

Optimal Trajectory Planning and Train Scheduling for Railway Systems

Yihui Wang



Optimal Trajectory Planning and Train Scheduling for Railway Systems

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op maandag 3 november 2014 om 12:30 uur
door

Yihui WANG

Bachelor of Science in Automation, Beijing Jiaotong University,
geboren te Huize, Yunnan, China.

Dit proefschrift is goedgekeurd door de promotoren:

Prof. dr. ir. B. De Schutter

Copromotor: Dr. ir. T.J.J. van den Boom

Samenstelling promotiecommissie:

Rector Magnificus

Prof. dr. ir. B. De Schutter

Dr. ir. T.J.J. van den Boom

Prof. dr. B. Ning

Prof. dr. ing. I.A. Hansen

Prof. dr. ir. C. Vuik

Prof. dr. ir. D. Pacciarelli

Prof. dr. B.F. Heidergott

Prof. dr. R. Babuška

voorzitter

Technische Universiteit Delft, promotor

Technische Universiteit Delft, copromotor

Beijing Jiaotong University

Technische Universiteit Delft

Technische Universiteit Delft

Università degli Studi Roma Tre

Vrije Universiteit Amsterdam

Technische Universiteit Delft (reservelid)

This thesis has been completed in partial fulfillment of the requirements of the Dutch Institute of Systems and Control (DISC) for graduate studies. The support of the Chinese Scholarship Council (CSC) is greatly acknowledged, as well as the support of the European Union 7th Framework Network of Excellence “Highly-complex and networked control systems (HYCON2)” program.

TRAIL Thesis Series T2014/7, The Netherlands TRAIL Research School

P.O. Box 5017

2600 GA Delft, The Netherlands

T: +31 (0) 15 278 6046

T: +31 (0) 15 278 4333

E: info@rstrail.nl

Published and distributed by: Yihui Wang

E-mail: yhwang1122@gmail.com

ISBN 978-90-5584-176-9

Keywords: trajectory planning, train scheduling, passenger demand, urban rail transit, optimization.

Copyright © 2014 by Yihui Wang

All rights reserved. No part of the material protected by this copyright notice may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage and retrieval system, without written permission of the author.

Printed in the Netherlands, Haveka B.V.

Acknowledgments

It has been a wonderful experience to stay in Delft for four years and to finish my thesis. I feel so blessed since many people have helped me, supported me, accompanied me, and shared the happiness and the sadness together with me. Here, I wish to express my gratitude to all of these people.

I would like to sincerely thank my main supervisor, Prof. Bart De Schutter, for his thoughtful guidance, great inspiration, and warm support throughout this study. His enthusiasm for research and teaching, kindness and care to students, and wide range of knowledge encouraged me to pursue an academic career. Thank you for the nice discussions we had in these four years and your efficient and precise comments on equations and English writing. I especially appreciated the confidence that you have in me.

I would also like to thank my second thesis supervisor Dr. Ton van den Boom for his insights on railways and control theories. His ideas, questions, and comments were important to ensure the research went in the right direction. I appreciated the nice discussions about daily life, cultural differences, technology, etc.

My wholehearted gratitude goes to Prof. Bin Ning for being my Ph.D. supervisor and bringing me into the academic world. I have appreciated you for giving me the opportunity to study abroad and to pursue my Ph.D. degree in Delft. Thank you for the freedom, the trust, and the unconditional support you have given to me.

I am also grateful to Prof. Tao Tang for all the valuable discussions we had and for all the opportunities he provided me to participate in the projects of train control systems. Further, I greatly appreciate Prof. Robert Babuška, Prof. Ingo Hansen, Prof. Bernd Heidergott, Prof. Dario Pacciarelli, and Prof. Kees Vuik for being part of my defense committee. Thank you all for the nice discussions and valuable comments.

During these four years, I enjoyed working with my colleagues at DCSC. A special thanks goes to my office mate Noortje for her company and for translating my summary and propositions into Dutch. I am thankful to Anil, Anna, Anqi, Arne, Bart, Dieky, Jia, Juan, Jun, Kim, Le, Mernout, Mohammad, Patricio, Pieter, Renshi, Sadegh, Shu, Shuai, Subramanya, Yashar, Yiming, Yu, Yue, Yuping, Zhao, Zhe, Zulkifli, and the colleagues from other departments and other universities for the good memories of the conferences, special events, discussions, lunches, dinners, and games. I appreciated Hans Hellendoorn for his help and kindness. Here I would also like to thank those who work in our secretariat (Kitty, Esther, Saskia, Marieke, and Heleen) for being kind and helpful.

Further, I am grateful to the railway people in TU Delft for nice presentations and discussions in our monthly meetings: Bart, Daniel, Dewei, Egidio, Fei, Francesco, Ingo, Jeroen, Lingyun, Nadjla, Nikola, Pavle, Rob, and Ton. I would also like to thank my colleagues in

the State Key Laboratory of Rail Traffic Control and Safety, especially to Fang Cao, Lijie Chen, Chunhai Gao, Qing Gu, Youneng Huang, Bo Liu, Jiang Liu, Chao Liu, Lianchuan Ma, Ru Niu, Shuai Su, Wumei Tang, Cheng Wang, Hongwei Wang, Jing Xun, Lei Yuan, Zhenyu Yu, Bobo Zhao, Xianqiong Zhao, and Li Zhu.

I would like to thank my friends from the church and fellowship, especially to Chang, David & Fera, Didy & Fera, Fei, Gerrit, Guojie, Hanning, Henk & Zhuna, Jizhe, Joanne, Liang, Lisa, Lu, Michael & Gracelyn, Philip, Priscilla, Qianrong, Quanquan, Shasha, Weishan, Xiucheng, Yanchun, Zhaoyuan for the wonderful times we had together. Special thanks to Yanchun for helping me drawing the picture on the thesis cover. I would also like to thank my friends Linfeng Chen, Jianbin Fang, Ping Liu, Yan Ni, Meng Ma, Shuhong Tan, Chang Wang, Jianbin Xin, Yong Zhang, and Lilan Zhou for being a nice fun community in Delft.

I would like to thank to my parents and my parents-in-law for their unconditional love, support, and encouragement. Thank you for giving me strength and freedom to chase my dreams. My sister, Xiaoyan Wang, and my brother-in-law, Gang Xu, deserve my whole-hearted thanks as well. Last but not least, I would like to congratulate my husband Jian Xu on getting his Ph.D. degree a few months before me. I am very lucky to have met you. I thank you for always being there for me, even though we have been almost eight thousand kilometers away from each other for four years. Your love, full understanding, and wise encouragements helped me go through tough moments and brought me peace and joy in these years.

Delft, October 2014

Yihui Wang

Contents

Acknowledgments	v
1 Introduction	1
1.1 A brief introduction on railway operations	1
1.2 Motivation and aim of the thesis	3
1.3 Scope and contributions of the thesis	4
1.4 Thesis outline	7
2 Background: Train Operations and Scheduling	9
2.1 Operation of trains	9
2.1.1 Automatic train operation	9
2.1.2 Principles of signaling systems	11
2.2 Optimal trajectory planning of trains	13
2.2.1 Optimal trajectory planning of a single train	13
2.2.2 Optimal trajectory planning of multiple trains	15
2.3 Urban rail transit scheduling process	15
2.3.1 Passenger demand	16
2.3.2 Train scheduling	17
2.4 Summary	19
3 Optimal Trajectory Planning for a Single Train	21
3.1 Introduction	21
3.2 Model formulation	22
3.2.1 Train model	22
3.2.2 An assumption about the line resistance	24
3.3 Mathematical formulation of the single train trajectory planning problem . .	24
3.4 Solution approaches	27
3.4.1 Pseudospectral method	27
3.4.2 Mixed integer linear programming	29
3.5 Case study	36
3.5.1 Set-up	36
3.5.2 Results and discussion	39
3.6 Conclusions	44
3.A A general formulation of the pseudospectral method	45
3.A.1 The multiple-phase optimal control problem	45
3.A.2 The solution process of the optimal control problem	46

4	Optimal Trajectory Planning for Multiple Trains	49
4.1	Introduction	49
4.2	Model formulation	50
4.2.1	Train dynamics	50
4.2.2	Operation of trains in a fixed block signaling system	50
4.2.3	Operation of trains in a moving block signaling system	53
4.3	Mathematical formulation of the multiple trains trajectory planning problem	54
4.4	Solution approaches	55
4.4.1	Greedy approach	55
4.4.2	Simultaneous approach	56
4.5	Mixed logical dynamic formulation for signaling system constraints	56
4.5.1	Multiple trains under fixed block signaling system	57
4.5.2	Multiple trains under moving block signaling system	59
4.5.3	Extension: mode vector constraints	61
4.6	Case study	62
4.6.1	Set-up	62
4.6.2	Results for the fixed block signaling system	64
4.6.3	Results for the moving block signaling system	68
4.6.4	Discussion	68
4.7	Conclusions	71
5	OD-Independent Train Scheduling for an Urban Rail Transit Line	73
5.1	Introduction	73
5.2	Model formulation	74
5.2.1	Arrivals and departures	75
5.2.2	Passenger demand characteristics	77
5.2.3	Passenger/vehicle interaction	78
5.3	Mathematical formulation of the train scheduling problem	78
5.4	Solution approaches	80
5.4.1	Gradient-free nonlinear programming	80
5.4.2	Gradient-based nonlinear programming	81
5.4.3	Mixed integer nonlinear programming	81
5.4.4	Mixed integer linear programming	82
5.4.5	A new approach: iterative convex programming	83
5.5	Extension: stop-skipping at small stations	84
5.6	Case Study	86
5.6.1	Set-up	86
5.6.2	Results and discussion	89
5.7	Conclusions	96
6	OD-Dependent Train Scheduling for an Urban Rail Transit Line	97
6.1	Introduction	97
6.2	Model formulation	98
6.2.1	Arrivals and departures with stop-skipping	100
6.2.2	OD-dependent passenger demand characteristics	101
6.3	Mathematical formulation of the scheduling problem	104

6.4	Solution approaches	105
6.4.1	Bi-level optimization approach	106
6.4.2	Efficient bi-level optimization approach	106
6.5	Case study	108
6.5.1	Set-up	108
6.5.2	Results and discussion	109
6.6	Conclusions	118
7	OD-Dependent Train Scheduling for an Urban Rail Transit Network	121
7.1	Introduction	121
7.2	Model formulation	122
7.2.1	Three types of events	122
7.2.2	Event-driven dynamics	128
7.3	Mathematical formulation for the scheduling problem	133
7.3.1	Performance criteria	133
7.3.2	Constraints	134
7.4	Rolling horizon approach and initial conditions	135
7.5	Solution approaches	136
7.6	Case study	136
7.6.1	Set-up	136
7.6.2	Results and discussion	138
7.7	Conclusions	146
8	Conclusions and Future Research	147
8.1	Conclusions	147
8.2	Recommendations for future research	149
8.2.1	Optimal trajectory planning and train scheduling	149
8.2.2	Additional directions for future research	152
	Bibliography	155
	Symbols and Abbreviations	167
	TRAIL Thesis Series Publications	171
	Samenvatting	173
	Summary	175
	Curriculum Vitae	177

Chapter 1

Introduction

Rail traffic plays a key role in public transportation since it combines high transport capacity and high efficiency. More specifically, a safe, fast, punctual, energy-efficient, and comfortable railway system is important for the economic, environmental, and social objectives of a country or a city. The main focus of this dissertation is on saving energy in railway operations and on enhancing the passenger satisfaction, which can be achieved via optimal trajectory planning for trains and the train scheduling according to passenger demands.

In this chapter we first provide a brief introduction to railway operations and then present the motivation for the research addressed in this thesis. We conclude this introductory chapter with a list of our contributions and the structure of this thesis.

1.1 A brief introduction on railway operations

A railway system consists of three essential elements: infrastructure (like tracks, stations, signaling equipment, etc.), rolling stock with locomotives and cars or electric multiple units (EMUs), and the operation rules and procedures for a safe and efficient operation [98]. The design and construction of the infrastructure and rolling stock are affected by the operation rules; thus the performance of the railway system is also impacted by the operation rules. Passenger railway systems could be classified into interurban railway systems (or standard railway systems) and urban rail transit systems (such as metros and subways). Rail infrastructure is a limited resource in interurban rail transit systems, where lines overlap or cross with each other and trains usually overtake or meet each other. On the other hand, in urban rail transit systems, the lines are separated from each other and each direction of the line has a dedicated infrastructure. Moreover, in principle trains do not overtake and meet each other in urban rail transit systems.

The optimal trajectory planning (i.e., speed profile calculation) methods for the operation of trains proposed in this dissertation can be applied both for interurban railway systems and urban rail transit systems. However, train scheduling approaches we present here are focused on urban rail transit systems.

In railway systems, the operation of trains is in general controlled through a hierarchical control framework with five levels, i.e., scheduling, real-time (re)scheduling, remote traffic control, interlocking and signaling, and train and infrastructure control (see Figure 1.1) [89].

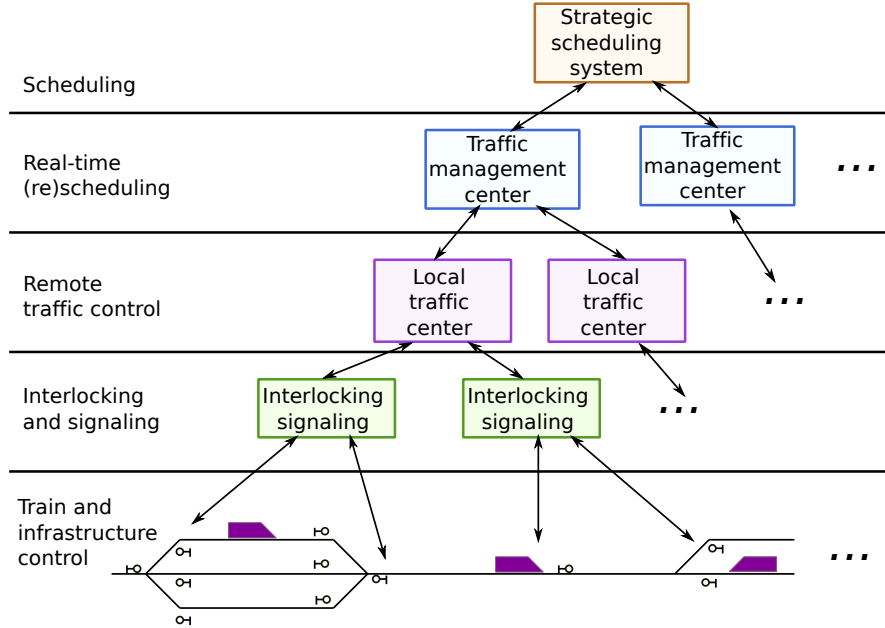


Figure 1.1: Hierarchical structure of the railway operations (adopted from [89])

The scheduling process for railway systems involves a complex procedure that includes demand analysis, line planning, train scheduling, rolling stock planning, and crew scheduling (see Section 2.3 for more detailed information). When delays, interruptions, or failures, etc. occur, dispatchers in the traffic management centers supervise the railway network and they resolve the conflicts through rescheduling. The decisions of the dispatchers are then forwarded to the remote traffic control level, where the local traffic centers set the routes and issue the speed limits for trains through interlocking systems and signaling systems. Moreover, there also exists an opposite information flow: the information of train operations is gathered automatically at the train control level and forwarded upwards to the traffic management systems.

Safety is an important issue for railway systems, where signaling systems and interlocking systems are employed to prevent collisions between trains and ensure safe train movements. There are two principles for signaling systems, viz. fixed block signaling principle and moving block signaling principle (for more information see Section 2.1.2). In practice, advanced signaling systems for train control and safety have been developed, such as the European train control system for interurban railway systems [98] and communication-based train control systems for urban rail transit systems [67]. More specifically, the European train control system has three levels, where level 1 and 2 are based on the fixed block signaling principle and level 3 is based on the moving block signaling principle. For modern urban rail transit systems, the moving block signaling principle is often implemented in the communication-based train control system and the fixed block signaling system is also implemented as a back-up. The architecture of the communication-based train control system may involve automatic train protection (ATP), automatic train operation (ATO), and automatic train supervision (ATS). In particular, the ATP system is used to guarantee the safety

of the operation of trains, where deviating driving behaviors (like over-speeding or passing red signals) caused by drivers or ATO systems would trigger an emergency brake. Furthermore, the ATO system can control the traction and braking effort automatically to operate trains between stations (see Section 2.1.1 for more detailed information). In addition, there are also some driver assistance systems that have been developed in practice to assist drivers to drive the train optimally, such as FreightMiser [63], Metromiser [63], and driving style manager [39].

Train scheduling is crucial for railway operations since it has a significant impact on infrastructure usage and punctuality. For interurban railway systems, the timetables are usually published to provide trip information to passengers. The drawback of fixed timetables is that adjustments (due to e.g., changes of demand) are difficult to implement. However, in urban rail transit systems, trains are operated with high frequencies and rail transport operators often do not publish the train schedule to passengers but only provide some real-time information, such as that a train will arrive within 2 minutes. Furthermore, the passenger demands for urban rail transit systems may change rapidly with the time of the day or due to some special events. Hence, rail transit operators could schedule trains in real time based on the passenger demands to increase passenger satisfaction with short and reliable travel times.

1.2 Motivation and aim of the thesis

Due to the increasing energy prices and environmental concerns, energy efficiency in transportation systems is becoming more and more important [57]. For the Beijing urban rail transit system, the power consumption in 2008 was 6.5 million kWh, which was 1% of the total power consumption of the city [55]. It is predicted that the power consumption of Beijing urban rail transits system will rise to 13.9 million kWh in 2015, which would then account for 1.2% of the total power consumption [55]. Furthermore, the energy consumption for the operation of trains is about 40-50% of the total power consumption for urban rail transit systems [138]. Therefore, the reduction of energy consumption for the operation of trains is one of the key objectives in the operation of railway systems. Meanwhile, the interest of railway operators in energy efficiency has been rising more and more in recent years, as even a small improvement in the energy consumption can make the railway operators save a lot of money.

Furthermore, with the dramatical increase of passenger demand in large cities like Beijing, Shanghai, Tokyo, New York, and Paris, urban rail transit system plays an increasing role for the efficiency and sustainability for the overall transportation system. Nowadays the operation of trains in urban rail transit systems is characterized by a high frequency, where the minimum headway between two successive trains is usually 2 to 5 minutes, which could even be reduced to 90 s with the development of advanced train control systems [115]. When trains are operated with such a high frequency, the scheduling of trains based on the passenger demand becomes more and more important for passenger satisfaction and for the reduction of operation costs. The passenger satisfaction depends on the waiting times, in-vehicle times, and the number of transfers, while the operation costs are determined by the number of train services and the energy consumption of train operations.

The aim of the thesis is to determine and develop mathematical models and solution

approaches to shorten the travel time of passengers and to reduce energy consumption in railway systems. With respect to the five-level hierarchical control framework discussed above (cf. Figure 1.1), scheduling, real-time (re)scheduling, and train control are closely related to the energy consumption of trains and the travel time of passengers, while the remote traffic control and the signaling and interlocking systems are responsible for the safety of the operation of trains. In the train control level, trains are operated by drivers or ATO systems under the supervision of ATP systems. In addition, trains are assumed to run according to the schedule given by the train scheduling or real-time (re)scheduling, i.e., the fixed running times for trains between two consecutive stations and fixed dwell time at stations. Generally, the scheduling of trains means to generate an off-line timetable for the railway system. Based on the existing timetable data, real-time rescheduling is used to handle route conflicts due to train delays or incidents. In this thesis, real-time scheduling means that there is no existing timetable or constant headways, but the schedule of trains is optimized in a rolling horizon way taking passenger demands and operation costs into consideration. The travel time of passengers is considered in the train scheduling, where the running times and dwell times of trains are determined. Note that the energy consumption depends on the running times of trains, so the energy saving can also be dealt with in the train scheduling layer. In this thesis, the following two topics are investigated:

- **Trajectory planning.** A nonlinear model for the operation of trains is derived and several approaches are presented to calculate the optimal trajectories in an energy-efficient way for trains based on a given train schedule.
- **Train scheduling.** To shorten the travel time of passengers and to reduce the energy consumption, we develop a train scheduling model for urban rail transit systems and optimization approaches to find a balanced trade-off between total passenger travel time and the operation cost of the rail transport operator.

1.3 Scope and contributions of the thesis

The main contributions of the work presented in this dissertation are as follows:

- We develop a new iterative convex programming (ICP) approach to solve the train scheduling problem for an urban rail transit line. Constant origin-destination-independent (OD-independent) passenger demands within the scheduling period is taken into account in the train scheduling problem.
- We include constant origin-destination-dependent (OD-dependent) passenger demands within the time period considered in the train scheduling model and we propose an efficient bi-level approach to solve the problem more efficiently.
- We introduce an event-driven model, which can take time-varying OD-dependent passenger demand, the splitting of passenger flows, and passenger transfer behavior at transfer stations into account for the train scheduling of urban rail transit network. This event-driven model consists of three types of events, viz. departure events, arrival events, and passenger arrival rate change events at platforms.

The contents and contributions can be found in Chapters 3-7 and can be summarized as follows:

Chapter 3: The optimal trajectory planning problem for the operation of a single train is considered. The varying line resistance, variable speed restrictions, and varying maximum traction force are included in the problem definition. The objective function is a trade-off between the energy consumption and the riding comfort. Two approaches are proposed to solve this optimal control problem:

- A pseudospectral method, i.e. a state-of-the-art method for optimal control, is applied for the first time in optimal train control, where the optimal trajectory planning problem is recast into a multiple-phase optimal control problem.
- The optimal trajectory planning problem is reformulated as a mixed-integer linear programming (MILP) problem by approximating the nonlinear terms in the problem by piecewise affine functions. The resulting MILP problem can be solved efficiently by existing solvers.

The main conclusion of the chapter is that if the available time for computing the solution is large enough, the pseudospectral method is recommended since it has a higher performance. Otherwise, the MILP approach can be applied to obtain the optimal train trajectory.

The contents of Chapter 3 can be mainly found in [130] and have been partially presented in [127, 128, 130].

