatching solution and a train control solution at the same time. A mixed-integer nonlinear programming approach (MINLP) is first proposed, which is then reformulated by approximating the nonlinear terms with piecewise affine functions, resulting in a mixed-integer linear programming (MILP) problem. A preprocessing method is further considered to generate the possible speed profile options for each train on each block section, one of which is further selected by a proposed MILP problem (i.e., the third optimization approach) with respect to safety, capacity, and speed consistency constraints. In these optimization approaches, the train speed is considered to be variable, and the blocking time of a train on a block section dynamically depends on its real operating speed. According to the experimental results, the third optimization approach yields the best overall performance within the required computation time. The experimental results demonstrate the benefits of the integration, i.e., train delays can be reduced by managing train speed.

- Distributed optimization of traffic control for large networks

  This topic focuses on improving the computational efficiency of the integrated optimization problem of traffic management and train control. Three decomposition methods, namely a geography-based decomposition, a train-based decomposition, and a time-interval-based decomposition, are presented to split the whole optimization problem into several subproblems. To deal with couplings among subproblems, three distributed optimization approaches are introduced to sequentially and iteratively solve each subproblem through coordination with other subproblems or with respect to the available solutions of other sub-problems. The three algorithms under consideration include an alternating direction method of multipliers (ADMM) algorithm, a priority-rule-based (PR) algorithm, and a cooperative distributed robust safe but knowledgeable (CDRSBK) algorithm. Experiments are conducted to comparatively examine the performance of the proposed decomposition methods and algorithms, in terms of feasibility, computational efficiency, solution quality, and estimated optimality.

About the author

Xiaojie Luan was born in December 12$^{\text{th}}$, 1989 in Yantai, Shandong, China. She obtained her B.Sc. degree in Traffic and Transportation at Beijing Jiaotong University in 2012. In the same year, she started her master study in Traffic and Transportation Planning and Management at Beijing Jiaotong University, under the supervision of Prof. Haiying Li and Prof. Lingyun Meng.

After receiving her M.Sc. degree in 2015, she joined the section of Transport Engineering and Logistics at Delft University of Technology as a Ph.D. candidate. Her Ph.D. project on Traffic Management Optimization of Railway Networks is funded by the Chinese Scholarship Council, under the supervision of Prof. Gabriel Lodewijks, Prof. Bart De Schutter, and Prof. Francesco Corman. During her PhD study, she participated in the Quintiqs ComBUStion Challenge, being the monthly winner of May 2017.

Her research interests include traffic management, delay management, operations research in railway transport systems.

List of publications

- Journal Paper
  - Luan, X., Wang, Y., De Schutter, B., Meng, L., Lodewijks, G., Corman, F. (2018). Integration of real-time traffic management and train control for rail networks - Part 1: Optimization problems and solution approaches. *Transportation Research Part B: Methodological*, 115, 41-71.
  - Luan, X., Wang, Y., De Schutter, B., Meng, L., Lodewijks, G., Corman, F. (2018). Integration of real-time traffic management and train control for rail networks - Part 2: Extensions towards energy-efficient train operations. *Transportation Research Part B: Methodological*, 115, 72-94.
  - Luan, X., De Schutter, B., Corman, F., Lodewijks, G. (2018). Integrating dynamic signaling commands under fixed-block signalling systems into train dispatching optimization problems. *Transportation Research Record*, 0361198118791628. No. 18-01197.
  - Luan, X., Miao, J., Meng, L., Corman, F., Lodewijks, G. (2017). Integrated optimization on train scheduling and preventive maintenance time slots planning. *Transportation Research Part C: Emerging Technologies*, 80, 329-359.

  - Luan, X., Corman, F., Meng, L. (2017). Non-discriminatory train dispatching in a rail transport market with multiple competing and collaborative train operating companies. *Transportation Research Part C: Emerging Technologies*, 80, 148-174.

- Conference Paper

  - Luan, X., De Schutter, B., van den Boom, T. J., Lodewijks, G., Corman, F. (2019). Distributed optimization approaches for the integrated problem of real-time railway traffic management and train control. In *Proceedings of the 8th International Conference on Railway Operations Modelling and Analysis*, Norrköping, Sweden, June 17-20, 2019.
  - Luan, X., De Schutter, B., van den Boom, T. J., Corman, F., Lodewijks, G. (June, 2018). Distributed optimization for real-time railway traffic management. In *Proceedings of the 15th IFAC Symposium on Control in Transportation Systems (CTS 2018)*, Savona, Italy, June 6-8, 2018. pp. 106-111.
  - Luan, X., De Schutter, B., Corman, F., Lodewijks, G. (2018). Integrating dynamic signalling commands under fixed-block signalling systems into train dispatching optimization models. In *Proceedings of the 97th Annual Meeting of the Transportation Research Board*, Washington, DC, January 7-11, 2018. Paper 18-01197.
  - Luan, X., Corman, F., Wang, Y., De Schutter, B., Meng, L., Lodewijks, G. (2017). Integrated Optimization of Traffic Management and Train Control for Rail Networks. In *Proceedings of the 7th International seminar on Railway Operations Modeling and Analysis*, Lille, France, April 5-7, 2017. **(Selected as one of the 10 best papers)**

TRAIL Thesis Series

The following list contains the most recent dissertations in the TRAIL Thesis Series. For a complete overview of more than 250 titles see the TRAIL website: www.rsTRAIL.nl.