Chapter 4: We investigate the optimal trajectory planning problem for multiple trains under fixed block signaling systems and moving block signaling systems. The constraints caused by the leading train in a fixed or moving block signaling system are included into the model for the operation of trains. Two solution approaches are proposed to solve the resulting optimal control problem for multiple trains:

- The greedy approach optimizes the trajectory of the leading train first and then based on the optimal trajectory of the leading train, the trajectory planning problem for the following train is solved. The MILP approach and the pseudospectral method are employed to solve the optimal trajectory for the leading train and the following train.
- The simultaneous approach optimizes the trajectory of all the trains in the problem formulation at the same time, where the MILP approach and the pseudospectral method are also applied.

In the simulation experiment, we show that simultaneous approach yields a better performance but requires a higher computation time compared with the greedy approach. The contents of Chapter 4 can be mainly found in [134] and have been partially presented in [129, 131].

Chapter 5: The real-time train scheduling problem for an urban rail transit line is investigated with the aim of minimizing the total travel time of passengers and the energy consumption of the operation of trains. Based on the OD-independent passenger demand of the urban rail transit system, the optimal departure times, running

times, and dwell times are obtained by solving the scheduling problem. Several approaches are proposed to solve this problem:

- A new iterative convex programming (ICP) approach is proposed to solve the train scheduling problem.
- Nonlinear programming approaches (sequential quadratic programming (SQP) and pattern search in particular), a mixed integer nonlinear programming approach, and an MILP approach are also applied to solve the problem.

We find that the ICP approach provides the best trade-off between performance and computational complexity comparing with other alternative solution approaches.

The contents of Chapter 5 can be mainly found in [135] and have been partially presented in [132, 133].

Chapter 6: We consider the train scheduling taking constant OD-dependent passenger demands into account for an urban rail transit line. A stop-skipping strategy is adopted to reduce the passenger travel time and the energy consumption. The train scheduling problem results in a mixed integer nonlinear programming problem, where two solution approaches are proposed:

- A bi-level approach is considered to solve the train scheduling problem, where the higher level optimizes the binary variables and the lower level solves a nonlinear nonconvex problem for each combination of binary variables.
- We also propose an efficient bi-level approach that first applies a threshold method to obtain a good initial solution for the problem and then limits the search space of the variables to speed up the optimization process.

Experiment indicates that the bi-level approach has better control performance, but the efficient bi-level approach can provide an acceptable solution with much less computation time.

The contents of Chapter 6 can be mainly found in [136] and have been partially presented in [137].

Chapter 7: An event-driven model is proposed for the train scheduling of an urban rail transit network, where a time-varying OD-dependent passenger demand, splitting of passenger flows, and passenger transfer behavior at transfer stations are included. This event-driven model involves three types of events, i.e., departure events, arrival events, and passenger arrival rate change events at platforms. The train scheduling problem that is constructed based on the event-driven model, is a real-valued nonlinear nonconvex programming problem. Several solution approaches, such as SQP, pattern search method, mixed integer linear programming, and genetic algorithms, can be applied to this train scheduling problem.

The simulation results¹ of the case study show that the SQP method provides a better trade-off between control performance and computational complexity than the genetic algorithm.

¹The pattern search and the MILP approach are not considered in the case study.

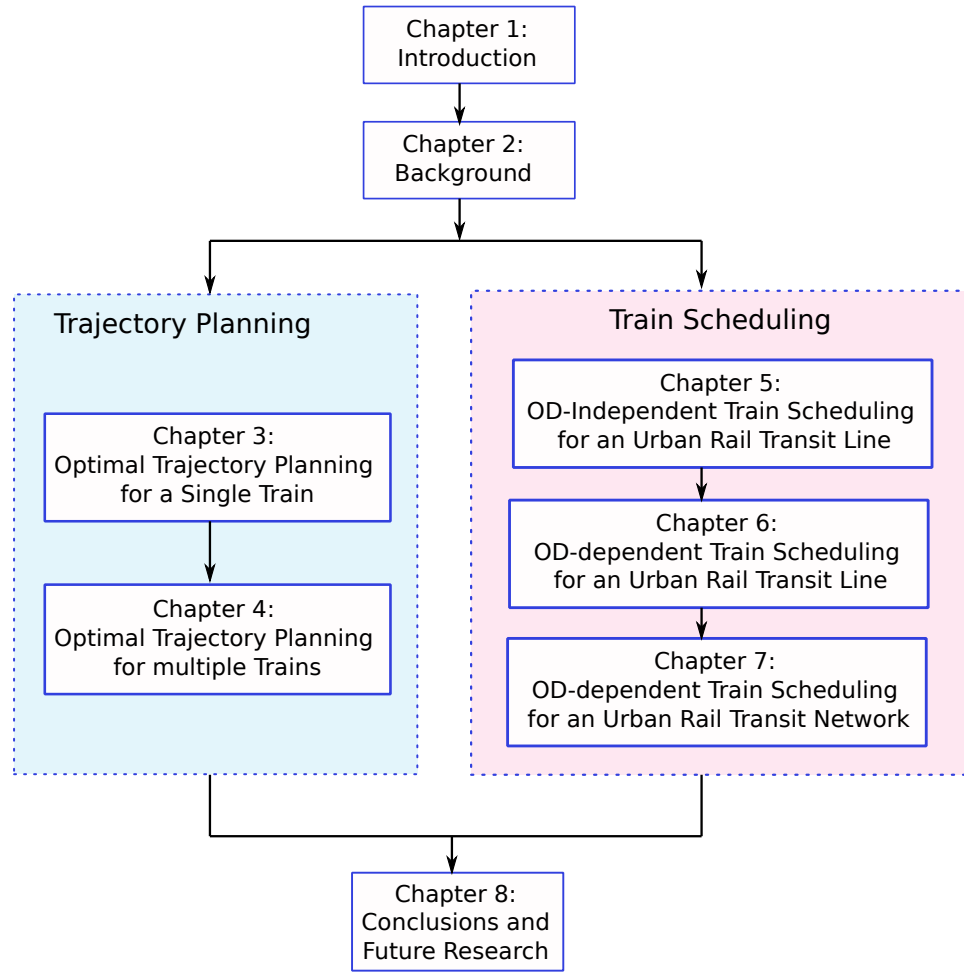


Figure 1.2: The road map of this dissertation

1.4 Thesis outline

A road map of the dissertation is presented in Figure 1.2, which clarifies the connections between the chapters. According to the road map, readers interested in optimal trajectory planning could read the dissertation in the following order: Chapter 1, Section 2.1, Section 2.1.2, and Section 2.4 of Chapter 2, Chapter 3, Chapter 4, and Chapter 8. Those interested in the train scheduling problem for urban rail transit system could read the dissertation in the following order: Chapter 1, Section 2.3 and Section 2.4 of Chapter 2, Chapter 5, Chapter 6, Chapter 7, and Chapter 8.

Chapter 1 gives the motivation and a general introduction to the topic of the thesis. Chapter 2 presents the background of the operation of trains and the train scheduling process and summarizes the related research work in the literature. Chapter 3 and Chapter 4 investigate on optimal trajectory planning for a single train and for multiple trains, respec-

tively. Chapter 5 and Chapter 6 focus on train scheduling for an urban rail transit line, where the passenger characteristics are described in different ways. The train scheduling problem for urban rail transit networks is formulated in Chapter 7, where the passenger transfers are included. Chapter 8 concludes the thesis with the main contributions and directions for future research.

Chapter 2

Background: Train Operations and Scheduling

In this chapter, background material and literature review on the operation of trains and on urban rail train scheduling will be presented. In Section 2.1, the operation of trains is introduced, where the automatic train operation (ATO) system is explained in detail. In addition, a brief introduction to fixed block signaling systems and moving block signaling systems are also given. An overview of optimal control approaches for the trajectory planning of a single train and multiple trains is provided in Section 2.2. In Section 2.3, the urban rail transit scheduling problem is introduced. This chapter concludes with a short summary in Section 2.4.

2.1 Operation of trains

Nowadays, several dedicated high-speed railway lines and urban rail transit systems with short headways are operated with a high degree of automation [57]. This requires advanced train control systems to fulfill safety and operational requirements, such as the European train control system and communication-based train control systems, which include equipment on board of trains as well as in control centers [93]. Advanced train control systems enable the energy-efficient driving of trains, which becomes more and more important because of the rising energy prices and environmental concerns [87].

The ATO system of an advanced train control system drives the train according to a pre-defined train trajectory (i.e., a speed profile) [100] to ensure punctuality and energy saving. In addition, signaling systems in train control systems are important for running safety of trains. In this section, we first give a brief introduction to ATO systems and then provide a short introduction to the principle of signaling systems.

2.1.1 Automatic train operation

With the development of modern railway systems, automatic train control systems have become vital equipment that ensures the running safety, shortens the train headways, and

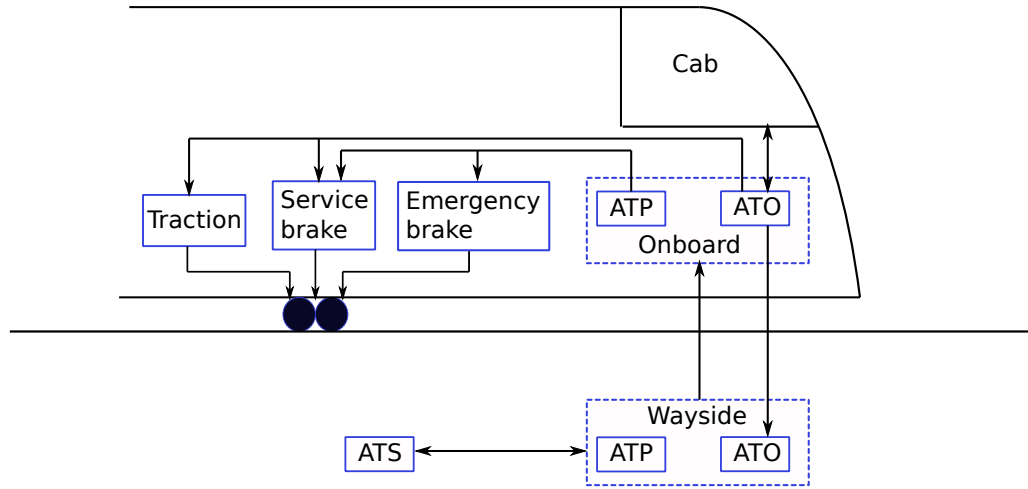


Figure 2.1: The structure of an advanced automatic train control (ATC) system [33]

improves the quality of train operations [100]. An advanced automatic train control system could consist of an automatic train protection (ATP) system, an automatic train supervision (ATS) system, and an ATO system as shown in Figure 2.1 [33]. The onboard ATP system is responsible for supervising the train speed according to the safety speed profile and for applying an appropriate braking force if necessary. In addition, the onboard ATP system also communicates with the wayside ATP system to exchange information (e.g., temporary speed limits and the limits of movement authority (i.e., the maximum position that a train is allowed to move to)) to guarantee the safety of the operations of trains. The ATS system acts as an interface between the operator and the railway system, managing trains according to the specific regulation criteria. The ATO system controls the traction and braking force to keep the train speed under the speed limit established by the ATP system. The ATO system can be used to facilitate the driver or to operate the train in a fully automatic mode; it thus plays a key role in ensuring accurate stopping, operation punctuality, energy saving, and riding comfort [100].

An onboard ATO system consists of two levels of control actions, as conceptually illustrated in Figure 2.2. The higher level optimizes the optimal speed-position reference trajectory for the operation of the train, where the line resistance, speed limits, maximum traction and braking forces, etc. are taken into account. The low-level control is used to make the train track the pre-planned reference trajectory via certain control methods (such as PID control, model predictive control, and robust control). The traction or braking control commands are implemented to the train and information on e.g. the speed and position of the train is collected by the sensors and transferred to the ATO system in real time.

The driving performance including punctuality, energy consumption, etc. strongly depends on the optimal reference trajectory both when the train is partly or fully controlled by the ATO system. In addition, there exist several driver assistance systems to enhance the driving performance of the drivers, such as the FrightMiser, Metromiser, and the driving style manager. The FrightMiser and Metromiser systems [63] were developed by the scheduling and control group of the University of South Australia in order to calculate the

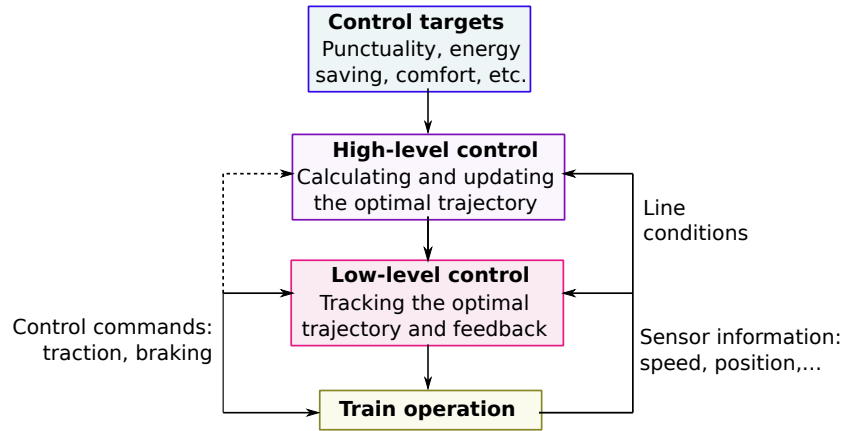


Figure 2.2: The schematic diagram of the control actions in an ATO system

optimal reference trajectory and to give advices to the drivers of long-haul trains and suburban trains respectively. That group mainly focused on minimizing the energy consumption through Pontryagin's principle. The driving style manager [39] developed by Bombardier implements discrete dynamic programming to calculate energy-efficient train trajectories, which are then displayed to the train driver. Whenever the train stops at a station, the driving style manager calculates the optimal trajectory to the subsequent station using real-time information.

ATO systems and driver assistance systems are able to take advantage of a precomputed train speed trajectory. However, if the operational conditions change, the ATO system will calculate an updated optimal trajectory. Therefore, it is important to design efficient algorithms to find the optimal speed-position reference trajectory. In the literature, various algorithms have been developed to optimize the speed trajectory for trains and these algorithms will be reviewed in Section 2.2.

2.1.2 Principles of signaling systems

Block signaling is used to maintain a safe distance between successive trains on the same track. There are two main types of signaling systems, namely fixed block signaling systems and moving block signaling systems. The main principles of those two signaling systems are presented next.

Fixed block signaling systems

Fixed block signaling (FBS) systems are commonly used in railway operation systems nowadays [98]. In FBS systems, a track is divided into blocks, the length of which depends on the maximum train speed, the worst-case braking rate, and the number of signal aspects, such as a green, yellow, or red. Each block is exclusively occupied by only one train and the presence of a train within a block is usually detected by the track circuits [115]. Furthermore, blocks are protected by wayside signals (i.e., signals next to the track) or cab signals

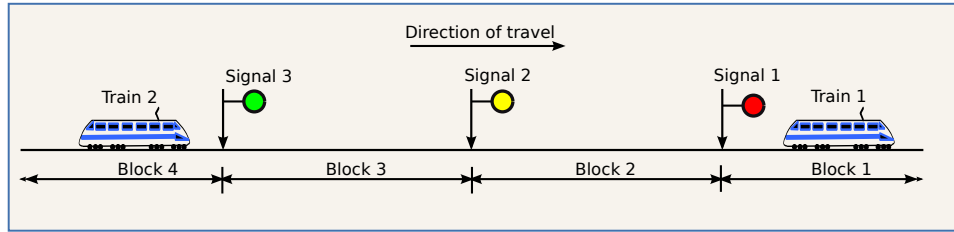


Figure 2.3: Three aspect fixed block signaling system

(i.e., visual signals on board of trains). Wayside signals are still typical in railways, however, cab signals are used more and more, in particular on high-speed lines where wayside signals cannot be watched clearly by drivers because of the high speed. There are one-block signaling and multiple-block signaling in FBS systems [98]. In one-block signaling, the indication of the block signal depends only on the state of the block section after the signal and every block signal must have a distant signal, which is supposed to provide the required approach information. In multiple-block signaling systems, the indication of a block signal depends on the state of two or more subsequent block sections.

A simple example is a two-block signaling system with three aspects, i.e. red, yellow, and green, and which is also called a three-aspect signaling system. Such a three-aspect signaling system on a line equipped with an ATP system is shown as Figure 2.3. Each block carries an electronic speed code through its track circuit. The speed code data consists of two parts: the authorized-speed code for this block and the target-speed code for the next block. The speed code data is coded by the electronic equipment controlling the track circuitry and is transmitted via tracks. This speed code data is then picked up by antennae on board of the train. If a train tries to enter a zero speed block or an occupied block, or if it enters a section at a speed higher than that authorized by the speed code, the onboard electronics will trigger an emergency brake.

Moving block signaling systems

With the increasing operational density in railway systems, railway systems with an FBS system are often suffering from a shortage in transportation capacity. Even though the line capacity of an FBS system can be increased by using shorter block lengths, the installation and maintenance cost of the signaling and track equipment may not be justified by the increased capacity. Consequently, moving block signaling (MBS) systems have been proposed to achieve a higher performance.

In an MBS system, the blocks are defined as dynamic safe zones around each train. Regular communication between trains and local traffic centers is needed for knowing the exact locations and speeds of all trains in the area controlled by the local traffic center at any given time. Therefore, compared to an FBS system, an MBS system allows trains to run closer together, thus increasing the transport capacity. The local traffic center computes the so-called limit of movement authority for every train in the area it controls and makes sure that each train will be running at a safe distance with respect to other trains (cf. Figure 2.4). More specifically, the limit of movement authority represents the maximum position that a train is allowed to move to and it is determined by the tail of the preceding train with

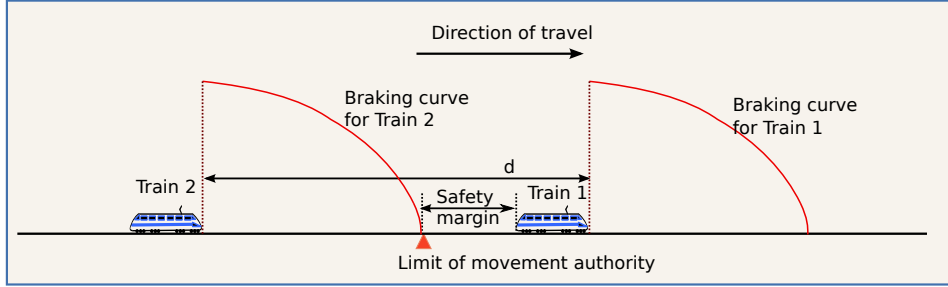


Figure 2.4: The principle of a pure MBS system

a safety margin included. In addition, the limit of movement authority of the following train moves forward continuously as the leading train travels. In the literature, four MBS schemes [99] have been discussed: moving space block signaling, moving time block signaling, pure MBS, and relative MBS. Takeuchi et al. [115] evaluated the first three schemes and compared them with the FBS scheme based on two basic criteria, viz. steady-state performance and perturbed performance. It is concluded that the pure MBS scheme gives the best performance. In addition, Takeuchi et al. [115] stated that the concept of the relative MBS has never been accepted for regular rail traffic even though it is routinely accepted for road traffic. Therefore, we will mainly consider the pure MBS scheme later on in this thesis. However, the proposed approaches can be extended to other MBS schemes too. Moreover, the pure MBS scheme is the basis of all systems currently implemented in practice [115].

In a pure MBS system, the minimum distance between two successive trains is basically the sum of the instantaneous braking distance required by the following train and a safety margin (which is introduced to avoid collisions even if the leading train comes to a sudden halt) as shown in Figure 2.4. However, the minimum distance between trains in practice should also take the train length and the running distance during the reaction time of the drivers or automatic train control systems into account.

2.2 Optimal trajectory planning of trains

In this section, we first give a literature review on the optimal trajectory planning of a single train and then the state-of-the-art on the trajectory planning of multiple trains with signaling constraints is reviewed.

2.2.1 Optimal trajectory planning of a single train

The research on optimal trajectory planning for a single train started in the 1960s. A simplified train optimal control problem was studied by Ichikawa [66], who solved the problem by using Pontryagin's principle. Later on, many researchers explored this optimal control problem by applying various methods, since it has significant effects for energy saving, punctuality, and riding comfort. These methods can be grouped into two main categories [39], viz., analytical solution and numerical optimization. The aim of this section is to give an overview of the research on optimal trajectory planning. Thereby, the research reported in

literature will be reviewed using these two categories.

- Analytical solution:

The train is usually modeled as a point mass in the optimal control problem. According to whether the traction and braking force is continuous or discrete, there are two kinds of models, i.e. continuous-input models and discrete-input models. The research on discrete-input models is mainly done by the SCG group of the University of South Australia [62, 63]. A type of diesel-electric locomotive is considered, the throttle of which can take only on a finite number of positions. Each position determines a constant level of power supply to the wheels. Several results, which include consideration of varying grades and speed restrictions, were presented. However, nowadays many locomotives or motor cars can provide a continuous traction and braking force making the use of continuous-input models necessary. For a continuous-input model, Khmelnitsky [72] described the mathematical model of the train by using the kinetic energy as the state variable. In that study, the optimal control problem was solved under varying grade profile and speed restrictions of rail lines. Liu and Golovicher [87] developed an analytical approach which combined the Pontryagin's principle and some algebraic equations to obtain the optimal solution, which contains the sequence of optimal controls and the change points, for the continuous-input model.

The optimal trajectory of an analytical solution typically contains four optimal control regimes: maximum acceleration, cruising at constant speed, coasting, and maximum deceleration. It is worth to note that the analytical methods often meet difficulties if more realistic conditions are considered that introduce complex nonlinear terms into the model equations and the constraints [74].

- Numerical optimization:

A number of advanced techniques such as fuzzy and genetic algorithms have been proposed to calculate the optimal reference trajectory for trains. Chang and Xu [22] proposed a modified differential evolution algorithm to optimally tune the fuzzy membership functions that provide a trade-off between punctuality, riding comfort, and energy consumption. The implementation of a genetic algorithm to optimize the coasting regions along a line is presented by Chang and Sim [21]. Han et al. [56] also used a genetic algorithm to construct the optimal reference trajectory taking non-constant grade profile, curve, and speed limits into account. They concluded that the performance of their genetic algorithm is better than that of the analytic solution obtained by Howlett and Pudney [63] in view of energy saving.

The train optimal control problem was solved by nonlinear programming and dynamic programming in [39]. The performance of a sequential quadratic programming algorithm and discrete dynamic programming were evaluated. Ko et al. [74] applied Bellman's dynamic programming to optimize the optimal reference trajectory. Multi-parametric quadratic programming¹ was used to calculate optimal control

¹The multi-parametric quadratic programming problem is defined as follows:

$$\begin{aligned} \min_x \quad & x^T H x + (C + q^T E)x \\ \text{s.t.} \quad & A x \leq b + Dq \end{aligned}$$

laws for trains in [123]. The nonlinear train model with quadratic resistance was approximated by a piecewise affine function. The resulting optimal control law was a piecewise affine function, which relates the traction force to the train position and speed.

A disadvantage of numerical solution methods is that the optimal solution is not always guaranteed and the convergence speed is uncertain in general. In addition, the computation often takes rather long.

2.2.2 Optimal trajectory planning of multiple trains

The solution approaches for the trajectory planning of a single train presented in Section 2.2.1 ignore the impact caused by signaling systems, e.g., an FBS system or an MBS system. In the literature, Lu and Feng [88] considered the operation of two trains on a same line and optimized the trajectory of the following train considering the constraints caused by the leading train in an FBS system. More specifically, a parallel genetic algorithm was used to optimize the trajectories for the leading train and the following train, resulting in a lower energy consumption [88]. Gu et al. [54] also considered the trajectory planning of two trains and they applied nonlinear programming to optimize the trajectory for the following train, where two situations of the leading train, i.e. running and stopped, were considered. In addition, Ding et al. [31] took the constraints caused by an MBS system into account and developed an energy-efficient multi-train control algorithm to calculate the optimal trajectories. Three optimal control regimes, i.e. maximum traction, coasting, and maximum braking, were adopted in the algorithm and the sequences of these three regimes were determined by a predefined logic [31].

For optimal trajectory planning of trains, the analytical methods often meet difficulties to find analytical solutions if more realistic conditions are considered that introduce complex nonlinear terms into the model equations and the constraints. For the numerical optimization approaches, the optimal solution is not always guaranteed. In addition, the computation is often too slow. In Chapter 3 and Chapter 4 of this thesis, we will develop efficient approaches to provide a balanced trade-off between accuracy and computational efficiency for the trajectory planning of trains. Furthermore, since the operation of trains is highly influenced by signaling systems and only a few researchers studied the impact of signaling systems in trajectory planning problem, we will also investigate the trajectory planning problem with signaling constraints in this thesis.

2.3 Urban rail transit scheduling process

A general scheduling both for interurban and urban rail transit systems is a highly complex process, which is often divided into several steps [48]: demand analysis, line planning, train scheduling, rolling stock planning, and crew scheduling as shown in Figure 2.5. First, the passenger demand has to be assessed and analyzed. Consequently, the amount of travelers wishing to go from certain origins to destinations is determined. Next, line planning is performed, which decides the routes or lines to be operated and the nominal frequency of

with q a parameter.

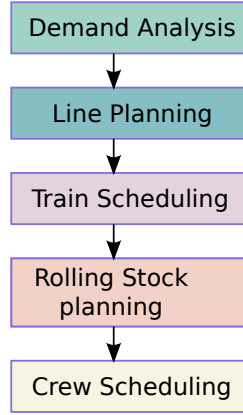


Figure 2.5: The hierarchical planning process of railway system [48]

the service. During the train scheduling step, all departure and arrival times at all stations of the lines are planned, i.e., the timetable is determined. The rolling stock planning assigns trains to all the lines. Similarly, the crews are distributed to different trains through the crew scheduling. Note that in this thesis we focus on train trajectory planning and train scheduling.

For urban rail transit systems, not all steps are equally important. There are specific characteristics for urban rail transit systems. The degree of freedom in the line planning is limited because the routes for the operation of trains have been fixed when the urban rail lines were constructed, i.e., trains do not move from one line to another during regular operation. Therefore, only the frequencies of the service, the stop-skip schedule on a certain line, and the size of train fleet can be regulated through coupling or decoupling of multiple train units to adapt varying passenger demands in urban rail transit lines. In this section, the passenger demand and the train scheduling for urban rail transit systems will be discussed in detail.

2.3.1 Passenger demand

Passenger demand estimation is the basis for the whole planning process. Traditionally, demand estimation relies heavily on costly and unreliable manual data collection, e.g., using passenger surveys to estimate origin-destination (OD) travel patterns. The results obtained by this kind of manual data collection maybe subject to bias and even error [146]. However, nowadays most urban rail transit systems have been equipped with automatic passenger counting systems and automatic fare collection systems, which can provide accurate passenger information to rail operators. Automatic passenger counting systems are used to count the number of passengers getting on and getting off trains at stations. With automatic fare collection systems, passengers need to use their fare cards when entering and exiting urban rail transit systems, so the location and time of each passenger's fare transactions can be recorded.

The passenger demand can be described by the following two ways:

- OD-independent passenger demands

When describing the passenger demand in an OD-independent way, the origin and destination of each passenger are not considered. The passenger arrival rate at a certain station is then e.g. defined as the number of passengers arriving at the station during a predefined time period [36].

- OD-dependent passenger demands

The OD-dependent passenger demand is defined as an estimation of the number of people wishing to travel from an origin to a destination over a certain period of time during the day. The OD-dependent passenger demand can be conducted using the available passenger information, see [82, 143, 146] for details.

2.3.2 Train scheduling

Train scheduling has been studied for decades via different techniques [23], such as linear programming [101, 114], integer or nonlinear programming [48, 59, 75, 81], and graph theory [28]. In these papers, the available resources, e.g., the single tracks and the crossings, are shared by trains with different origins and destinations. Thus, the trains may overtake and cross each other at some specific locations, such as sidings and crossings. However, the lines in urban rail transit usually have double tracks, where each track is used for one direction of train operation. Train overtaking and crossing is normally not allowed during the operations of urban rail transit systems. Here, we concentrate on urban rail transit systems.

Scheduling of trains for urban rail transit

In 1980, Cury et al. [26] presented a methodology to generate optimal schedules for metro lines based on a model of the train movements and of the passenger behavior. The performance index included passenger delay, passenger comfort, and the efficiency of the operation of trains. The resulting nonlinear scheduling problem was recast into several sub-problems by Lagrangian relaxation and then solved in a hierarchical manner [26]. Since the convergence rate of the hierarchical decomposition algorithm can be quite poor in some cases, Assis and Milani [4] proposed a model predictive control algorithm based on linear programming to optimize the train schedule. The algorithm proposed in [4] can effectively generate train schedules for the whole day. Kwan and Chang [78] applied a heuristic-based evolutionary algorithm to solve the train scheduling problem, where the operation costs and the passenger dissatisfaction are included in the performance index. The train scheduling problem is formulated as a periodic event-scheduling problem based on a graph model in [83], which is then solved using integer programming methods. The approach proposed by Liebchen has been applied in Berlin subway systems [84]. The passenger transfer behavior and transfer waiting times are considered in [142], which presents a mixed-integer programming optimization model to synchronize the train schedules for different urban rail transit lines. Furthermore, a demand-oriented timetable design is proposed in [1], where the optimal train frequency and the capacity of trains are first determined and then the schedule of trains are optimized. Vazquez et al. [124] proposed a stochastic approximation approach to adjust the frequencies of different urban transit lines according to the observed variable

passenger demand. However, the energy consumption of railway operation and dwell times at stations are not included in the model of [124].

Real-time scheduling or rescheduling of trains

Since trains do not run exactly according to the predefined schedule in practice, real-time scheduling approaches have been proposed. In the literature, there are several interpretations for real-time scheduling. For interurban railway systems, real-time scheduling is based on the existing timetable data and is used to handle route conflicts due to train delays or incidents [17, 25, 28, 32, 70, 71, 76, 92, 121]. However, in urban rail transit systems, real-time scheduling regulates the headways between trains based on a train schedule with a constant headway.

Several rescheduling approaches have been proposed for urban rail transit systems [42]: holding, zone scheduling, short turning, deadheadin, and/or stop-skipping [20, 36, 47, 111]. Holding is used to regulate the headways by holding an early-arriving train, or a train with a relatively short leading headway [36]. In zone scheduling [47], the whole line is divided into several zones, where the trains stop at all stations within a single zone and then run to the terminal station without stopping. The required number of trains and drivers and passenger travel times may be reduced by the zone scheduling, where the zones are defined based on the passenger flows. There are short-turning and full-length trips operating on the line in the short-turning strategy [20, 111], where the short-turning trips serve only the zone with high demands and the full-length trips run the whole line. The deadheading strategy involves some trains running empty through a number of stations at the beginning of their trips to reduce the headways at later stations [35, 42]. A dynamic stop-skipping strategy is frequently used in lines with high demands, as it allows those trains that are late and behind the schedule to skip certain low-demand stations and in that way increase the running speed.

Wong and Ho [141] proposed dwell time and running time control for the real-time rescheduling problem of urban rail transit systems. They applied a dynamic programming approach to their rescheduling model to devise an optimal set of dwell times and running times [141]. In addition, Goodman and Murata [52] formulated the train rescheduling problem from the perspective of passengers, where a gradient calculation method was developed to solve the rescheduling problem in real time. Furthermore, Norio et al. [94] proposed to use passenger dissatisfaction as a criterion for the rescheduling and applied a meta-heuristics algorithm to solve the rescheduling problem.

As demonstrated in [41, 79], the stop-skipping strategy can reduce the passenger travel time and the operation cost of rail transit operators. The stop-skipping operation was first developed for the Chicago metro system in 1947 [41]. Now, the SEPTA line in Philadelphia, Helsinki commuter rail, and the metro system in Santiago, Chile apply the stop-skipping train schedule in practice. They apply a static stop-skipping strategy [79], i.e., the A/B skip-stop strategy, where stations are divided into three types: A, B, and AB; A train services stop at A stations and AB stations, while B train services stop at B stations and AB stations. Major stations are usually labeled with the type AB; so all trains stop there. The transit operators provide the stop-skipping information to passengers via panels at platforms and announcements in the trains. The Santiago metro operator stated that passengers adapt to the stop-skipping strategy quickly [41]. Elberlein [34] formulated the stop-skipping problem as a mixed integer nonlinear programming problem, where trains can skip some station strings

(i.e., a collection of consecutive stations). Fu et al. [42] represented the skipping of stations by trains as binary variables and obtained a mixed integer nonlinear programming problem, which was solved using an exhaustive approach. Lee [79] applied genetic algorithm to obtain the optimal train schedule and to find the best combination of the stop-skipping trains and the all-stop trains based on the A/B stop-skipping strategy.

The passenger demand for urban rail transit systems increases dramatically and varies significantly along urban rail transit lines and the time of the day. To satisfy the passenger demand, trains are operated with small headway, which is around 2-5 minutes. Therefore, the scheduling of trains according to the passenger demand becomes more and more important for reducing the operation costs and for guaranteeing passenger satisfaction².

2.4 Summary

A brief introduction to the operation of trains and the principle of signaling systems has been presented in this chapter. We have briefly discussed the literature of the optimal trajectory planning for trains and of the train scheduling for urban rail transit systems. In addition, we have motivated why the work of this thesis is needed.

²Passenger satisfaction can be characterized by waiting times, onboard times, the number of transfers, the onboard crowdedness, etc.

Chapter 3

Optimal Trajectory Planning for a Single Train

In this chapter, the optimal trajectory planning problem for the operation of a single train under various constraints and with a fixed arrival time is considered. The objective function is a trade-off between the energy consumption and the riding comfort. Two approaches are proposed to solve this optimal control problem, viz. a pseudospectral method and a mixed integer linear programming (MILP) approach. In the pseudospectral method, the optimal trajectory planning problem is recast into a multiple-phase optimal control problem, which is then transformed into a nonlinear programming problem. For the MILP approach, the optimal trajectory planning problem is reformulated as an MILP problem by approximating the nonlinear terms by piecewise affine (PWA) functions. The performance of these two approaches will be compared through a case study.

The research discussed in this chapter is based on [127, 128, 130].

3.1 Introduction

As has been mentioned in the literature survey of Chapter 2, it is important to design efficient algorithms to find the optimal speed-position reference trajectory. This chapter proposes two approaches to determine the optimal trajectory, viz. a pseudospectral method and an MILP approach.

Over the last decade, pseudospectral methods have risen to prominence in the numerical optimal control area [37]. Pseudospectral methods were applied to solving optimal control problems [50], such as orbit transfers, lunar guidance, magnetic control. However, to the author's best knowledge, pseudospectral methods have not been applied to trajectory planning of trains. Therefore, the pseudospectral method is used for the first time to solve the train trajectory planning problem.

On the other hand, multi-parametric quadratic programming is used in [123] to calculate the optimal control law for train operations. The nonlinear train model with quadratic resistance is approximated by an PWA function. Inspired by [123], in this chapter we propose to solve the optimal trajectory problem as an MILP problem.

The remainder of this chapter is organized as follows. In Section 3.2 a nonlinear model of train operations is presented. Section 3.3 formulates the optimal trajectory planning problem. In Section 3.4, two solution approaches are proposed to solve the resulting optimal control problem: a pseudospectral method and a mixed integer linear programming approach. Section 3.5 illustrates with a case study how to calculate the optimal reference trajectory by the pseudospectral method and the MILP approach and it also compares these two approaches with the discrete dynamic programming approach. We conclude with a short discussion of some topics for future work in Section 3.6.

3.2 Model formulation

3.2.1 Train model

In the literature on train optimal control, the mass-point model of train is often used [40]. The motion of a train can then be described by the following simple continuous-time model [87]:

$$mp \frac{dv}{dt} = u(t) - R_b(v) - R_l(s, v), \quad (3.1)$$

$$\frac{ds}{dt} = v, \quad (3.2)$$

where m is the mass of the train, p is a factor that represents the rotating mass [57], v is the velocity of the train, s is the position of the train, u is the control variable, i.e., the traction or braking force, which is bounded by the maximum traction force u_{\max} and the maximum braking force u_{\min} , $u_{\min} \leq u \leq u_{\max}$, $R_b(v)$ is the basic resistance including roll resistance and air resistance, and $R_l(s, v)$ is the line resistance caused by track grade, curves, and tunnels.

The maximum traction force u_{\max} is often considered as constant in the literature [62]. However, in reality it is a function of the velocity v . Due to the maximum adhesion and the characteristics of the power equipment [57], the diagram of the maximum traction force u_{\max} as a function of the speed v normally looks like the one shown in Figure 3.1 [57]. This diagram is described as a group of hyperbolic or parabolic formulas in [57], where each formula approximates the actual traction force for a certain speed interval. For example, if the train speed v belongs to interval $[v_j, v_{j+1}]$, then the maximum traction force can be written as

$$u_{\max}(v) = c_{1,j} + c_{2,j}v + c_{3,j}v^2, \quad v \in [v_j, v_{j+1}], \quad (3.3)$$

or

$$u_{\max}(v) = c_{h,j}/v, \quad v \in [v_j, v_{j+1}], \quad (3.4)$$

for $j = 1, 2, \dots, M-1$, where v_j , v_{j+1} , $c_{1,j}$, $c_{2,j}$, $c_{3,j}$, and $c_{h,j}$ are determined by the characteristics of the train.

According to the arguments for the maximum braking force given in [57], the full braking effort is reserved for an emergency stop. More specifically, under normal circumstances the train driver or automatic train operation system brakes in a comfort mode, where the

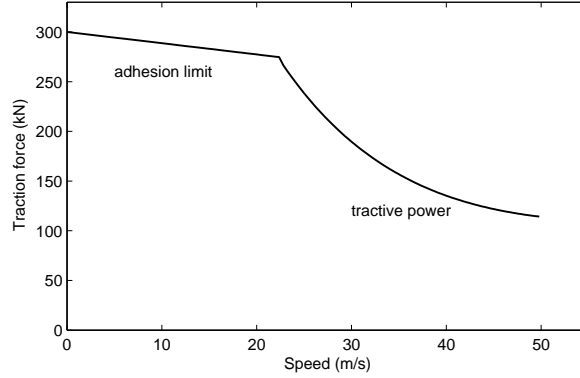


Figure 3.1: Maximum traction force as a function of train speed [57]

maximum force for the service braking is 0.75 times that of the emergency braking, i.e., the full braking effort. On the other hand, the braking effort (including the maximum braking effort) is considered as constant by some common safety systems, such as the European Train Control System and the German continuous train control system [57]. Therefore, the maximum force for service braking is taken to be constant in this chapter.

In practice, according to the Strahl formula [103] the basic resistance $R_b(v)$ can be described as

$$R_b(v) = m(a_1 + a_2v^2),$$

where the coefficients a_1 and a_2 depend on the train characteristics and the wind speed. These coefficients can be estimated from the known data about the train.

The line resistance $R_l(s, v)$ caused by track slope, curves, and tunnels can be described as [90]

$$R_l(s, v) = mg \sin \alpha(s) + f_c(r(s)) + f_t(l_t(s), v), \quad (3.5)$$

where g is the gravitational acceleration, $\alpha(s)$, $r(s)$, and $l_t(s)$ are the slope, the radius of the curve, and the length of the tunnel along the track, respectively. The curve resistance $f_c(\cdot)$ and the tunnel resistance $f_t(\cdot)$ are given by empirical formulas. An example of such an empirical formula of the curve resistance is Roeckl's formula [64]:

$$f_c(r(s)) = \begin{cases} \frac{6.3}{r(s)-55}m & \text{for } r(s) \geq 300 \text{ m,} \\ \frac{4.91}{r(s)-30}m & \text{for } r(s) < 300 \text{ m.} \end{cases}$$

When running in tunnels, the train experiences a higher air resistance that depends on the tunnel form, the smoothness of tunnel walls, the exterior surface of the train, and so on. An example of an expression for the tunnel resistance is as follows [43, 64]. If there is a limiting gradient¹ in the tunnel, then an empirical formula for the tunnel resistance is

$$f_t(l_t(s), v) = 1.296 \cdot 10^{-9} l_t(s) m g v^2.$$

¹A limiting gradient is defined as the maximum railway gradient that can be climbed without the help of a second power unit.

If there does not exist a limiting gradient, the tunnel resistance can be calculated by the following empirical formula:

$$f_t(l_t(s), v) = 1.3 \cdot 10^{-7} l_t(s) mg.$$

For the tracks outside the tunnels, the tunnel resistance is equal to zero.

Remark 3.1 Different types of rolling stock can be modeled by the mass-point model, the parameters of which, such as mass, maximum traction force, and resistance coefficients, may vary according to different types of rolling stock. The existing infrastructure of tracks can be described accurately by using the line resistance (3.5), which includes track slope, curves, tunnels. In addition, the signaling aspects and the disturbances caused by other trains are assumed to be taken care of by a lower control level. Furthermore, different train categories (high speed trains, regional and intercity trains, freight trains) can be handled by a higher control level during train scheduling, which specifies different running times and dwell times for each train. The approaches proposed in this chapter can then be applied to obtain the optimal trajectory for each trip between two stations to save energy and to ensure passenger comfort based on the given timetable. \square

3.2.2 An assumption about the line resistance

The line resistance $R_l(s, v)$ caused by track slope, curves, and tunnels is a nonlinear function of the train's position and speed. In order to simplify the consideration of the line resistance, we rewrite $R_l(s, v)$ in (3.5) as

$$R_l(s, v) = \xi_1(s) + \xi_2(s)v^2, \quad (3.6)$$

where $\xi_1(s)$ collects terms that do not depend on the train's speed. In the sequel of this chapter, $\xi_1(s)$ and $\xi_2(s)$ are assumed to be piecewise constant functions, which can be written as

$$\begin{aligned} \xi_1(s) &= \xi_1^{(i)} \quad \text{for } s \in [s_0^{(i)}, s_f^{(i)}], \\ \xi_2(s) &= \xi_2^{(i)} \quad \text{for } s \in [s_0^{(i)}, s_f^{(i)}], \end{aligned} \quad (3.7)$$

for $i = 1, 2, \dots, N_R$, where N_R is the number of the piecewise constant subfunctions, $s_0^{(1)} = s_{\text{start}}$ is the position at the beginning of the route, $s_f^{(N_R)} = s_{\text{end}}$ is the position at the end of the route, and $s_0^{(i+1)} = s_f^{(i)}$ for $i = 1, 2, \dots, N_R - 1$. Therefore, the line resistance can be written as

$$R_l(s, v) = \xi_1^{(i)} + \xi_2^{(i)}v^2, \quad \text{for } s \in [s_0^{(i)}, s_f^{(i)}]. \quad (3.8)$$

3.3 Mathematical formulation of the single train trajectory planning problem

As stated in [87], reference trajectory planning for trains can be formulated as an optimal control problem. The traction or braking force u is then the control variable. The state variables are the train position s and speed v . The objective function to be minimized could be the trip time, the energy consumption for a given trip time, or the total operation cost (a

weighted sum of energy consumption and trip time). In this chapter, we consider the objective criterion to be the energy consumption in a fixed time span $[0, T]$ with T determined by a fixed or a flexible timetable [29, 57], or being the result of a rescheduling operation of railway traffic after disturbances [76]. In addition, the riding comfort is considered, which is expressed as a function of the change of the control variable u , since reducing the number of transitions and the rate of change of u may improve passenger comfort [22]. The objective function can thus be written as:

$$J = \int_0^T \left(u(t) \cdot v(t) + \lambda \cdot \left| \frac{du(t)}{dt} \right| \right) dt \quad (3.9)$$

where J is the weighted integral of the energy consumption and riding comfort and $\lambda > 0$ is the weight which can be decided by rail operators based on experience. This function will be minimized subject to the train dynamics (3.1) and (3.2), the constraints

$$u_{\min} \leq u(t) \leq u_{\max}(v) \quad (3.10)$$

$$0 \leq v(t) \leq V_{\max}(s) \quad (3.11)$$

and the boundary conditions

$$s(0) = s_{\text{start}}, \quad v(0) = v_{\text{start}}, \quad (3.12)$$

$$s(T) = s_{\text{end}}, \quad v(T) = v_{\text{end}}, \quad (3.13)$$

where the maximum allowable velocity $V_{\max}(s)$ depends on the train characteristics and the line conditions, and as such it is usually a piecewise constant function of the coordinate s [72, 87]; v_{start} and v_{end} are the velocity at the beginning and the end of the route.