The TRAIL Thesis Series is a series of the Netherlands TRAIL Research School on transport, infrastructure and logistics.

Luan, X., *Traffic Management Optimization of Railway Networks*, T2019/10, July 2019, TRAIL Thesis Series, the Netherlands

Hu, Q., *Container Transport inside the Port Area and to the Hinterland*, T2019/9, July 2019, TRAIL Thesis Series, the Netherlands

Andani, I.G.A., *Toll Roads in Indonesia: transport system, accessibility*, spatial and equity impacts, T2019/8, June 2019, TRAIL Thesis Series, the Netherlands

Ma, W., *Sustainability of Deep Sea Mining Transport Plans*, T2019/7, June 2019, TRAIL Thesis Series, the Netherlands

Alemi, A., *Railway Wheel Defect Identification*, T2019/6, January 2019, TRAIL Thesis Series, the Netherlands

Liao, F., Consumers, *Business Models and Electric Vehicles*, T2019/5, May 2019, TRAIL Thesis Series, the Netherlands

Tamminga, G., *A Novel Design of the Transport Infrastructure for Traffic Simulation Models*, T2019/4, March 2019, TRAIL Thesis Series, the Netherlands

Lin, X., *Controlled Perishable Goods Logistics: Real-time coordination for fresher products*, T2019/3, January 2019, TRAIL Thesis Series, the Netherlands

Dafnomilis, I., *Green Bulk Terminals: A strategic level approach to solid biomass terminal design*, T2019/2, January 2019, TRAIL Thesis Series, the Netherlands

Feng, Fan, *Information Integration and Intelligent Control of Port Logistics System*, T2019/1, January 2019, TRAIL Thesis Series, the Netherlands

Beinum, A.S. van, *Turbulence in Traffic at Motorway Ramps and its Impact on Traffic Operations and Safety*, T2018/12, December 2018, TRAIL Thesis Series, the Netherlands

Bellsolà Olba, X., *Assessment of Capacity and Risk: A Framework for Vessel Traffic in Ports*, T2018/11, December 2018, TRAIL Thesis Series, the Netherlands

Knapper, A.S., *The Effects of using Mobile Phones and Navigation Systems during Driving*, T2018/10, December 2018, TRAIL Thesis Series, the Netherlands

Varotto, S.F., *Driver Behaviour during Control Transitions between Adaptive Cruise Control and Manual Driving: empirics and models*, T2018/9, December 2018, TRAIL Thesis Series, the Netherlands

Stelling-Kończak, A., *Cycling Safe and Sound*, T2018/8, November 2018, TRAIL Thesis Series, the Netherlands

Essen, van M.A., *The Potential of Social Routing Advice*, T2018/7, October 2018, TRAIL Thesis Series, the Netherlands

Su, Zhou, *Maintenance Optimization for Railway Infrastructure Networks*, T2018/6, September 2018, TRAIL Thesis Series, the Netherlands

Cai, J., *Residual Ultimate Strength of Seamless Metallic Pipelines with Structural Damage*, T2018/5, September 2018, TRAIL Thesis Series, the Netherlands

Ghaemi, N., *Short-turning Trains during Full Blockages in Railway Disruption Management*, T2018/4, July 2018, TRAIL Thesis Series, the Netherlands

Gun, van der J.P.T., *Multimodal Transportation Simulation for Emergencies using the Link Transmission Model*, T2018/3, May 2018, TRAIL Thesis Series, the Netherlands

Van Riessen, B., *Optimal Transportation Plans and Portfolios for Synchromodal Container Networks*, T2018/2, March 2018, TRAIL Thesis Series, the Netherlands

Saeedi, H., *Network-Level Analysis of the Market and Performance of Intermodal Freight Transport*, T2018/1, March 2018, TRAIL Thesis Series, the Netherlands

Ypsilantis, P., *The Design, Planning and Execution of Sustainable Intermodal Port-hinterland Transport Networks*, T2017/14, December 2017, TRAIL Thesis Series, the Netherlands

Han, Y, *Fast Model Predictive Control Approaches for Road Traffic Control*, T2017/13, December 2017, TRAIL Thesis Series, the Netherlands

Wang, P., *Train Trajectory Optimization Methods for Energy-Efficient Railway Operations*, T2017/12, December 2017, TRAIL Thesis Series, the Netherlands

Weg, G.S. van de, *Efficient Algorithms for Network-wide Road Traffic Control*, T2017/11, October 2017, TRAIL Thesis Series, the Netherlands

He, D., *Energy Saving for Belt Conveyors by Speed Control*, T2017/10, July 2017, TRAIL Thesis Series, the Netherlands

Bešinović, N., *Integrated Capacity Assessment and Timetabling Models for Dense Railway Networks*, T2017/9, July 2017, TRAIL Thesis Series, the Netherlands

Chen, G., *Surface Wear Reduction of Bulk Solids Handling Equipment Using Bionic Design*, T2017/8, June 2017, TRAIL Thesis Series, the Netherlands