As proposed in some previous works [40, 62, 72, 87], it is better to choose the position s as an independent variable rather than the time t . On the one hand, the choice of the position s as the independent variable will simplify the consideration of track-related data, such as line resistance and speed limits. On the other hand, the analytical and numerical study of the optimal control problem will be significantly simplified then. Furthermore, Khmel'nitsky [72] chose the total energy of the train and time t as states where the total energy includes kinetic and potential energy. Similarly, Franke et al. [40] used kinetic energy per mass unit and time as states. The choice of kinetic energy instead of speed v will facilitate the study of the optimal control problem, because this choice eliminates some (but not all) of the model nonlinearities. Therefore, we also choose kinetic energy per mass unit $\tilde{E} = 0.5v^2$ and time t as states, and the position s as the independent variable. The continuous-time model (3.1) and (3.2) can then be rewritten as the following continuous-space model²:

$$mp \frac{d\tilde{E}}{ds} = u(s) - R_b(\sqrt{2\tilde{E}}) - R_l(s, v), \quad (3.14)$$

$$\frac{dt}{ds} = \frac{1}{\sqrt{2\tilde{E}}}. \quad (3.15)$$

²The transformation from $\frac{dv}{dt}$ to $\frac{d\tilde{E}}{ds}$ goes as follows:

$$\frac{dv}{dt} = \frac{dv}{ds} \frac{ds}{dt} = v \frac{dv}{ds} = \frac{d\tilde{E}}{ds}$$

where $\tilde{E} = 0.5v^2$.

The optimal control problem corresponding to (3.9)-(3.13) can be stated as: minimize the objective function³

$$J = \int_{s_{\text{start}}}^{s_{\text{end}}} \left(u(s) + \lambda \cdot \left| \frac{du(s)}{ds} \right| \right) ds \quad (3.16)$$

subject to the model (3.14) and (3.15), the constraints

$$u_{\min} \leq u(s) \leq u_{\max}(v), \quad (3.17)$$

$$0 \leq \tilde{E}(s) \leq \tilde{E}_{\max}(s), \quad (3.18)$$

and the boundary conditions,

$$\tilde{E}(s_{\text{start}}) = \tilde{E}_{\text{start}}, \quad \tilde{E}(s_{\text{end}}) = \tilde{E}_{\text{end}}, \quad (3.19)$$

$$t(s_{\text{start}}) = 0, \quad t(s_{\text{end}}) = T, \quad (3.20)$$

where $\tilde{E}_{\max}(s) = 0.5V_{\max}^2(s)$, $\tilde{E}_{\text{start}} = 0.5v_{\text{start}}^2$, and $\tilde{E}_{\text{end}} = 0.5v_{\text{end}}^2$. For the above equations, it is assumed that the unit kinetic energy $\tilde{E}(s)$ satisfies $\tilde{E}(s) \geq E_{\min} > 0$ with E_{\min} a small positive number, which means the train's speed is always strictly larger than zero, i.e., the train travels nonstop. Khmelnitsky [72] states that this assumption is not restrictive in practice for two reasons. First, the speed of the initial start and the terminal stop can be approximated by small nonzero velocities. Second, stops at an intermediate point of the trip will in principle not be planned deliberately in the optimal control design for a single train's operation since intermediate stops and the corresponding restarts would result in an increased energy consumption.

Remark 3.2 There may also exist some other constraints that result from the timetable, real-time operation restrictions, or the real-time rescheduling process. Albrecht et al. [2, 3] classified these operational constraints into two groups: target points and target windows. Target points correspond to fixed passing times, which could be arrival and departure times at stations. In dense networks, target points could also be passing times at certain places where overtaking and crossing of trains is planned. The scheduled arrival times at minor stations without connections with other trains can be in general regarded as target windows. If the train reaches a certain place exactly on time according to the defined target point or in the target window, then conflicts can be avoided. It is assumed that the positions corresponding to target points or target window constraints are s_{k_j} with k_j in $\{1, 2, \dots, N\}$. The operational constraints can be included in the optimal control problem as follows:

- for target points:

$$t(s_{k_j}) = T_{\text{target},j}, \quad (3.21)$$

³The transformation from $uvdt$ to uds goes as follows:

$$u \cdot v \, dt = u \frac{ds}{dt} dt = u \, ds.$$

In addition, the transformation from $\left| \frac{du}{dt} \right| dt$ to $\left| \frac{du}{ds} \right| ds$ goes as follows:

$$\left| \frac{du}{dt} \right| dt = \left| \frac{du}{ds} \right| \left| \frac{ds}{dt} \right| dt = \left| \frac{du}{ds} \right| ds, \text{ if } \frac{ds}{dt} > 0.$$

- for target windows:

$$T_{\text{target}_{\min},j} \leq t(s_{k_j}) \leq T_{\text{target}_{\max},j}, \quad (3.22)$$

where $T_{\text{target},j}$ is the fixed passing time for train to pass position s_{k_j} , and $T_{\text{target}_{\min},j}$ and $T_{\text{target}_{\max},j}$ are the minimum and maximum passing time at position s_{k_j} respect to the target window constraints. \square

Remark 3.3 In order to deal with disturbance, we can apply a rolling horizon approach when solving the trajectory planning problem. A detailed description on how to do this for the case of train scheduling for an urban rail transit network is given in Section 7.4. Those ideas can be adopted to the setting of the current chapter. \square

3.4 Solution approaches

In this section, the train trajectory planning problem is solved by a pseudospectral method and a mixed integer linear programming approach.

3.4.1 Pseudospectral method

A brief introduction

Pseudospectral methods were researched widely in the 1970s for solving partial differential equations (PDEs) in fluid dynamics [19]. Later on, they became an important methodology for the numerical solution of PDEs. From the 1990s on, pseudospectral methods were applied for solving optimal control problems [50], such as orbit transfers, lunar guidance, and magnetic control. Recently, the scope of application has been broadened as a result of significant progress in large-scale computation.

The pseudospectral method directly formulates the original optimal control problem into a nonlinear programming problem, which can be solved numerically using a sparse nonlinear programming solver to find approximate locally optimal solutions [37]. It is shown by approximation theory and practice that the pseudospectral method is well-suited for approximating smooth functions, integrations, and differentiations [19]. All those approximations are relevant to optimal control problems, e.g., the differential equations of the optimal control problem can then be approximated by algebraic equations [105]. The main advantages of the pseudospectral method are the exponential rate of convergence and that it is possible to achieve a good accuracy with coarse grids [19, 51].

In the pseudospectral method, the continuous-time state and control functions are approximated using orthogonal polynomials based on interpolation at orthogonal collocation points [38], such as the commonly used Legendre-Gauss-Lobatto points, which are the roots of $(1 - x^2) \frac{dL_N(x)}{dx}$, where L_N is the Legendre polynomial of order N [19]. The derivative of the approximated state can be expressed in terms of the approximated state vector by using a differentiation matrix at the collocation points [106]. When the optimal control problem includes discontinuities in states, control inputs, objective functional, or dynamic constraints, the pseudospectral method is employed in the form of a multiple-phase approach, where the problem is divided into a relatively small number of subintervals and global collocation is performed in each subinterval [106].

There exist several commercial and free packages that implement the pseudospectral method: PROPT [107] and DIDO [104] are examples of commercial software that run under Matlab. A Matlab-based open source tool that uses the Gauss pseudospectral method is GPOPS [102]. PSOPT is an open source optimal control package written in C++, including Legendre and Chebyshev pseudospectral discretizations [9].

Formulation of the optimal trajectory planning problem

We can reformulate the train trajectory planning problem (3.14)-(3.20) into the following general optimal control problem with N_p phases [8, 106]. It is worth noting that N_p is not equal to N_R of (3.7)-(3.8), but it will in general be larger. The objective function (3.16) to be minimized can be rewritten as

$$J = \sum_{i=1}^{N_p} \left[\int_{s_0^{(i)}}^{s_f^{(i)}} \left[u^{(i)}(s) + \lambda \left| \frac{du^{(i)}(s)}{ds} \right| \right] ds \right]. \quad (3.23)$$

Given that non-smoothness causes problems in gradient-based nonlinear optimization, a smooth version of the absolute value function can be written as

$$|\sigma| \approx \psi(\sigma) = \frac{\sigma^2}{\sqrt{\sigma^2 + c^2}}, \quad (3.24)$$

where c is a constant deciding the smoothness of the function. Thus, the smooth objective function can be written as

$$J = \sum_{i=1}^{N_p} \left[\int_{s_0^{(i)}}^{s_f^{(i)}} \left[u^{(i)}(s) + \lambda \psi\left(\frac{du^{(i)}(s)}{ds}\right) \right] ds \right]. \quad (3.25)$$

The objective function (3.25) is subject to the differential constraints

$$\dot{x}^{(i)}(s) = \phi^{(i)}(x^{(i)}(s), u^{(i)}(s), s), \quad s \in [s_0^{(i)}, s_f^{(i)}], \quad (3.26)$$

where $x^{(i)}(s)$ is the state of the system in the i th phase, i.e., $x^{(i)}(s) = [E^{(i)}(s) \quad t^{(i)}(s)]^T$, and the functions $\phi^{(i)}(\cdot)$ are defined by model equations (3.14)-(3.15) and the piecewise line resistance (3.8). The path constraints of the optimal control problem are defined by (3.3), (3.4), and (3.17). Note that the path constraints caused by the maximum traction force are non-smooth. They can be approximated by smooth constraints by introducing a smooth version of the Heaviside function $H(\sigma)$, defined as $H(\sigma) = 1$ if $\sigma > 0$, and $H(\sigma) = 0$ otherwise [69]. The approximation is implemented as

$$H(\sigma) \approx 0.5(1 + \tanh(\sigma/h)) \quad (3.27)$$

where $h > 0$ is a small real number. The path constraints can then be written as

$$\mathbf{p}_L^{(i)} \leq \mathbf{p}^{(i)}(x^{(i)}(s), u^{(i)}(s), s) \leq \mathbf{p}_U^{(i)}, \quad s \in [s_0^{(i)}, s_f^{(i)}]. \quad (3.28)$$

For the train trajectory planning problem, the initial position of the $(i+1)$ th phase is equal to the final position of the i th phase, so one of the phase boundary constraints can be written as

$$s_0^{(i+1)} - s_f^{(i)} = 0. \quad (3.29)$$

In addition, the states and the control variables are continuous across the phase boundary, which is

$$x(s_0^{(i+1)}) - x(s_f^{(i)}) = 0, \quad (3.30)$$

$$u(s_0^{(i+1)}) - u(s_f^{(i)}) = 0. \quad (3.31)$$

In general, phase boundary constraints [8, 12] that link all the states and control inputs across the boundaries can be included in

$$\begin{aligned} \Psi_L \leq & \Psi(x^{(1)}(s_0^{(1)}), u^{(1)}(s_0^{(1)}), x^{(1)}(s_f^{(1)}), u^{(1)}(s_f^{(1)}), s_0^{(1)}, s_f^{(1)}, \\ & x^{(2)}(s_0^{(2)}), u^{(2)}(s_0^{(2)}), x^{(2)}(s_f^{(2)}), u^{(2)}(s_f^{(2)}), s_0^{(2)}, s_f^{(2)}, \\ & \vdots \\ & x^{(N_p)}(s_0^{(N_p)}), u^{(N_p)}(s_0^{(N_p)}), x^{(N_p)}(s_f^{(N_p)}), u^{(N_p)}(s_f^{(N_p)}), s_0^{(N_p)}, s_f^{(N_p)}) \leq \Psi_U. \end{aligned} \quad (3.32)$$

Note that (3.29)-(3.31) are special cases of (3.32) with $\Psi_{L,i} = \Psi_{U,i}$.

The bound constraints can be written as

$$\begin{aligned} u_L^{(i)} & \leq u^{(i)}(s) \leq u_U^{(i)}, \quad s \in [s_0^{(i)}, s_f^{(i)}], \\ x_L^{(i)} & \leq x^{(i)}(s) \leq x_U^{(i)}, \quad s \in [s_0^{(i)}, s_f^{(i)}]. \end{aligned} \quad (3.33)$$

The resulting multiple-phase optimal control problem can be solved using the nonlinear programming methods.

However, the computation of the pseudospectral method is in general too slow for the real-time application of ATO system. When the operational conditions (e.g., speed limits or trip time) change while the train is driving (e.g., due to an accident or bad weather conditions), the ATO system needs to recalculate the optimal trajectory. If the algorithm of the ATO system takes a large computation time to calculate the optimal trajectory, then it is too late for the train to react timely. Therefore, in the next subsection we propose an alternative approach, i.e., an MILP approach, to calculate the optimal trajectory. It is worth to note that the optimal solution of the pseudospectral method satisfies the necessary (but not always sufficient) conditions of optimality [107]. So it is guaranteed that the returned solution cannot be improved by an infinitesimal change in the trajectory, but there may exist completely different trajectories that yield a better performance. On the contrary, an MILP problem can be solved efficiently by existing solvers that guarantee the global optimum for the proposed MILP problem.

3.4.2 Mixed integer linear programming

Vařak et al. [123] proposed a discrete-time model of the train operation to calculate the optimal control law by multi-parametric quadratic programming. They split the time period into K intervals and assumed the traction force or braking force to be constant on each interval $[kT_s, (k+1)T_s]$, where T_s is the sampling time. Franke et al. [40] similarly split the position horizon $[s_{\text{start}}, s_{\text{end}}]$ into N intervals to get a discrete-space model. They assumed that the track and train parameters as well as traction or braking force can be considered as constant in each interval $[s_k, s_{k+1}]$ with length $\Delta s_k = s_{k+1} - s_k$, for $k = 1, 2, \dots, N$. Note

that $s_1 = s_{\text{start}}$ and $s_{N+1} = s_{\text{end}}$. In this chapter, we obtain a discrete-space model in a similar way as in [40], since the optimal control problem is stated by the choice of s as the independent variable. By redefining the discretization of the interval $[s_{\text{start}}, s_{\text{end}}]$ if necessary, we can assume without loss of generality that $\xi_1(s)$ and $\xi_2(s)$ (cf. Section 3.2.2) are of the following form:

$$\begin{aligned}\xi_1(s) &= \xi_{1,k} \quad \text{for } s \in [s_k, s_{k+1}], \\ \xi_2(s) &= \xi_{2,k} \quad \text{for } s \in [s_k, s_{k+1}].\end{aligned}$$

for $k = 1, 2, \dots, N$.

Transformation properties

First, we introduce three properties according to [140]. Consider the statement $\tilde{f}(\tilde{x}) \leq 0$, where $\tilde{f}: \mathbb{R}^n \rightarrow \mathbb{R}$ is affine, $\tilde{x} \in \chi$ with $\chi \subset \mathbb{R}^n$ and let

$$\tilde{M} = \max_{\tilde{x} \in \chi} \tilde{f}(\tilde{x}), \quad \tilde{m} = \min_{\tilde{x} \in \chi} \tilde{f}(\tilde{x}). \quad (3.34)$$

If we introduce the logical variable $\delta \in \{0, 1\}$, then the following equivalence holds:

$$[\tilde{f}(\tilde{x}) \leq 0] \Leftrightarrow [\delta = 1] \quad \text{is true iff} \quad \begin{cases} \tilde{f}(\tilde{x}) \leq \tilde{M}(1 - \delta) \\ \tilde{f}(\tilde{x}) \geq \varepsilon + (\tilde{m} - \varepsilon)\delta \end{cases} \quad (3.35)$$

where ε is a small positive number (typically the machine precision) that is introduced to transform a strict equality into a non-strict inequality, which fits the mixed integer linear programming (MILP) frameworks [10].

The product of two logical variables $\delta_1 \delta_2$ can be replaced by an auxiliary logical variable $\delta_3 = \delta_1 \delta_2$, i.e., $[\delta_3 = 1] \Leftrightarrow [\delta_1 = 1] \wedge [\delta_2 = 1]$, which is equivalent to

$$\begin{cases} -\delta_1 + \delta_3 \leq 0, \\ -\delta_2 + \delta_3 \leq 0, \\ \delta_1 + \delta_2 - \delta_3 \leq 1. \end{cases} \quad (3.36)$$

Moreover, the product $\delta \tilde{f}(\tilde{x})$ can be replaced by the auxiliary real variable $z = \delta \tilde{f}(\tilde{x})$, which satisfies $[\delta = 0] \Rightarrow [z = 0]$ and $[\delta = 1] \Rightarrow [z = \tilde{f}(\tilde{x})]$. Then $z = \delta \tilde{f}(\tilde{x})$ is equivalent to

$$\begin{cases} z \leq \tilde{M}\delta, \\ z \geq \tilde{m}\delta, \\ z \leq \tilde{f}(\tilde{x}) - \tilde{m}(1 - \delta), \\ z \geq \tilde{f}(\tilde{x}) - \tilde{M}(1 - \delta). \end{cases} \quad (3.37)$$

It is noted that (3.35), (3.36), and (3.37) yield linear inequalities since \tilde{f} is affine.

The mixed logical dynamic model

In the interval $[s_k, s_{k+1}]$, the differential equation of the kinetic energy (3.14) can now be rewritten as

$$\frac{d\tilde{E}}{ds} = \frac{1}{mp} u(k) - \frac{2(a_2 + \xi_{2,k})}{\rho} \tilde{E}(s) - \frac{1}{\rho} (a_1 + \xi_{1,k}),$$

where $u(k)$ is a constant in the interval $[s_k, s_{k+1}]$. By defining $\zeta = \frac{1}{mp}$, $\eta_k = -\frac{2(a_2 + \xi_{2,k})}{p}$, $\gamma_k = -\frac{1}{p}(a_1 + \xi_{1,k})$, this equation can be rewritten as

$$\frac{d\tilde{E}}{ds} = \zeta u(k) + \eta_k \tilde{E}(s) + \gamma_k. \quad (3.38)$$

We have to solve this differential equation with initial condition $\tilde{E}(s_k) = E(k)$. Then we obtain the following formula for $E(s_k + 1)$:

$$E(s_k + 1) = e^{\eta_k \Delta s_k} E(s_k) + (e^{\eta_k \Delta s_k} - 1) \frac{\zeta}{\eta_k} u(k) + (e^{\eta_k \Delta s_k} - 1) \frac{\gamma_k}{\eta_k} \quad (3.39)$$

with $E(s_1) = \tilde{E}_{\text{start}}$ and $E(s_{N+1}) = \tilde{E}_{\text{end}}$.

Remark 3.4 For the sake of simplicity, we use $E(k)$ as a short-hand notation for $\tilde{E}(s_k)$ from now on. \square

Defining $a_k = e^{\eta_k \Delta s_k}$, $b_k = (e^{\eta_k \Delta s_k} - 1) \frac{\zeta}{\eta_k}$ and $c_k = (e^{\eta_k \Delta s_k} - 1) \frac{\gamma_k}{\eta_k}$, (3.39) can now be simplified as follows:

$$E(k+1) = a_k E(k) + b_k u(k) + c_k. \quad (3.40)$$

Note that this is an affine equation. As regards the differential equation (3.15), we approximate it by using a trapezoidal integration rule [6]:

$$t(k+1) = t(k) + \frac{1}{2} \left(\frac{1}{\sqrt{2E(k)}} + \frac{1}{\sqrt{2E(k+1)}} \right) \Delta s_k \quad (3.41)$$

with $t(1) = 0$. In addition, the nonlinear part in this equation will be approximated by an PWA function. There are various methods for approximating functions in an PWA way, see e.g., the overview by Azuma et al. [7]. In this chapter, we first select the number of regions of the PWA function and then optimize the interval lengths and parameters of the affine functions using least-squares optimization, minimizing the squared difference between the original function and the approximation. Recall that E_{\min} denotes the minimum kinetic energy. Define the maximum kinetic energy

$$E_{\max} = \max_{k=1,2,\dots,N} (E_{\max}(k)) = \max_{k=1,2,\dots,N} \left(\frac{1}{2} v_{\max}^2(k) \right).$$

Then the nonlinear function $f(E) = \frac{1}{2\sqrt{2E}}$ can be approximated over the interval $[E_{\min}, E_{\max}]$ by an PWA function with 3 continuous affine subfunctions. However, the speed limit depends on the space interval, i.e., different space intervals may have different speed limit, which may be less than the overall maximum of the speed limit. Therefore, we adapt their coefficients of the PWA approximations depending on the space interval index k , i.e., we can have different PWA subfunctions for different space intervals within valid speed intervals. In this way the approximation error will be reduced. For example, if we consider an approximation using 3 affine subfunctions (cf. Figure 3.2), the PWA approximation⁴ of the

⁴The approximation error can be reduced by taking more regions.

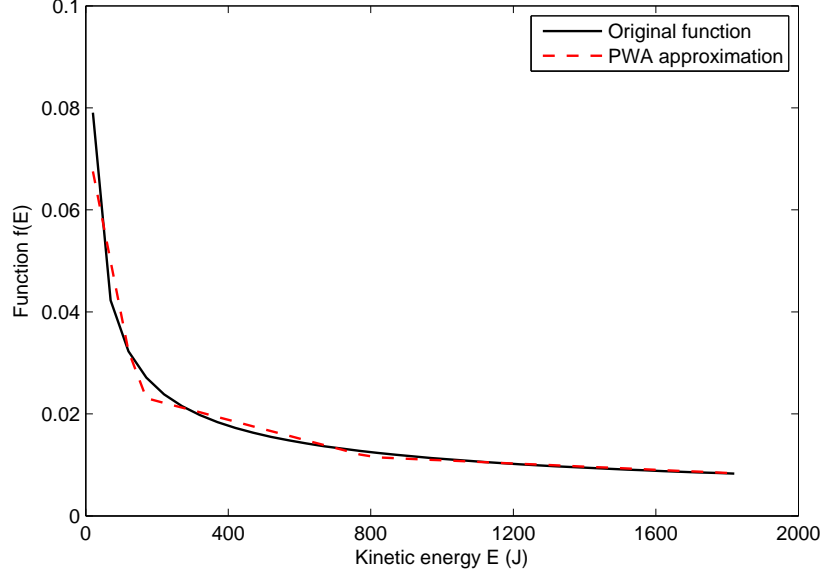


Figure 3.2: The PWA approximation of the nonlinear function $f(\cdot)$

nonlinear function $f(E(k)) = \frac{1}{2\sqrt{2E(k)}}$ can be written as

$$f_{\text{PWA}}(E(k)) = \begin{cases} \alpha_{1,k}E(k) + \beta_{1,k} & \text{for } E_{0,k} \leq E(k) \leq E_{1,k}, \\ \alpha_{2,k}E(k) + \beta_{2,k} & \text{for } E_{1,k} \leq E(k) \leq E_{2,k}, \\ \alpha_{3,k}E(k) + \beta_{3,k} & \text{for } E_{2,k} \leq E(k) \leq E_{3,k}, \end{cases} \quad (3.42)$$

with $E_{0,k} = E_{\min}$ and $E_{3,k} = E_{\max}(k)$ for the interval $[s_k, s_{k+1}]$. Furthermore, the values of $E_{1,k}$ and $E_{2,k}$ are determined by least-squares optimization.

Now the time dynamics (3.41) can be approximated as

$$t(k+1) = t(k) + (\alpha_{l,k}E(k) + \beta_{l,k} + \alpha_{m,k+1}E(k+1) + \beta_{m,k+1})\Delta s_k, \quad (3.43)$$

with $E_{l-1,k} \leq E(k) \leq E_{l,k}$, $E_{m-1,k+1} \leq E(k+1) \leq E_{m,k+1}$ for $l, m \in \{1, 2, 3\}$.

Furthermore, the maximum traction force u_{\max} is a nonlinear function of the velocity as given in (3.3) or (3.4) that can be reformulated as a nonlinear function of the kinetic energy. In a similar way as the approximation of the nonlinear function $f(\cdot)$, we can obtain an PWA approximation of the maximum traction force. If we consider an approximation using⁵ 3 affine subfunctions (cf. (3.42)), then the approximation can be written as

$$u_{\max, \text{PWA}}(E(k)) = \begin{cases} \lambda_{1,k}E(k) + \mu_{1,k} & \text{for } E_{4,k} \leq E(k) \leq E_{5,k}, \\ \lambda_{2,k}E(k) + \mu_{2,k} & \text{for } E_{5,k} \leq E(k) \leq E_{6,k}, \\ \lambda_{3,k}E(k) + \mu_{3,k} & \text{for } E_{6,k} \leq E(k) \leq E_{7,k}, \end{cases} \quad (3.44)$$

⁵For M affine subfunctions with $M > 3$ a similar procedure can be used.

with $E_{4,k} = E_{\min}$ and $E_{7,k} = E_{\max}(k)$ and where the values of $E_{5,k}$ and $E_{6,k}$ are decided by the approximation process.

The above PWA model with PWA constraints can be transformed into a mixed logical dynamic model by introducing some auxiliary logical variables [10]. First consider (3.42). In order to transform this equation, we introduce auxiliary logical variables $\delta_1(k)$ and $\delta_2(k)$, defined as

$$\begin{aligned} [E(k) \leq E_{1,k}] &\Leftrightarrow [\delta_1(k) = 1], \\ [E(k) \leq E_{2,k}] &\Leftrightarrow [\delta_2(k) = 1]. \end{aligned} \quad (3.45)$$

Then we get

$$\begin{aligned} f_{\text{PWA}}(E(k)) &= \delta_1(k)\delta_2(k)[\alpha_{1,k}E(k) + \beta_{1,k}] + (1 - \delta_1(k))\delta_2(k)[\alpha_{2,k}E(k) + \beta_{2,k}] \\ &\quad + (1 - \delta_1(k))(1 - \delta_2(k))[\alpha_{3,k}E(k) + \beta_{3,k}]. \end{aligned} \quad (3.46)$$

Since the maximum and minimum values of $E(k)$ are $E_{\max}(k)$ and E_{\min} , according to the transformation property (3.35), the logical conditions (3.45) can be rewritten as linear inequalities. Furthermore, an auxiliary logical variable $\delta_3(k)$ is introduced to replace the product $\delta_1(k)\delta_2(k)$. The condition $\delta_3(k) = \delta_1(k)\delta_2(k)$ can be rewritten as a system of linear inequalities according to (3.36). By defining new auxiliary variables $z_1(k) = \delta_1(k)E(k)$, $z_2(k) = \delta_2(k)E(k)$, and $z_3(k) = \delta_3(k)E(k)$, which can be expressed as a system of linear inequalities according to (3.37), the function $f_{\text{PWA}}(\cdot)$ can be rewritten as

$$\begin{aligned} f_{\text{PWA}}(E(k)) &= [-\alpha_{3,k} \quad \alpha_{2,k} - \alpha_{3,k} \quad \alpha_{1,k} - \alpha_{2,k} + \alpha_{3,k}] [z_1(k) \quad z_2(k) \quad z_3(k)]^T \\ &\quad + [-\beta_{3,k} \quad \beta_{2,k} - \beta_{3,k} \quad \beta_{1,k} - \beta_{2,k} + \beta_{3,k}] [\delta_1(k) \quad \delta_2(k) \quad \delta_3(k)]^T \\ &\quad + \alpha_{3,k}E(k) + \beta_{3,k}, \end{aligned} \quad (3.47)$$

In order to deal with the PWA constraints of the maximum traction force (cf. (3.44)), auxiliary logical variables $\delta_4(k)$ and $\delta_5(k)$ are introduced that are defined by

$$\begin{aligned} [E(k) \leq E_{5,k}] &\Leftrightarrow [\delta_4(k) = 1], \\ [E(k) \leq E_{6,k}] &\Leftrightarrow [\delta_5(k) = 1]. \end{aligned} \quad (3.48)$$

Similar to (3.45), the logical conditions (3.48) can be recast as linear inequalities by applying transformation property (3.35). In addition, another binary variable $\delta_6(k)$ is introduced similarly as $\delta_3(k)$, and it is defined as $\delta_6(k) = \delta_4(k)\delta_5(k)$. Furthermore, auxiliary variables $z_4(k) = \delta_4(k)E(k)$, $z_5(k) = \delta_5(k)E(k)$, and $z_6(k) = \delta_6(k)E(k)$ are defined in order to rewrite the constraints into a system of linear inequalities. The PWA constraints $u(k) \leq u_{\max, \text{PWA}}(E(k))$ can then be written as

$$\begin{aligned} u(k) &\leq [-\lambda_{3,k} \quad \lambda_{2,k} - \lambda_{3,k} \quad \lambda_{1,k} - \lambda_{2,k} + \lambda_{3,k}] [z_4(k) \quad z_5(k) \quad z_6(k)]^T \\ &\quad + [-\mu_{3,k} \quad \mu_{2,k} - \mu_{3,k} \quad \mu_{1,k} - \mu_{2,k} + \mu_{3,k}] [\delta_4(k) \quad \delta_5(k) \quad \delta_6(k)]^T + \lambda_{3,k}E(k) + \mu_{3,k}. \end{aligned} \quad (3.49)$$

Now the dynamics of the system can be rewritten as the following mixed logical dynamic model

$$x(k+1) = A_k x(k) + B_k u(k) + C_{1,k} \delta(k) + C_{2,k} \delta(k+1) + D_{1,k} z(k) + D_{2,k} z(k+1) + e_k, \quad (3.50)$$

where

$$\begin{aligned}
x(k) &= [E(k) \quad t(k)]^T, A_k = \begin{bmatrix} a_k & 0 \\ \Delta s_k(\alpha_{3,k} + a_k\alpha_{3,k+1}) & 1 \end{bmatrix}, B_k = \begin{bmatrix} b_k \\ \Delta s_k\alpha_{3,k+1}b_k \end{bmatrix}, \\
C_{1,k} &= \Delta s_k \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ -\beta_{3,k} & \beta_{2,k} - \beta_{3,k} & \beta_{1,k} - \beta_{2,k} + \beta_{3,k} & 0 & 0 & 0 \end{bmatrix}, \\
C_{2,k} &= \Delta s_k \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ -\beta_{3,k+1} & \beta_{2,k+1} - \beta_{3,k+1} & \beta_{1,k+1} - \beta_{2,k+1} + \beta_{3,k+1} & 0 & 0 & 0 \end{bmatrix}, \\
D_{1,k} &= \Delta s_k \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ -\alpha_{3,k} & \alpha_{2,k} - \alpha_{3,k} & \alpha_{1,k} - \alpha_{2,k} + \alpha_{3,k} & 0 & 0 & 0 \end{bmatrix}, \\
D_{2,k} &= \Delta s_k \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ -\alpha_{3,k+1} & \alpha_{2,k+1} - \alpha_{3,k+1} & \alpha_{1,k+1} - \alpha_{2,k+1} + \alpha_{3,k+1} & 0 & 0 & 0 \end{bmatrix}, \\
\text{and } e_k &= \begin{bmatrix} c_k \\ \Delta s_k(\alpha_{3,k+1}c_k + \beta_{3,k} + \beta_{3,k+1}) \end{bmatrix}.
\end{aligned}$$

The mixed logical dynamic model (3.50) is subject to the linear constraints of the form (3.35), (3.36), and (3.37) resulting from the transformation as well as the upper bound and lower bound constraints for $E(k)$, $t(k)$, and $u(k)$. All these constraints can be written more compactly as

$$R_{1,k}\delta(k) + R_{2,k}\delta(k+1) + R_{3,k}z(k) + R_{4,k}z(k+1) \leq R_{5,k}u(k) + R_{6,k}x(k) + R_{7,k}, \quad (3.51)$$

with appropriately defined coefficient matrices $R_{i,k}$, for $i = 1, 2, \dots, 7$.

The objective function (3.16) can be discretized as

$$J = \sum_{k=1}^N u(k)\Delta s_k + \sum_{k=1}^{N-1} \lambda |\Delta u(k)|, \quad (3.52)$$

where $\Delta u(k) = u(k+1) - u(k)$. We introduce a new variable $\omega(k)$ to deal with the absolute value of $\Delta u(k)$, and we add the linear inequalities:

$$\begin{aligned}
\omega(k) &\geq u(k+1) - u(k), \\
\omega(k) &\geq u(k) - u(k+1).
\end{aligned} \quad (3.53)$$

Since $\lambda > 0$, minimizing (3.52) is equivalent to minimizing

$$\tilde{J} = \sum_{k=1}^N u(k)\Delta s_k + \sum_{k=1}^{N-1} \lambda \omega(k). \quad (3.54)$$

subject to (3.53). Indeed, it is easy to verify that when we minimize the objective function (3.54) subject to (3.53), the optimal value of $\omega(k)$ will be equal to $|\Delta u(k)|$, so (3.52) will also be minimized.

The mixed integer linear programming problem

Now the optimal control problem can be recast as an MILP problem, where some of decision variables are binary (i.e., $\tilde{\delta}$) and some are real variables (i.e., $\tilde{u}, \tilde{\omega}, \tilde{z}$) with

$$\tilde{u} = \begin{bmatrix} u(1) \\ u(2) \\ \vdots \\ u(N) \end{bmatrix}, \quad \tilde{\delta} = \begin{bmatrix} \delta(1) \\ \delta(2) \\ \vdots \\ \delta(N+1) \end{bmatrix}, \quad \tilde{z} = \begin{bmatrix} z(1) \\ z(2) \\ \vdots \\ z(N+1) \end{bmatrix}, \quad \tilde{\omega} = \begin{bmatrix} \omega(1) \\ \omega(2) \\ \vdots \\ \omega(N-1) \end{bmatrix},$$

Furthermore, if we define $\tilde{V} = [\tilde{u}^T \ \tilde{\delta}^T \ \tilde{z}^T \ \tilde{\omega}^T]^T$, the equivalent formulation of the optimal control problem is obtained as follows:

$$\min_{\tilde{V}} C_J^T \tilde{V}, \quad (3.55)$$

subject to

$$F_1 \tilde{V} \leq F_2 x(1) + f_3 \quad (3.56)$$

$$F_4 \tilde{V} = F_5 x(1) + f_6 \quad (3.57)$$

where $C_J = [\Delta s_1 \ \cdots \ \Delta s_N \ 0 \ \cdots \ 0 \ \lambda \ \cdots \ \lambda]^T$. This can be shown as follows. The constraints for the MILP problem (3.51) are considered for $k = 1, 2, \dots, N$. We can substitute $x(k)$ in the constraints by using the state equation (3.50) recursively. The substituted form is obtained as the following expression:

$$\begin{aligned} x(k) = & \left[\prod_{j=1}^{k-1} A_j \right] x(1) + \sum_{i=1}^{k-1} \left[\prod_{j=i+1}^{k-1} A_j \right] B_i u(i) + \left[\prod_{j=2}^{k-1} A_j \right] C_{1,1} \delta(1) \\ & + \sum_{i=2}^{k-1} \left[\prod_{j=i+1}^{k-1} A_j \right] (A_i C_{i-1,2} + C_{i,1}) \delta(i) + C_{k-1,2} \delta(k) \\ & + \left[\prod_{j=2}^{k-1} A_j \right] D_{1,1} z(1) + \sum_{i=2}^{k-1} \left[\prod_{j=i+1}^{k-1} A_j \right] (A_i D_{i-1,2} + D_{i,1}) z(i) \\ & + D_{k-1,2} z(k) + \sum_{i=1}^{k-1} \left[\prod_{j=i+1}^{k-1} A_j \right] e_i. \end{aligned}$$

In addition, the end point condition $x(N+1) = [E_{\text{end}} \ T]^T$ needs to be considered in (3.57). Because we know the value of $x(N+1)$, the values of α_m and β_m in (3.43) are also known. So the state equation at the end point can be written as

$$x(N+1) = A_N x(N) + B_N u(N) + C_{1,N} \delta(N) + D_{1,N} z(N) + e_N$$

where $A_N = \begin{bmatrix} a_N & 0 \\ \Delta s_N(\alpha_{3,N} + \alpha_{m,N+1} a_N) & 1 \end{bmatrix}$, $B_N = \begin{bmatrix} b_N \\ \Delta s_N \alpha_{m,N+1} b_N \end{bmatrix}$, and

$e_N = \begin{bmatrix} c_N \\ \Delta s_N(\alpha_{m,N+1} c_N + \beta_{m,N+1} + \beta_{3,N}) \end{bmatrix}$. By properly defining F_1, F_2, f_3, F_4, F_5 , and f_6 , we can write all these constraints in the form (3.56) and (3.57). The MILP problem (3.55)-(3.57) can be solved by several existing commercial and free solvers, such as CPLEX, Xpress-MP, GLPK (see e.g., [5, 86]).

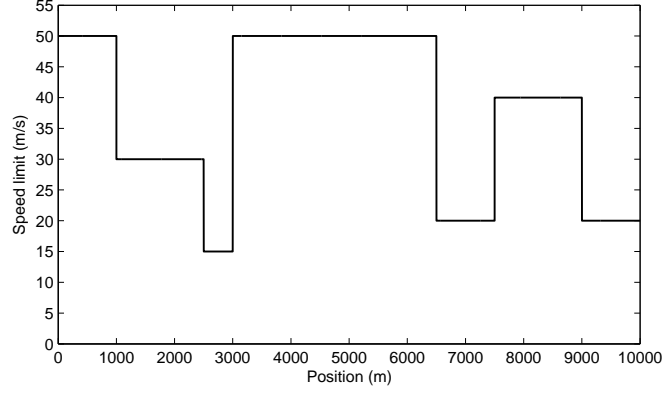


Figure 3.3: The speed limits along the track

Remark 3.5 The variable $t(s_{k_i})$ in (3.21) and (3.22) is one of the state variables of the model (3.50) - (3.51). Note that (3.21) and (3.22) are linear constraints. Hence, we still have an MILP problem if the target point and target window constraints are included. \square

3.5 Case study

In order to demonstrate the performance of the pseudospectral method and the MILP approach, we use a case study (inspired by Vařak et al. [123]) to compare these two approaches to the discrete dynamic programming (DDP) approach proposed in [39]. The reason for selecting the DDP approach for comparison is that [39] concludes that the performance of the DDP approach is better than that of the sequential quadratic programming approach and the coasting strategy obtained by the maximum principle. The optimal trajectories obtained by those the pseudospectral, MILP approach, and DDP approaches are compared with each other. In addition, both the computation time and the performance with respect to the optimization objective and constraints violations of those approaches are analyzed.

3.5.1 Set-up

The case study in this chapter is inspired by that of [123], where the track length between the departure station and arrival station is 10 km. In [123], there were no speed limit and grade profile. We add them as shown in Figures 3.3 and 3.4. The rolling stock includes an SBB Re 460 locomotive [39, 46, 108], the parameters of which are shown in Table 3.1. The rotating mass factor is often chosen as 1.06 in the literature [57] and therefore we also adopt this value. According to the assumption made in Section 3.3, the unit kinetic energy should be larger than zero. In this test case, the minimum kinetic energy is chosen as 0.1 J. The maximum traction force of the SBB Re 460 locomotive is a nonlinear function of the train's velocity and the maximum value of this function is 300 kN as shown in Figure 3.1. The objective function of the optimal train control problem considered here is a weighted sum of the energy consumption and passenger comfort, where the weight λ is taken as 500.

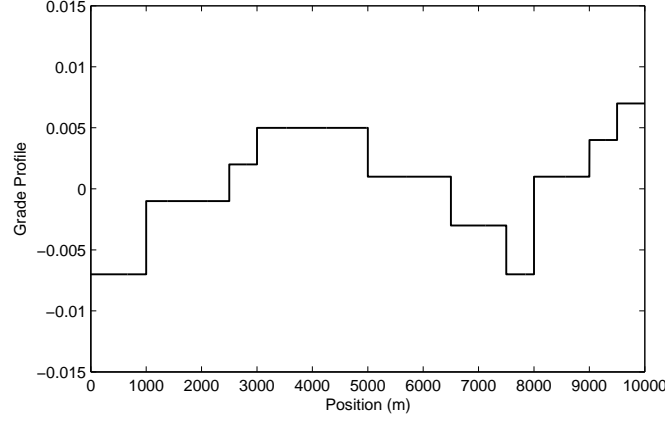


Figure 3.4: The grade profile of the track

Table 3.1: Parameters of the train and the line path

Property	Symbol	Value
Train mass [kg]	m	$5.07 \cdot 10^5$
Basic resistance [N/kg]	R_b	$0.014 + 2.564 \cdot 10^{-5} v^2$
Mass factor [-]	ρ	1.06
Line length [m]	s_T	10^4
Minimum kinetic energy [J]	E_{\min}	0.1
Maximum braking force (regular) [N]	u_{\min}	$-4.475 \cdot 10^5$

The total running time for this trip is given by the timetable or the rescheduling process. Here, the total running time is 450 s, which consists of the minimum running time plus 5% running time supplements.

Two cases will be considered here:

- Case A: the maximum traction force is constant.
- Case B: the maximum traction force is a nonlinear function of the velocity.

In Case A, just like the case study in [123], we assume that the maximum traction force E_{\max} is constant: $E_{\max} = 300$ kN. First, the optimal trajectory planning problem is solved using PSOPT [8], which implements a pseudospectral method. In this case study, the problem is solved using the Legendre pseudospectral discretizations, with local automatic mesh refinement, starting with 40 nodes.

Second, the problem is solved using the MILP approach. Since the maximum traction force is constant, in this case the linear constraints caused by the PWA constraints (3.44) of the maximum traction force will not be considered here. In this chapter, the PWA approximations of the nonlinear function $f(E) = \frac{1}{2\sqrt{2E}}$ may depend on the space interval index k as stated in Section 3.4.2, i.e., we can have different PWA subfunctions for different space intervals. In Figure 3.3 there are five speed limits, i.e., 15 m/s, 20 m/s, 30 m/s, 40 m/s, and

Table 3.2: The PWA approximations of the nonlinear function $f(\cdot)$

Approx. no.	Segment m	α_m [(s/m) ³]	β_m [s/m]	$E_m - E_{m+1}$ [(m/s) ²]
Approx. 1	Segment 1	$-5.0943 \cdot 10^{-4}$	0.0767	0.1 – 71.2
	Segment 2	$-1.7393 \cdot 10^{-4}$	0.0528	71.2 – 112.5
Approx. 2	Segment 1	$-3.1153 \cdot 10^{-4}$	0.0665	0.1 – 115
	Segment 2	$-6.7188 \cdot 10^{-5}$	0.0384	115 – 200
Approx. 3	Segment 1	$-9.4977 \cdot 10^{-5}$	0.0443	0.1 – 240
	Segment 2	$-2.3470 \cdot 10^{-5}$	0.0272	240 – 450
Approx. 4	Segment 1	$-4.4240 \cdot 10^{-5}$	0.0346	0.1 – 415
	Segment 2	$-9.6462 \cdot 10^{-6}$	0.0202	415 – 800
Approx. 5	Segment 1	$-1.8122 \cdot 10^{-5}$	0.0251	0.1 – 640
	Segment 2	$-6.2127 \cdot 10^{-6}$	0.0175	640 – 1250

Table 3.3: PWA approximations of the nonlinear function $f(\cdot)$ for the first and the last space intervals

Approx. no.	Segment m	α_m [(s/m) ³]	β_m [s/m]	$E_m - E_{m+1}$ [(m/s) ²]
Approx. 6	Segment 1	$-4.6463 \cdot 10^{-4}$	0.0734	0.1 – 80.8
	Segment 2	$-4.6463 \cdot 10^{-4}$	0.0734	80.8 – 312.5
Approx. 7	Segment 1	$-1.4458 \cdot 10^{-4}$	0.0534	0.1 – 229.9
	Segment 2	$-1.4514 \cdot 10^{-6}$	0.0235	229.9 – 450

50 m/s. Therefore, five different approximations with 2 subfunctions of $f(\cdot)$ are obtained, the parameters of which are given in Table 3.2. For each space interval, an appropriate PWA approximation can be chosen based on the given speed limit. In addition, we introduce two additional PWA approximations of $f(\cdot)$ for the first space interval $[s_{\text{start}}, s_2]$ and the last space interval $[s_N, s_{\text{end}}]$. The train speed in these intervals will usually be low; hence, in order to obtain a good fit we apply a weighted least-squares optimization to optimize the coefficients of the PWA approximations for these intervals, where the weight function should have a high value in the low-speed range. The parameters of the PWA approximations for the first and the last space intervals are given in Table 3.3.

The length Δs_k for the interval $[s_k, s_{k+1}]$ depends on the speed limits, gradient profile, tunnels, and so on. In addition, if the number of space intervals N is larger, then the computation time of the MILP approach will be longer, but the accuracy will be better. According to the speed limits and grade profile given in Figure 3.3 and Figure 3.4, the length of each interval is chosen to equal 500 m, i.e. $\Delta s_k = 500$ m for $k = 1, 2, \dots, 20$, which provides a good balance between the computation time and the accuracy. As MILP solver, we use CPLEX, implemented through the cplex interface function of the Matlab Tomlab toolbox.

Table 3.4: The coefficients of the varying maximum traction force

Segment j	$c_{1,j}$ [kg · m/s ²]	$c_{2,j}$ [kg/s]	$c_{3,j}$ [kg/m]	$v_j - v_{j+1}$ [m/s]
1	$3.000 \cdot 10^5$	$-1.125 \cdot 10^3$	0	0-22.22
2	$7.263 \cdot 10^5$	$-2.726 \cdot 10^4$	$3.128 \cdot 10^2$	22.22-38.89
3	$4.237 \cdot 10^5$	$-1.120 \cdot 10^4$	$1.000 \cdot 10^2$	38.89-50

Table 3.5: The coefficients of the PWA approximation of maximum traction force

Segment m	λ_m [kg/m]	μ_m [kg · m/s ²]	$E_m - E_{m+1}$ [(m/s) ²]
1	$-2.9396 \cdot 10^2$	$4.1992 \cdot 10^5$	0.1 – 500
2	$-0.9637 \cdot 10^2$	$3.2112 \cdot 10^5$	500 – 1250

For the DDP approach, the continuous nonlinear model of train (3.14)-(3.15) is discretized in space. The number of the space intervals is 100 and the length of each space interval is 100 m. To compute the optimal trajectory with DDP, we use a matlab function for dynamic programming that was introduced in [113].

For Case B, we consider a varying maximum traction force as shown in Figure 3.1, the coefficients of which according to (3.3) are based on [39, 46, 108] and listed here in Table 3.4. In PSOPT, non-smooth path constraints can be handled by introducing a smooth version of the Heaviside function (see Section 3.4.1). In the MILP approach we need to approximate the nonlinear maximum traction force by PWA functions in (3.44), where the coefficients may also depend on the space interval index k . Here, for simplicity, we just use one PWA approximation with two affine subfunctions for all k . The parameters of the PWA function are listed in Table 3.5.

3.5.2 Results and discussion

Results for Case A: the maximum traction force is constant

The optimal solution of the pseudospectral method using PSOPT, which is obtained after 7 mesh refinement iterations, has 179 nodes. The calculation time for PSOPT is 6 min and 10 s on a 1.8 GHz Intel Core2 Duo CPU running a 64-bit Linux operating system and the computation time for DDP is 2 min and 8 s with 100 space intervals as shown in Table 3.6. However, the calculation time for the MILP approach is 0.32 s on the same CPU and operation system as above, which is much shorter than the calculation time of PSOPT and DDP.

Figure 3.5 shows the optimal control inputs with constant maximum traction force, where the dotted line, the solid line, and the dashed line represent the results calculated by PSOPT, MILP, and DDP, respectively. It can be seen from Figure 3.5 that the results obtained by these three approaches show a similar trend, but there exist more discrete changes but with a smaller magnitude in the control signals of PSOPT and DDP. This is mainly caused by the larger number of space intervals: there are 178 and 100 space intervals in PSOPT and DDP, respectively, but in the MILP approach, there are just 20 space intervals.

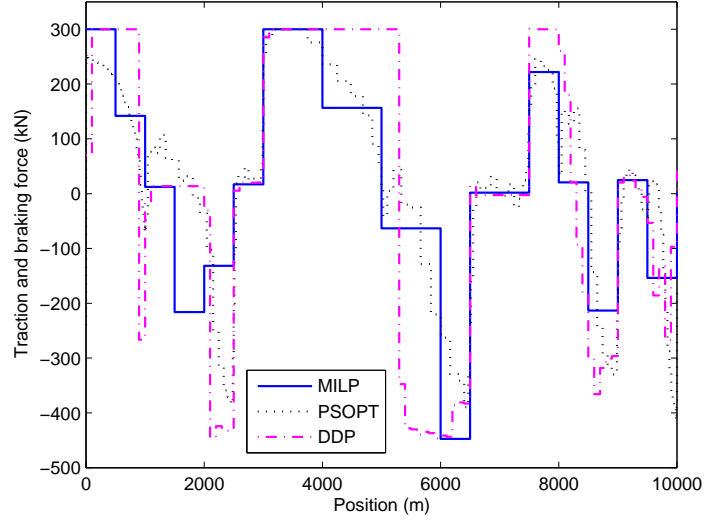


Figure 3.5: The optimal control inputs with constant maximum traction force for the following approaches: MILP, PSOPT, and DDP

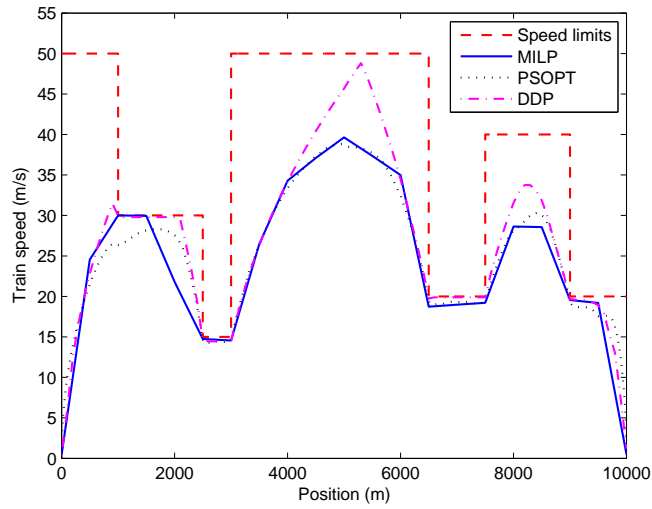


Figure 3.6: The trajectories generated by the nonlinear continuous-time train model (3.1)-(3.2) using optimal control inputs with constant maximum traction force for the following approaches: MILP, PSOPT, and DDP

Table 3.6: Performance comparison of PSOPT, MILP, and DDP for Case A with constant maximum traction force

	PSOPT	MILP	DDP
J_{\min} [-]	$2.424 \cdot 10^8$	$2.696 \cdot 10^8$	$2.482 \cdot 10^8$
CPU time [s]	370	0.32	128
End position s_{end} violation [m]	0	0	0
End kinetic energy E_{end} violation [m/s ²]	0.1	0.005	0.596
End time T_{end} violation [s]	0.496	9.560	6.049
Speed limit violation	No	No	No

The optimal control inputs calculated by these three approaches are applied to the nonlinear continuous-time train model (3.1)-(3.2). The differential equation of the nonlinear model is solved numerically using a variable step Runge-Kutta method and the resulting speed-position trajectories for the train are shown in Figure 3.6. The dashed line shows the speed limit for the trip, which is determined by the characteristics of the train, line, etc. The dotted line, the solid line, and the dash-dotted line show the optimal trajectories obtained using control inputs generated by PSOPT, MILP, and DDP, respectively. It can be observed that these optimal trajectories are below the speed limit, which means that the speed constraints are satisfied, i.e., there is no speed limit violation. In addition, we can see from Figure 3.6 that the optimal trajectories obtained using control inputs generated by PSOPT and DDP are smoother than the one obtained with the MILP approach, which is mainly caused by the number of space intervals as stated before.

In Table 3.6, the values of the objective function, the computation time, and the constraints violations (i.e. speed limit violation, end position violation, end kinetic energy violation, and end time violation) are compared for the control inputs generated by PSOPT, MILP and DDP applied to the nonlinear continuous-time train model (3.1)-(3.2). The values of the objective function obtained by the PSOPT, the MILP, and the DDP approach are 2.424×10^8 and 2.696×10^8 , and 2.482×10^8 , respectively. The relative differences of the MILP and DDP control performance are 11.2% and 2.4% of that of the pseudospectral method. Therefore, the pseudospectral approach yields the smallest objective value and the constraints violations for the pseudospectral method are also small.

Results for Case B: the maximum traction force is a nonlinear function

Figure 3.7 shows the optimal control inputs for Case B. The dotted line, the solid line, and the dashed line in Figure 3.7 represent the optimal control inputs obtained using PSOPT, MILP, and DDP, respectively. When we compare Figure 3.7 to Figure 3.5, the maximum traction force in Figure 3.7 is no longer equal to 300 kN for the MILP approach in the space interval [3000,4000], but it becomes smaller and smaller when the speed grows. This is caused by the varying maximum traction force, which is decreasing when the speed goes up. Similar results can be observed for the optimal inputs calculated by PSOPT and DDP. Figure 3.8 shows the speed-position trajectories for the train under varying maximum traction force constraints when applying these inputs to the nonlinear train model (3.1)-(3.2). The dashed line, the dotted line, the solid line, and the dash-dotted line show the speed lim-

Table 3.7: Performance comparison of PSOPT, MILP, and DDP for Case B with varying maximum traction force

	PSOPT	MILP	DDP
J_{\min} [-]	$2.625 \cdot 10^8$	$2.819 \cdot 10^8$	$2.683 \cdot 10^8$
CPU time [s]	1147	0.54	134
End position s_{end} violation [m]	0	0	0
End kinetic energy E_{end} violation [m/s^2]	0.1	0.005	0.0328
End time T_{end} violation [s]	0	4.170	5.404
Speed limit violation	No	No	No

its and the trajectories obtained using control inputs generated by PSOPT, MILP, and DDP, respectively. These trajectories are still below the speed limit, so there is no speed limit violation. In addition, we can see from Figure 3.8 that the slopes of these three optimal trajectories obtained in the space interval $[3500, 5000]$ are smaller than those of Figure 3.6, because the maximum traction force is becoming smaller with the increase of the train's speed.

The values of the objective function, the computation time, and the constraints violations are compared for PSOPT, MILP, and DDP in Table 3.7. Similar as the results in Case A, the pseudospectral approach obtains the minimum objective function value $2.625 \cdot 10^8$, which is higher than that in Table 3.6 (this is due to the inclusion of the constraint of the varying maximum traction force). The relative differences of the MILP and DDP approach in control performance are 7.4% and 2.2% when compared to that of the pseudospectral method. PSOPT includes a local automatic mesh refinement. In this case study, we take 40 nodes as initial value. The final solution is obtained after 7 mesh refinement iterations and has 199 nodes. The calculation time is 19 min and 7 s. Compared with the problem in Case A, 20 nodes are added and the computation time is almost 13 min longer. For the DDP approach, the computation time is 2 min and 14 s, which is 6 s longer than that of Case A. In the MILP approach, for each space interval an extra binary variable and an auxiliary real variable are introduced in the mixed logical dynamic model compared with Case A, since the maximum traction force is considered as a nonlinear function that is approximated by an PWA approximation with 2 subfunctions. Therefore, 40 variables are added to the MILP problem since the number of the space intervals is 20. The computation time is now 0.54 s, which is larger than the 0.32 s of Case A, but it still is much lower than the computation time the pseudospectral method and the DDP approach. Similar to the results shown in Table 3.6, there are no speed limit violations and the end kinetic energy violation is very small. Furthermore, the end time violation for the pseudospectral method is also very small, but for the MILP and DDP approach this violation is about 1% of the total running time.

Discussion

It is concluded that for the given case study the pseudospectral approach obtains the best control performance, which considers the value of the objective function and the constraints violations. However, when the computation time is also taken into consideration, the MILP approach yields the best overall performance.

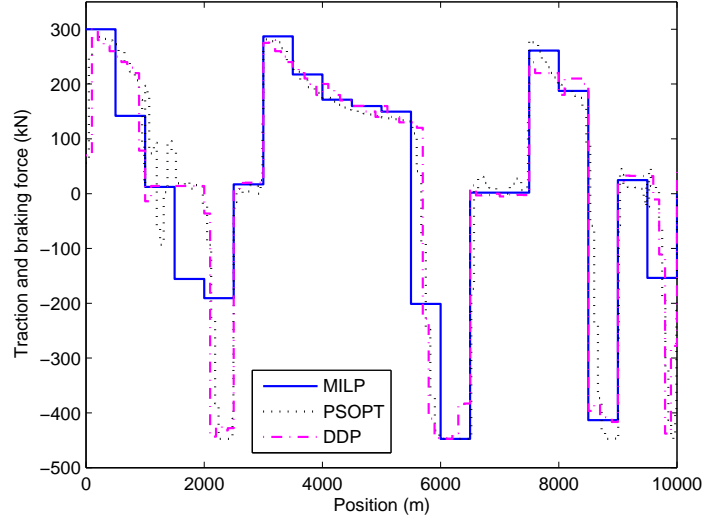


Figure 3.7: The optimal control inputs with varying maximum traction force for the following approaches: MILP, PSOPT, and DDP

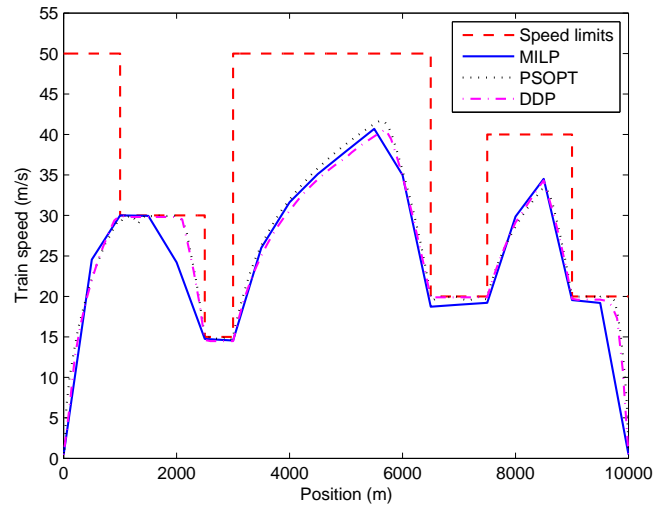


Figure 3.8: The trajectories generated by the nonlinear continuous-time train model (3.1)-(3.2) using optimal control inputs with varying maximum traction force for the following approaches: MILP, PSOPT, and DDP

It is worth to note that we apply a trapezoidal integration rule to approximate the time differential equation (3.15) and then use PWA functions to approximate the nonlinear function (3.41) in the MILP approach. Therefore, the end time violation in Table 3.6 of the MILP approach is probably caused by these approximations. Furthermore, we can make the error even smaller by adjusting the PWA approximations according to Footnote 4 on page 31, but then the CPU time goes up.

3.6 Conclusions

In this chapter, the optimal trajectory planning problem for a single train has been considered. We have proposed two approaches to solve this problem: the pseudospectral method and the mixed integer linear programming (MILP) based approach. In the pseudospectral method, the optimal trajectory planning problem is formulated as a multiple-phase optimal control problem based on piecewise line resistance and speed limits. The constraints caused by the varying maximum traction force are defined as nonlinear path constraints. In the MILP approach, the nonlinear train operation model is formulated as a mixed logical dynamic model by using piecewise affine approximations. The variable line resistance (including variable grade profile, tunnels, curves) and speed restrictions are included in the constraints of the mixed logical dynamic model. Furthermore, the optimal control problem is recast as an MILP problem. The case study shows that the pseudospectral method has the best control performance and the MILP has the best overall performance if the computation time is also taken into account. In addition, the computation time of the MILP approach is much shorter compared with that of the pseudospectral method and the discrete dynamic programming approach. For the given case study, the relative difference between the performance of the MILP approach and that of the pseudospectral approach is about 10%.

When the timetable is known, the two approaches proposed in this chapter (i.e., the MILP and the pseudospectral approach) can be applied to calculate the optimal trajectory for trains between stations to save energy and to ensure passenger comfort. If there are some disturbances in the network, then one could use a rescheduling approach to reorder trains and determine new timetables [29, 57]. Next, the affected trains have to optimize their trajectories according to the new timetable. In this case, the trajectory planning problem needs to be solved quickly to satisfy the real-time requirements; so then the MILP approach could be applied since it gives the best trade-off between computational speed and accuracy.

An extensive comparison and assessment of the pseudospectral method, the MILP approach, and other approaches in the literature for various case studies and a wide range of scenarios will be a topic for future work. In addition, in this chapter we have focused on the trajectory planning for a single train between two stations with the assumption that the constraints and disturbances caused by signaling systems and other trains are handled by the lower control level. However, in practice these constraints and disturbances are significant for the capacity of the railway network, and therefore some interesting conflict detection and resolution approaches have been proposed to manage these constraints and disturbances during the rescheduling phases [24, 27, 28]. In future work, one could combine these conflict detection and resolution approaches with the trajectory planning approaches proposed here to solve the trajectory planning for multiple trains. There one could also use the MILP ap-

proach (including hierarchical and distributed optimization if the problem grows too large). Furthermore, the pseudospectral and MILP solvers used in this chapter are general-purpose solvers. By making use of the specific structure and properties of the optimal trajectory planning problem, significant speed-ups can be expected. Therefore, tailored pseudospectral and MILP solvers for the optimal trajectory planning problem for trains could be developed.

3.A A general formulation of the pseudospectral method

The optimal train trajectory planning problem of Chapter 3 can be formulated as a multiple phase optimal control problem and then can be solved by the pseudospectral method. Below we describe a general optimal control problem with multiple phases and the solution procedure of the general optimal control problem using the pseudospectral method. This explanation is based on [9, 19, 37, 51, 106].

3.A.1 The multiple-phase optimal control problem

The general optimal control problem with N_p phases is formulated as follows [106]. The objective function is

$$J = \sum_{i=1}^{N_p} \left(\varphi^{(i)}(x^{(i)}(t_f^{(i)}), p^{(i)}, t_f^{(i)}) + \int_{t_0^{(i)}}^{t_f^{(i)}} L^{(i)}(x^{(i)}(t), u^{(i)}(t), p^{(i)}, t) dt \right), \quad (3.58)$$

where $[t_0^{(i)}, t_f^{(i)}]$ is the time interval for the i th phase, $u^{(i)}(\cdot)$ and $x^{(i)}(\cdot)$ are the control trajectories and state trajectories, $p^{(i)}$ are the static parameters, for $i = 1, 2, \dots, N_p$. The objective function (3.58) is subject to the differential constraints

$$\dot{x}^{(i)}(t) = f^{(i)}(x^{(i)}(t), u^{(i)}(t), p^{(i)}, t), \quad t \in [t_0^{(i)}, t_f^{(i)}], \quad (3.59)$$

the path constraints

$$h_L^{(i)} \leq h^{(i)}(x^{(i)}(t), u^{(i)}(t), p^{(i)}, t) \leq h_U^{(i)}, \quad t \in [t_0^{(i)}, t_f^{(i)}], \quad (3.60)$$

the event constraints

$$e_L^{(i)} \leq e^{(i)}(x^{(i)}(t_0^{(i)}), u^{(i)}(t_0^{(i)}), x^{(i)}(t_f^{(i)}), u^{(i)}(t_f^{(i)}), p^{(i)}, t_0^{(i)}, t_f^{(i)}) \leq e_U^{(i)}, \quad t \in [t_0^{(i)}, t_f^{(i)}], \quad (3.61)$$

the linkage constraints

$$\begin{aligned} \Psi_L &\leq \Psi(x^{(1)}(t_0^{(1)}), u^{(1)}(t_0^{(1)}), x^{(1)}(t_f^{(1)}), u^{(1)}(t_f^{(1)}), p^{(1)}, t_0^{(1)}, t_f^{(1)}, \\ &\quad x^{(2)}(t_0^{(2)}), u^{(2)}(t_0^{(2)}), x^{(2)}(t_f^{(2)}), u^{(2)}(t_f^{(2)}), p^{(2)}, t_0^{(2)}, t_f^{(2)}, \\ &\quad \dots \\ &\quad x^{(N_p)}(t_0^{(N_p)}), u^{(N_p)}(t_0^{(N_p)}), x^{(N_p)}(t_f^{(N_p)}), u^{(N_p)}(t_f^{(N_p)}), p^{(N_p)}, t_0^{(N_p)}, t_f^{(N_p)}) \leq \Psi_U, \end{aligned} \quad (3.62)$$

the bound constraints

$$\begin{aligned} u_L^{(i)} &\leq u^{(i)}(t) \leq u_U^{(i)}, \quad x_L^{(i)} \leq x^{(i)}(t) \leq x_U^{(i)}, \quad t \in [t_0^{(i)}, t_f^{(i)}], \\ p_L^{(i)} &\leq p^{(i)} \leq p_U^{(i)}, \quad t_{L0}^{(i)} \leq t_0^{(i)} \leq t_{U0}^{(i)}, \quad t_{Lf}^{(i)} \leq t_f^{(i)} \leq t_{Uf}^{(i)}, \end{aligned} \quad (3.63)$$

and the time constraints

$$t_f^{(i)} - t_0^{(i)} \geq 0. \quad (3.64)$$

3.A.2 The solution process of the optimal control problem

Let $i \in \{1, 2, \dots, N_p\}$ be a particular phase of the optimal control problem (3.58)-(3.64) and let $(\cdot)^{(i)}$ denote information for the i th phase. The i th phase of the optimal control problem can be transformed from the interval $t \in [t_0^{(i)}, t_f^{(i)}]$ to the interval $\tau \in [-1, 1]$ for $i = 1, 2, \dots, N_p$ by introducing the following transformation [106]:

$$\tau = \frac{2}{t_f^{(i)} - t_0^{(i)}} t - \frac{t_f^{(i)} + t_0^{(i)}}{t_f^{(i)} - t_0^{(i)}}. \quad (3.65)$$

Now we approximate the state and control functions using Legendre pseudospectral approximation. The state $x_k^{(i)}(\tau)$, $\tau \in [-1, 1]$ is approximated by the N_i th order Lagrange polynomial $\bar{x}_k^{N_i, (i)}(\tau)$ based on interpolation at the Legendre-Gauss-Lobatto (LGL) points [19]:

$$x_k^{(i)}(\tau) \approx \bar{x}_k^{N_i, (i)}(\tau) = \sum_{n=0}^{N_i} \bar{x}_k^{N_i, (i)}(\tau_n) \phi_n^{(i)}(\tau), \quad (3.66)$$

where $\bar{x}_k^{N_i, (i)}(\tau_n)$ is a discrete approximation of the LGL point τ_n and the Lagrange basis polynomials $\phi_n^{(i)}(\tau)$ for $n = 0, 1, \dots, N_i$ are defined as

$$\phi_n^{(i)}(\tau) = \prod_{m=0, m \neq n}^{N_i} \frac{\tau - \tau_m^{(i)}}{\tau_n^{(i)} - \tau_m^{(i)}}, \quad (3.67)$$

and $\tau_n^{(i)}$ for $n = 0, 1, \dots, N_i$ are the LGL points, which are defined as $\tau_0^{(i)} = -1$, $\tau_{N_i}^{(i)} = 1$, and τ_n for $n = 1, 2, \dots, N_i - 1$ being the subsequent roots of the derivative of the Legendre polynomial

$$L_{N_i}(\tau) = \frac{1}{2^{N_i} N_i!} \frac{d^{N_i}}{d\tau^{N_i}} (\tau^2 - 1)^{N_i},$$

in the interval $[-1, 1]$. The control $u^{(i)}(\tau)$ can be approximated in a similar way. The derivative of $\bar{x}_k^{N_i, (i)}(\tau)$ at the LGL points τ_n can be obtained by differentiating (3.66), which can be expressed as a matrix multiplication as follows:

$$\dot{x}_k^{(i)}(\tau_n) \approx \dot{\bar{x}}_k^{N_i, (i)}(\tau_n) = \sum_{j=0}^{N_i} \tilde{D}_{nj}^{(i)} \bar{x}_k^{N_i, (i)}(\tau_j), \quad (3.68)$$

where $\tilde{D}^{(i)}$ is the $(N_i + 1) \times (N_i + 1)$ differential approximation matrix [50] given by

$$\tilde{D}_{nj}^{(i)} = \begin{cases} \frac{\phi_{N_i}^{(i)}(\tau_n)}{\phi_{N_i}^{(i)}(\tau_j)} \frac{1}{\tau_n - \tau_j}, & \text{if } n \neq j, \\ -N_i(N_i + 1)/4, & \text{if } n = j = 0, \\ N_i(N_i + 1)/4, & \text{if } n = j = N_i, \\ 0, & \text{otherwise.} \end{cases} \quad (3.69)$$

The differential constraints can be recast into algebraic constraints via the differential approximation matrix. In addition, the path constraints (3.60) can be discretized at the LGL points. Note that the dynamic constraints and path constraints are only considered at the LGL points, which means both the dynamic and path constraints might be violated in between the LGL points [107]. The objective function (3.58) can be approximated using the LGL points as

$$J = \sum_{i=1}^{N_p} \left(\phi^{(i)} \left(\bar{x}^{N_i, (i)}(-1), \bar{x}^{N_i, (i)}(1), p^{(i)}, t_0^{(i)}, t_f^{(i)} \right) + \frac{t_f^{(i)} - t_0^{(i)}}{2} \sum_{n=0}^{N_i} L^{(i)} \left(\bar{x}^{N_i, (i)}(\tau_n), \bar{u}^{N_i, (i)}(\tau_n), p^{(i)}, \tau_n \right) \omega_n \right), \quad (3.70)$$

where ω_n are weights given by

$$\omega_n = \frac{2}{N(N+1)} \frac{1}{(L_{N_i}(\tau_n))^2}, \quad \text{for } n = 0, 1, \dots, N_i. \quad (3.71)$$

If we include all the decision variables in vector y , the optimal control problem can then be expressed as a nonlinear programming problem:

$$\min_y J(y) \quad (3.72)$$

subject to

$$\begin{aligned} G_L &\leq G(y) \leq G_U \\ y_L &\leq y \leq y_U. \end{aligned} \quad (3.73)$$

By defining $G(y)$, G_L , G_U , y_L , and y_U properly, we can write all constraints in the form (3.73).

There exist several commercial and free packages that implement the pseudospectral method: SOCS [13] and DIRCOL [112] are Fortran-based proprietary packages, while PROPT [107] and DIDO [104] are commercial software packages with Matlab interface. A Matlab-based open source tool that uses the Gauss pseudospectral method is GPOPS [102]. PSOPT is an open source optimal control package written in C++, including Legendre and Chebyshev pseudospectral discretizations [9]. These software packages start with a general optimal problem formulated as (3.58)-(3.64), then transform this problem into a nonlinear programming problem with objective function (3.72) and constraints (3.73), and finally solve it possibly using an NLP solver, such as SNOPT [49].

Chapter 4

Optimal Trajectory Planning for Multiple Trains

In this chapter, the optimal trajectory planning problem for multiple trains under fixed block signaling systems and moving block signaling systems is considered. Two solution approaches are proposed to solve this optimal control problem for multiple trains: the greedy approach and the simultaneous approach. The greedy approach optimizes the trajectories of trains sequentially, where first the trajectory of the leading train is optimized and then the trajectory of the following train is optimized based on the trajectory of the leading train. In the simultaneous approach, the trajectories of all the trains in the problem formulation are optimized at the same time. In each approach, the trajectory planning problem is similar to the problem of Chapter 3, and therefore it can also be solved using the pseudospectral method and the mixed integer linear programming (MILP) approach. The performance of the proposed approaches is compared via a case study.

This chapter is based on [134] and is supported by the results presented in [129, 131].

4.1 Introduction

The approaches proposed in Chapter 3 ignore the impact caused by signaling systems, e.g., a fixed block signaling (FBS) systems or a moving block signaling (MBS) systems (see Section 2.1.2 for more detailed information about signaling systems). In this chapter, the constraints caused by the leading train in an FBS system and an MBS system are included into the trajectory planning problem. For the MILP approach, the constraints caused by signaling systems are discretized and then recast as linear constraints by piecewise affine (PWA) approximations. Hence, these constraints can easily be included into the MILP formulation.

The remainder of this chapter is structured as follows. In Section 4.2, the train model is summarized based on Section 3.2.1. In addition, the constraints for the operation of trains in an FBS system and an MBS system are also formulated in this section. Section 4.3 gives the mathematical formulation of the trajectory planning problem for multiple trains. In Section 4.4, we propose two solution schemes to solve the trajectory planning problem for

multiple trains. For the MILP approach, Section 4.5 presents the transformation process of the FBS constraints and the MBS constraints. In addition, the mode vector constraints are proposed to reduce the computation time of the MILP approach. Section 4.6 illustrates the performance of the proposed approaches via a case study. Conclusions are finally presented in Section 4.7.

4.2 Model formulation

In this section, the formulation of operation of trains in the FBS system and in the MBS system are presented.

4.2.1 Train dynamics

Although the model for the operation of trains has been described in the previous chapter, we here repeat the main equations that will be referred to in the remainder of this chapter. As shown in Chapter 3, the continuous-space model (3.14) and (3.15) of train operations can be discretized in space: the position horizon between two consecutive stations $[s_{\text{start}}, s_{\text{end}}]$ is split into N intervals and it is assumed that the track and train parameters as well as the traction or the braking force can be considered as constant in each interval $[s_k, s_{k+1}]$ with length $\Delta s_k = s_{k+1} - s_k$, for $k = 1, 2, \dots, N$. The discrete-space model is then transcribed into an PWA model by approximating the nonlinear terms through PWA functions. Furthermore, by applying the transformation properties described in Section 3.4.2, the PWA model is formulated as the following mixed logical dynamic model:

$$x(k+1) = A_k x(k) + B_k u(k) + C_{1,k} \delta(k) + C_{2,k} \delta(k+1) + D_{1,k} z(k) + D_{2,k} z(k+1) + e_k, \quad (4.1)$$

$$R_{1,k} \delta(k) + R_{2,k} \delta(k+1) + R_{3,k} z(k) + R_{4,k} z(k+1) \leq R_{5,k} u(k) + R_{6,k} x(k) + R_{7,k}, \quad (4.2)$$

where $x(k) = [E(k) \quad t(k)]^T$, $\delta(\cdot)$ and $z(\cdot)$ are the binary variables and auxiliary variables introduced by the transformation, and (4.2) also includes the upper bounds and lower bounds constraints for $E(k)$, $t(k)$, and $u(k)$. The coefficient matrices in the mixed logical dynamic model are determined by the train model, the PWA approximations, upper bounds and lower bounds constraints, etc.

4.2.2 Operation of trains in a fixed block signaling system

Figure 4.1 shows an example of a three-aspect signaling system with ATP speed codes. Later on, we will discuss the constraints caused by fixed block signaling system using this simple three-aspect signaling system. However, the methodology proposed in this chapter can be extended to other types of FBS systems.

In the FBS system given in Figure 4.1, the speed code data consists of two parts [115], the authorized-speed code for this block and the target-speed code for the next block as illustrated in Figure 4.1. When a train in Block 4 approaching to Signal 3 will receive a $v_{\text{max}}/v_{\text{yellow}}$ code, to indicate a permitted speed of v_{max} in this block and a target speed of v_{yellow} for the next. When the train enters Block 2, the code changes to $v_{\text{yellow}}/v_{\text{min}}$ because the next block (Block 1) is occupied by train 1, so the speed must be v_{min} (usually taking

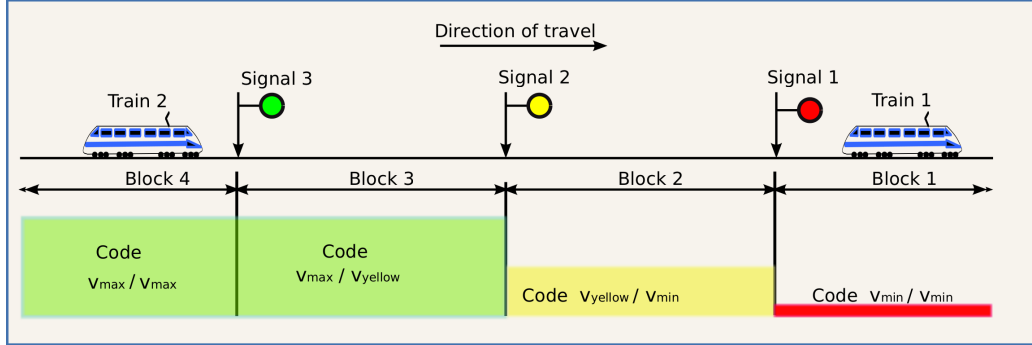


Figure 4.1: Three-aspect fixed block signaling system with ATP speed codes

the value 0 m/s) by the time train reaches the end of Block 2. If the train attempts to pass the indication point for service braking distance before signal 1, the onboard equipment will cause an emergency brake application.

Minimum headway for an FBS system

In order to ensure that a train's operation is not impeded by the signaling system, i.e., a train's operation is not then affected by the train in front, the minimum headway is introduced. The minimum headway is the minimum time separation between successive trains at stations. For undisturbed running in FBS system, the minimum headway can be defined as [60]

$$H_{\min, \text{FBS}} = \frac{L_a}{v_{\max}^F} \left[2 + \text{INT} \left\{ \frac{L_r^F + (v_{\max}^F)^2 / (2a_b^F)}{L_a} \right\} \right] + \frac{v_{\max}^F}{2a_b^F} + t_d^L + \sqrt{\frac{2(L_t^L + L_s)}{a_{\text{acc}}^L}}, \quad (4.3)$$

where $\text{INT}\{\cdot\}$ is a ceiling function that maps a real number to the least integer greater or equal to the argument, L_a is the block length, L_r^F is the distance that the following train will travel during the reaction time t_r^F of the driver and/or train control equipment of the following train, v_{\max}^F is the maximum speed of the following train, a_b^F is the maximum service braking rate, t_d^L is the station dwell time of the leading train, L_t^L is the length of the leading train, L_s is the length of the secure section (a special section to protect the leading train), and a_{acc}^L is the acceleration of the leading train.

The constraints caused by an FBS system

We assume that the total number of fixed block sections between two consecutive stations, i.e., in the interval $[s_{\text{start}}, s_{\text{end}}]$ is M . The index of block sections is denoted as m and the boundaries of the block sections are denoted as $s_{\text{FB}, m}$ with $m \in \{1, 2, \dots, M\}$. We assume that there exists an index $l_m \in \{1, 2, \dots, N+1\}$ such that

$$s_{\text{FB}, m} = s_{l_m}, \quad (4.4)$$

and we define a piecewise constant function such that

$$\ell(k) = m, \quad \text{for } l_m < k \leq l_{m+1}, \quad \text{for } m \in \{1, 2, \dots, M\} \quad (4.5)$$

Then s_k is in the fixed block $(s_{l(k)}, s_{l(k)+1}]$. The constraints caused by the leading train in a three-aspect fixed block signaling system shown in Figure 4.1 can be formulated as the follows:

- If the following train and the leading train are in the same block section, i.e., $t^F(s) \in (t^L(s_{FB,m}), t^L(s_{FB,m+1})]$ with $s \in [s_{FB,m}, s_{FB,m+1}]$, which is in fact not allowed by signaling system, then the speed of the following train must equal to the minimum speed, i.e.

$$v^F(s) = v_{\min}. \quad (4.6)$$

- If the leading train is just one block section before the following train, i.e., $t^F(s) \in (t^L(s_{FB,m+1}), t^L(s_{FB,m+2})]$ with $s \in [s_{FB,m}, s_{FB,m+1}]$, then the speed of the following train at positions $s_{FB,m}$ and $s_{FB,m+1}$ should be less than or equal to v_{Yellow} and equal to v_{\min} , respectively, i.e.

$$\begin{aligned} v^F(s_{FB,m}) &\leq v_{\text{Yellow}}, \\ v^F(s_{FB,m+1}) &= v_{\min}. \end{aligned} \quad (4.7)$$

The deceleration is assumed as a constant for the entire interval $[s_{FB,m}, s_{FB,m+1}]$. Based on the relationship among position, speed, and acceleration, we have

$$2a_{FB,m}(s_{FB,m+1} - s_{FB,m}) = v_{\min}^2 - v_{\text{Yellow}}^2, \quad (4.8)$$

$$2a_{FB,m}(s - s_{FB,m}) = \bar{v}_{\text{Yellow}}^2(s) - v_{\text{Yellow}}^2, \quad (4.9)$$

where $a_{FB,m}$ is the deceleration and $\bar{v}_{\text{Yellow}}(s)$ is the maximum speed for trains at position s for $s \in [s_{FB,m}, s_{FB,m+1}]$. By eliminating $a_{FB,m}$ in (4.8) and (4.9), we obtain

$$\bar{v}_{\text{Yellow}}(s) = \sqrt{v_{\text{Yellow}}^2 + (v_{\min}^2 - v_{\text{Yellow}}^2) \frac{s - s_{FB,m}}{s_{FB,m+1} - s_{FB,m}}}, \quad (4.10)$$

where $\bar{v}_{\text{Yellow}}(\cdot)$ is a function only depending on s . Therefore, in this case we have the constraint

$$v^F(s) \leq \bar{v}_{\text{Yellow}}(s). \quad (4.11)$$

- If the leading train is two blocks before the following train, i.e. $t^F(s) \in (t^L(s_{FB,m+2}), t^L(s_{FB,m+3})]$ with $s \in [s_{FB,m}, s_{FB,m+1}]$, then the speed of the following train at positions $s_{FB,m}$ and $s_{FB,m+1}$ should be less than or equal to v_{\max} and v_{Yellow} , respectively, i.e.

$$\begin{aligned} v^F(s_{FB,m}) &\leq v_{\max}, \\ v^F(s_{FB,m+1}) &\leq v_{\text{Yellow}}. \end{aligned} \quad (4.12)$$

Similarly as $\bar{v}_{\text{Yellow}}(\cdot)$, we can obtain

$$\bar{v}_{\max}(s) = \sqrt{v_{\max}^2 + (v_{\text{Yellow}}^2 - v_{\max}^2) \frac{s - s_{FB,m}}{s_{FB,m+1} - s_{FB,m}}}, \quad (4.13)$$

where $\bar{v}_{\max}(s)$ is the maximum speed for trains at position s . Note that $\bar{v}_{\max}(s)$ only depends on s . Therefore, we in this case have the constraint

$$v^F(s) \leq \bar{v}_{\max}(s). \quad (4.14)$$

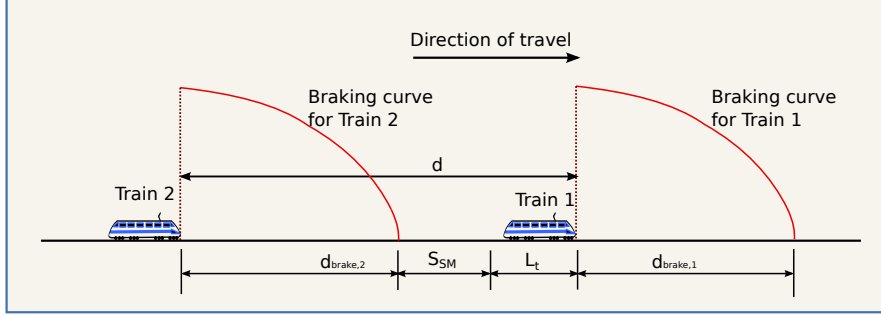


Figure 4.2: An moving block signaling system

4.2.3 Operation of trains in a moving block signaling system

In a pure MBS system¹, the distance between two consecutive trains should be larger than the minimum distance at any time as shown in Figure 4.2. The minimum distance between two successive trains is basically the instantaneous braking distance required by the following train plus a safety margin. As mentioned in Section 2.1.2, in practice the minimum distance in the MBS system is larger than that defined in theory because the driver or the automatic train control system need time to react to situations. The distance between the leading train and the following train in an MBS system should satisfy [115]

$$s^L(t) - s^F(t) \geq L_t^F + (v^F(t))^2 / (2a_b^F) + S_{SM} + L_t^L, \quad (4.15)$$

where $s^L(t)$ and $s^F(t)$ are the positions of the front of the leading train and the following train at time t , $v^F(t)$ is the speed of the following train, a_b^F is the maximal deceleration, S_{SM} is the safety margin distance, L_t^F is the distance that the following train will travel during the reaction time t_r^F of the driver and/or train equipment of the following train, and L_t^L is the length of the leading train. The value of the reaction time could be obtained from historical data.

Minimum headway for a MBS system

The minimum distance between two successive trains (4.15) can be recast as the minimum time difference of two successive trains

$$t^F(s) - t^L(s) \geq t_r^F + t_b^F(s) + t_{safe}^F(s), \quad (4.16)$$

where $t^L(s)$ and $t^F(s)$ are the time instants at which the front of the leading train and the following train pass position s , respectively. The braking time of the following train $t_b^F(s)$ and the time margin $t_{safe}^F(s)$ caused by the safe margin distance and the train length can be computed as

$$t_b^F(s) = v^F(s) / a_b^F, \quad (4.17)$$

¹As stated in Chapter 2, there exist four MBS schemes: moving space blocking signaling, moving time block signaling, pure MBS, and relative MBS. In this thesis, we only consider the pure MBS system, so the MBS system later on refers to the pure MBS system.

$$t_{\text{safe}}^F(s) = (S_{\text{SM}} + L_t^L)/v^F(s), \quad (4.18)$$

where $v^F(s)$ is the speed of the following train at position s .

In order to ensure that near stations a train's operation is not impeded by the signaling system, i.e. a train's operation is not then affected by the (in principle slowly moving or stopped) train in front, the minimum headway is introduced. The minimum headway is the minimum time separation between successive trains at train stations, and it is defined as [115]

$$H_{\text{min,MBS}} = t_d^L + t_{\text{in-out}} = t_d^L + t_r^F + t_{b,\text{max}}^F + t_{\text{safe}}^L, \quad (4.19)$$

with the run-in/run-out time $t_{\text{in-out}} = t_r^F + t_{b,\text{max}}^F + t_{\text{safe}}^L$, where $t_{b,\text{max}}^F$ is the time it takes the following train to come to a full stop when it is running at its maximum speed, i.e. $t_{b,\text{max}}^F = v_{\text{max}}^F/a_b^F$, and the run-out time t_{safe}^L is the time that the leading train needs to completely clear the secure section (i.e. a special section to protect the leading train), if present, and including a safety margin, i.e., $t_{\text{safe}}^L = \sqrt{2(S_{\text{SM}} + L_t^L + L_s)/a_{\text{acc}}^L}$. The acceleration of the leading train a_{acc}^L is usually considered as a constant value for the minimum headway calculation [60].

The constraints caused by an MBS system

The constraints caused by the leading train in MBS system is different in open track and in the station area. In open track area, the minimum time difference between the leading train and the following train should satisfy (4.16). In the station area, the minimum time distance should be larger than the minimum headway defined in (4.19), i.e.,

$$t^F(s) - t^L(s) \geq H_{\text{min,MBS}}. \quad (4.20)$$

4.3 Mathematical formulation of the multiple trains trajectory planning problem

For simplicity, we consider the optimal trajectory planning problem for two trains. However, the solution approaches can be extended to multiple trains. The trajectory planning problem for two trains (i.e., the leading train and the following train) can be formulated as:

$$J = \int_{s_{\text{start}}}^{s_{\text{end}}} \max(0, u^L(s)) ds + \int_{s_{\text{start}}}^{s_{\text{end}}} \max(0, u^F(s)) ds \quad (4.21)$$

subject to

$$\begin{aligned} u_{\text{min}} &\leq u^L(s) \leq u_{\text{max}}, \quad u_{\text{min}} \leq u^F(s) \leq u_{\text{max}}, \quad \forall s \in [s_{\text{start}}, s_{\text{end}}] \\ 0 &< \tilde{E}^L(s) \leq \tilde{E}_{\text{max}}^L(s), \quad 0 < \tilde{E}^F(s) \leq \tilde{E}_{\text{max}}^F(s), \quad \forall s \in [s_{\text{start}}, s_{\text{end}}] \\ \tilde{E}^L(s_{\text{start}}) &= \tilde{E}_{\text{start}}^L, \quad \tilde{E}^F(s_{\text{start}}) = \tilde{E}_{\text{start}}^F, \\ \tilde{E}^L(s_{\text{end}}) &= \tilde{E}_{\text{end}}^L, \quad \tilde{E}^F(s_{\text{end}}) = \tilde{E}_{\text{end}}^F, \\ t^L(s_{\text{start}}) &= T_{\text{start}}^L, \quad t^F(s_{\text{start}}) = T_{\text{start}}^F, \\ t^L(s_{\text{end}}) &= T_{\text{end}}^L, \quad t^F(s_{\text{end}}) = T_{\text{end}}^F, \end{aligned} \quad (4.22)$$

the train model constraints (4.1) and (4.2) for the leading train and the following train, and the constraints caused by the signaling systems in Section 4.2.2 and Section 4.2.3 (i.e.,

(4.6), (4.7), (4.11), (4.12), (4.14), (4.16), and (4.20)), where the objective function J is the energy consumption for these two trains without regenerative braking; $\tilde{E}_{\max}^L(\cdot)$ is equal to $0.5 \cdot (V_{\max}^L(\cdot))^2$ with $V_{\max}^L(\cdot)$ the maximum allowable velocity, which depends on the train characteristics and line conditions, and as such it is usually a piecewise constant function of the coordinate s [72, 87]; s_{start} , $\tilde{E}^L(s_{\text{start}})$, and $t(s_{\text{start}})$ are the position, the kinetic energy per mass, and the departure time for the leading train at the beginning of the route; s_{end} , $\tilde{E}^L(s_{\text{end}})$, and $t^L(s_{\text{end}})$ are the position, the kinetic energy per mass, and the arrival time for the leading train at the end of the route. The notation definitions for the following train are similar. It is assumed that the unit kinetic energy $\tilde{E}^L(s) > 0$ and $\tilde{E}^F(s) > 0$ for the leading train and the following train, which means the train's speed is always strictly larger than zero, i.e. the train travels nonstop [72]. This assumption is nonrestrictive in practice because the initial start and terminal stop can be modeled by small nonzero velocities. Furthermore, in principle the traffic management system does not plan stops intentionally at an intermediate point of the trip.

In addition, the constraints caused by the FBS system or the MBS system proposed in Section 4.2.2 and Section 4.2.3 should also be included in the trajectory planning problem.

Remark 4.1 The ideas on solving the train scheduling problem in a rolling horizon way in Section 7.5 could be adopted to the train trajectory planning for multiple trains in this chapter. \square

4.4 Solution approaches

Now two solution approaches, i.e., the greedy approach and the simultaneous approach, are proposed for solving the optimal control problem for multiple trains under an FBS and or an MBS system.

4.4.1 Greedy approach

In the greedy approach, the leading train's trajectory is first determined and then the trajectory of the following train is optimized based on the results of the leading train. For the trajectory planning problem of the leading train, the objective is only part of (4.21), i.e.

$$J^L = \int_{s_{\text{start}}}^{s_{\text{end}}} \max(0, u^L(s)) ds, \quad (4.23)$$

and the constraints are those related to the leading train. The constraints pertain to both the leading train and the following train are discarded for the trajectory planning for the leading train. The optimization problem of the leading train is the same as the problem solved in Chapter 3, and it can be solved using the pseudospectral method or the MILP approach.

The optimal trajectory planning problem for the following train is similar with that of the leading train. When optimizing the trajectory of the following train, the trajectory of the leader train is already fixed and the value of J^L is also fixed. The objective for the planning problem of the following train is

$$J^F = \int_{s_{\text{start}}}^{s_{\text{end}}} \max(0, u^F(s)) ds. \quad (4.24)$$

The constraints do not only include those in (4.22) related to the following train, but also the constraints caused by the FBS or the MBS system as presented in Sections 4.2.2 or 4.2.3.

In the pseudospectral method, the constraints caused by the signaling system can be easily formulated as the path constraints (see Section 3.4.1 for more information). However, for the MILP approach, we need to approximate the nonlinear constraints into linear constraints by using the transformation properties in Section 3.4.2. The details for the transformation process of the FBS constraints and MBS constraints will be given in Section 4.5.

4.4.2 Simultaneous approach

The simultaneous approach optimizes the trajectories of the leading train and the following train simultaneously. When optimizing the trajectories of multiple trains at the same time, the model for each train is a model of the form (3.1). The trajectories of the leading train and the following train are obtained at the same time and the value of the objective function (4.21) is minimized by solving an optimization problem involving multiple trains. This problem is similar with the trajectory planning problem for a single train and therefore it can also be solved by the pseudospectral method or the MILP approach. The constraints caused by the FBS system and MBS system can be handled similarly as in the greedy approach.

However, compared to the case of a single train, the number of the state variables and constraints of the problem for multiple trains increases linearly with the number of trains. Therefore, the size of the optimal trajectory planning problem for multiple trains is much bigger than the problem for a single train and the computation time of the bigger problem will be much longer. However, since we are now optimizing the train trajectories of two trains at the same time instead of optimizing them one by one, the control performance will in general be better than that of the greedy approach.

Remark 4.2 For the trajectory planning for large number of trains, the simultaneous approach will become slow. So distributed optimization approaches [18] can be applied, where the trajectory of each train could be calculated separately with consideration of constraints caused by other trains, then these trains negotiate with each other and finally converge to a global equilibrium. \square

4.5 Mixed logical dynamic formulation for signaling system constraints

As shown in previous chapter, the optimal control problem can be recast as an MILP problem of the following form:

$$\min_{\tilde{V}} C_J^T \tilde{V}, \quad (4.25)$$

subject to

$$\begin{aligned} F_1 \tilde{V} &\leq F_2 x(1) + f_3, \\ F_4 \tilde{V} &= F_5 x(1) + f_6. \end{aligned} \quad (4.26)$$

In order to include the constraints caused by the FBS system and MBS system (shown in Section 4.2.2 and Section 4.2.3) in the MILP problem, these constraints should be transformed into linear constraints by using the transformation properties in Section 3.4.2. The transformation of these constraints is given as below next.

4.5.1 Multiple trains under fixed block signaling system

The constraints caused by the leading train in the FBS system are first discretized at each grid point s_k for $k = 1, 2, \dots, N + 1$. These logical constraints are then transformed into linear constraints, which can be easily included in the MILP approach.

Discretizing the FBS system constraints

As described in Section 4.2.2, the fixed block sections are indexed by m with $m \in \{1, 2, \dots, M\}$ and the boundaries for the block sections are denoted $s_{\text{FB},m}$ and grid point s_k is in the fixed block $(s_{l_{\ell(k)}}, s_{l_{\ell(k)+1}}]$. The constraints caused by the leading train in a three-aspect fixed block signaling system shown in Figure 4.1 can be transformed at each discrete point s_k as follows:

- If the following train and the leading train are in the same block section, i.e. $t^F(k) \in (t^L(l_{\ell(k)}), t^L(l_{\ell(k)+1})]$, then the speed of the following train must equal to the minimum speed, i.e.

$$v^F(k) = v_{\min}. \quad (4.27)$$

- If the leading train is one block section before the following train, i.e. $t^F(k) \in (t^L(l_{\ell(k)+1}), t^L(l_{\ell(k)+2})]$, then the speed of the following train at positions $s_{l_{\ell(k)}}$ and $s_{l_{\ell(k)+1}}$ should be less than or equal to V_{Yellow} and equal to V_{\min} , respectively, i.e.

$$\begin{aligned} v_{l_{\ell(k)}}^F &\leq v_{\text{Yellow}}, \\ v_{l_{\ell(k)+1}}^F &= v_{\min}. \end{aligned} \quad (4.28)$$

For the grid points $s_k \in (s_{l_{\ell(k)}}, s_{l_{\ell(k)+1}})$, we have

$$v^F(k) \leq \bar{v}_{\text{Yellow},k}, \quad (4.29)$$

where

$$\bar{v}_{\text{Yellow},k} = \sqrt{v_{\text{Yellow}}^2 + (v_{\min}^2 - v_{\text{Yellow}}^2) \frac{s_k - s_{l_{\ell(k)}}}{s_{l_{\ell(k)+1}} - s_{l_{\ell(k)}}}}. \quad (4.30)$$

- If the leading train is two blocks before the following train, i.e. $t^F(k) \in (t^L(l_{\ell(k)+2}), t^L(l_{\ell(k)+3})]$, then the speed of the following train at positions $s_{l_{\ell(k)}}$ and $s_{l_{\ell(k)+1}}$ should be less than or equal to V_{\max} and V_{Yellow} , respectively, i.e.

$$\begin{aligned} v_{l_{\ell(k)}}^F &\leq v_{\max}, \\ v_{l_{\ell(k)+1}}^F &\leq v_{\text{Yellow}}. \end{aligned} \quad (4.31)$$

For the grid points $s_k \in (s_{l_{\ell(k)}}, s_{l_{\ell(k)+1}})$, we have

$$v^F(k) \leq \bar{v}_{\max,k}, \quad (4.32)$$

where

$$\bar{v}_{\max,k} = \sqrt{v_{\max}^2 + (v_{\text{Yellow}}^2 - v_{\max}^2) \frac{s_k - s_{l_{\ell(k)}}}{s_{l_{\ell(k)+1}} - s_{l_{\ell(k)}}}}. \quad (4.33)$$

Considering the FBS constraints into the MILP approach

In order to transform above logical constraints into linear constraints, the following binary variables are introduced:

$$\begin{aligned} [t^F(k) \leq t^L(l_{\ell(k)+1})] &\Leftrightarrow [\delta_1(k) = 1], \\ [t^F(k) \leq t^L(l_{\ell(k)+2})] &\Leftrightarrow [\delta_2(k) = 1], \\ [t^F(k) \leq t^L(l_{\ell(k)+3})] &\Leftrightarrow [\delta_3(k) = 1]. \end{aligned} \quad (4.34)$$

Note that an extra constraint is needed, i.e.

$$t^F(k) > t^L(k), \quad (4.35)$$

because the passing time of the following train at position s_k should be larger than that of the leading train in order to avoid the collision of trains. In addition, based on the definition of $\delta_1(k)$, $\delta_2(k)$, and $\delta_3(k)$, the following logical conditions are satisfied

$$\begin{aligned} [\delta_1(k) = 1] &\Leftrightarrow [\delta_2(k) = \delta_3(k) = 1], \\ [\delta_2(k) = 1] &\Leftrightarrow [\delta_3(k) = 1]. \end{aligned} \quad (4.36)$$

The constraints caused by the leading train in the three-aspect FBS system can then be reformulated as:

$$\delta_1(k)v^F(k) \leq v_{\min}, \quad (4.37)$$

$$(1 - \delta_1(k))\delta_2(k)v^F(k) \leq \bar{v}_{\text{Yellow},k}, \quad (4.38)$$

$$(1 - \delta_2(k))\delta_3(k)v^F(k) \leq \bar{v}_{\max,k}, \quad (4.39)$$

where ' \leq ' is used in (4.37) instead of '=' in (4.27) because $\delta_1(k)v^F(k)$ is equal to 0 and is not equal to v_{\min} when $\delta_1(k) = 0$.

By defining $\tilde{M}_i = T_{\max}^F - t^L(l_{\ell(k)+i})$ that is larger than $\max(t^F(k) - t^L(l_{\ell(k)+i}))$, $\tilde{m}_i = \min(t^F(k) - t^L(l_{\ell(k)+i}))$ that is larger than $T_{\min}^F - t^L(l_{\ell(k)+i})$, and by applying transformation property (3.35) the logical constraints (4.34) can be shown to be equivalent to the following inequalities:

$$\begin{aligned} t^F(k) - t^L(l_{\ell(k)+i}) &\leq \tilde{M}_i(1 - \delta_i(k)), \\ t^F(k) - t^L(l_{\ell(k)+i}) &\geq \varepsilon + (\tilde{m}_i - \varepsilon)\delta_i(k), \end{aligned} \quad (4.40)$$

for $i = 1, 2, 3$, where T_{\max}^F is the arrival time of the following train at the final destination, T_{\min}^F the departure time of the following train, and ε is the machine precision. In addition, we define binary variables $\delta_4(k) = \delta_1(k)\delta_2(k)$ and $\delta_5(k) = \delta_2(k)\delta_3(k)$ to deal with the nonlinear terms $\delta_1(k)\delta_2(k)$ and $\delta_2(k)\delta_3(k)$ in (4.38)-(4.39) respectively. According to the transformation properties in Section 3.4.2, the definitions of $\delta_4(k)$ and $\delta_5(k)$ are equivalent to linear constraints of the form (3.36). In addition, auxiliary variables $z_i^F(k)$ are introduced to deal with the nonlinear terms $\delta_i(k)v^F(k)$, which is defined as

$$z_i^F(k) = \delta_i(k)v^F(k), \quad \text{for } i = 1, 2, 3, 4, 5. \quad (4.41)$$

This definition is equivalent to linear constraints of the form (3.37). The constraints caused by the leading train in fixed block systems can thus be formulated into the MILP problem setting.

In the greedy approach, first the trajectory of the leading train is determined by solving the MILP problem. Next, the optimal control problem of the following train is solved, which is similar to the one of Chapter 3 but with the extra constraints caused by the leading train in FBS system. The coefficient matrices in the mixed logical dynamic model (4.1)-(4.2) are determined by the following train. Since the trajectory of the leading train is known, $t^L(l_{\ell(k)+i})$ is also known. Therefore, \tilde{M}_i and \tilde{m}_i are constants and (4.40) is a system of linear constraints.

When optimizing the trajectories of multiple trains simultaneously, the models for these two trains are determined by a model of the form (4.1)-(4.2). The optimal control problem of these two successive trains can also be rewritten in the form of the MILP problem (4.25)-(4.26) but including the model and constraints of each train and the constraints caused by the FBS system. However, $t^L(l_{\ell(k)+i})$ is now also a variable in this case since the leading train's trajectory also has to be optimized. The constraints (4.40) can be rewritten as

$$\begin{aligned} (T_{\max}^F - t^L(l_{\ell(k)+i}))\delta_i(k) &\leq -t^F(k) + T_{\max}^F, \\ (T_{\min}^F - t^L(l_{\ell(k)+i}) - \varepsilon)\delta_i(k) &\leq t^F(k) - \varepsilon - t^L(l_{\ell(k)+i}). \end{aligned} \quad (4.42)$$

In order to deal with the nonlinear terms in (4.42), we define

$$z_i^L(k) = t^L(l_{\ell(k)+i})\delta_i(k), \quad \text{for } i = 1, 2, 3. \quad (4.43)$$

Similar to $z_i^F(k)$, (4.43) is equivalent to linear constraints according to the transformation properties in Section 3.4.2.

4.5.2 Multiple trains under moving block signaling system

Just as the constraints caused by the FBS system, the constraints caused by the MBS system are first discretized at the grid points s_k , $k = 1, \dots, N+1$. Next, these constraints are approximated by linear constraints, which can easily be included in the MILP formulation (4.25)-(4.26).

Discretizing the MBS constraints

We discretize the constraint (4.16) caused by the MBS system at the grid points s_k as

$$t^F(k) \geq t^L(k) + t_r^F + t_b^F(k) + t_{\text{safe}}^F(k), \quad \text{for } k = 1, 2, \dots, N, \quad (4.44)$$

$$t^F(k) \geq t^L(k) + t_r^F + t_{b,\max}^F + t_d^L + t_{\text{safe}}^L, \quad \text{for } k = N+1. \quad (4.45)$$

In addition, some intermediate constraints are introduced to ensure that the points between the grid points also satisfy the constraints caused by the MBS system. According to (4.16), we obtain the following constraint for each $s \in [s_k, s_{k+1}]$ as:

$$t^F(s) - t_r^F - t_b^F(s) - t_{\text{safe}}^F(s) \geq t^L(s). \quad (4.46)$$

If we assume the left-hand side of (4.46) to be an affine function in the interval $[s_k, s_{k+1}]$, then we can add the following constraints:

$$\begin{aligned} (1 - \alpha)(t^F(k) - t_r^F - t_b^F(k) - t_{\text{safe}}^F(k)) + \alpha(t^F(k+1) - t_r^F - t_b^F(k+1) - t_{\text{safe}}^F(k+1)) \\ \geq t^L(s + \alpha\Delta s_k), \end{aligned} \quad (4.47)$$

for some values α in a finite subset $S_\alpha \in [0, 1]$, e.g. $S_\alpha = \{0.1, 0.2, \dots, 0.9\}$, where $t^L(s + \alpha\Delta s_k)$ is known if the optimal trajectory of the leading train is fixed. Note that for $\alpha = 0$ and $\alpha = 1$, the constraint (4.44) is retrieved (except if $k = N - 1$). However, when we solve the trajectory planning problem for multiple trains using the simultaneous approach, the leading train's trajectory is not known beforehand. So the term $t^L(s + \alpha\Delta s_k)$ is unknown. If we assume the right-hand side of (4.46) is also an affine function, i.e. $t^L(s + \alpha\Delta s_k) = (1 - \alpha)t^L(k) + \alpha t^L(k + 1)$, then it is sufficient to check (4.46) at the points k and $k + 1$ (i.e., for $\alpha = 0$ and $\alpha = 1$), since due to linearity (4.46) will then also automatically be satisfied for all intermediary points.

Considering the MBS constraints into the MILP approach

Note that the constraints (4.44), (4.45), and (4.47) are linear in $t^L(k)$, $t^L(k + 1)$, and $t^F(k)$. However, they are nonlinear in $v^F(k)$ and $v^F(k + 1)$ since the time safety margin (4.18) is a nonlinear function of the following train's velocity $v^F(k)$. Furthermore, the kinetic energy per mass $E^F(k)$ is one of the states instead of $v^F(k)$ with $E^F(k) = 0.5(v^F(k))^2$ (cf. Section 4.2.1). Therefore, both the braking time $t_b^F(k)$ and the safe time margin $t_{\text{safe}}^F(k)$ are nonlinear functions of $E^F(k)$, where

$$t_b^F(k) = \frac{1}{a_b^F} \sqrt{2E^F(k)} \quad (4.48)$$

and

$$t_{\text{safe}}^F(k) = (S_{\text{SM}} + L_t^L) \frac{1}{\sqrt{2E^F(k)}}. \quad (4.49)$$

The nonlinear functions $f_1(\cdot) : E^F \rightarrow \sqrt{2E^F}$ and $f_2(\cdot) : E^F \rightarrow \frac{1}{\sqrt{2E^F}}$ could be approximated by PWA functions as follows (see Chapter 3 for more details about PWA approximation):

$$f_{1,\text{PWA}}(E^F(k)) = \begin{cases} \alpha_1 E^F(k) + \beta_1 & \text{for } E_{\min} \leq E^F(k) < E_1, \\ \alpha_2 E^F(k) + \beta_2 & \text{for } E_1 \leq E^F(k) \leq E_{\max}, \end{cases} \quad (4.50)$$

$$f_{2,\text{PWA}}(E^F(k)) = \begin{cases} \lambda_1 E^F(k) + \mu_1 & \text{for } E_{\min} \leq E^F(k) < E_1, \\ \lambda_2 E^F(k) + \mu_2 & \text{for } E_1 \leq E^F(k) \leq E_{\max}, \end{cases} \quad (4.51)$$

with optimized parameters α_1 , α_2 , β_1 , β_2 , λ_1 , λ_2 , μ_1 , μ_2 , and E_1 . For more details of this transformation into PWA functions, see [127]. Now the constraint (4.44) can be approximated as the following linear constraint:

$$t^F(k) \geq t^L(k) + t_r^F + \frac{1}{a_b^F}(\alpha_1 E^F(k) + \beta_1) + (S_{\text{SM}} + L_t^L)(\lambda_1 E^F(k) + \mu_1), \text{ if } E_{\min} \leq E^F(k) < E_1 \quad (4.52)$$

$$t^F(k) \geq t^L(k) + t_r^F + \frac{1}{a_b^F}(\alpha_2 E^F(k) + \beta_2) + (S_{\text{SM}} + L_t^L)(\lambda_2 E^F(k) + \mu_2), \text{ if } E_1 \leq E^F(k) \leq E_{\max} \quad (4.53)$$

Similarly, the constraints (4.45) and (4.46) can also be written as linear constraints. These approximated linear constraints caused by the MBS system can be easily included in the MILP approach and we still get an MILP problem.

The greedy approach and the simultaneous approach presented in Section 4.4 can be applied for trajectory planning of multiple trains in MBS systems. In the greedy approach

$t^L(k)$ and $t^L(s + \alpha \Delta s_k)$ are known for the trajectory planning problem for the following train since the trajectory of the leading train is known by the zone controller or the following train. The trajectory planning problem for the following train is similar to the one of Chapter 3 but with the MBS constraints of Section 4.5.2. When optimizing the trajectories of multiple trains at the same time, the model for each of these trains is determined by a model of the form (4.1)-(4.2). The optimal control problem of these successive trains can also be rewritten in the form of the MILP problem (4.25)-(4.26).

4.5.3 Extension: mode vector constraints

The computation time needed for solving an MILP problem for a single train is usually small if we take a small value for N . However, the computation time increases quickly with the value of N and the number of trains considered in the MILP problem for multiple trains. In the worst case, the computation time grows exponentially with the number of integer variables. In order to solve the MILP problem for multiple trains in a reasonable time, we introduce the so-called mode vector constraints, which have already been applied to sewer networks [68].

A mode of the MLD model for a single train refers to a specific value of the binary vector $\delta^i = [\delta_1^i \ \delta_2^i \ \dots \ \delta_{M_i}^i]^T$, where i represents the i th train and M_i denotes the dimension of the binary vector of the i th train (see the MILP formulation (4.25)-(4.26)). Furthermore, a mode vector is defined as a tuple of binary vectors for each train considered in the MILP problem, i.e. $\Delta = (\delta^{1T}, \delta^{2T}, \dots, \delta^{IT})^T$, where I is the number of trains considered in the problem. Let $\bar{\Delta} = (\bar{\delta}^{1T}, \bar{\delta}^{2T}, \dots, \bar{\delta}^{IT})^T$ be a reference mode vector of the MILP problem for multiple trains. Note however that this comes at a cost of reduced optimality; the optimality can next be improved again by solving the full I -trains MILP problem with the mode constraints. The mode vector constraints can be defined as

$$\sum_{m=1}^{M_i} |\bar{\delta}_m^i - \delta_m^i| \leq D_i \quad \text{for } i = 1, 2, \dots, I, \quad (4.54)$$

or as

$$\sum_{i=1}^I \sum_{m=1}^{M_i} |\bar{\delta}_m^i - \delta_m^i| \leq D, \quad (4.55)$$

where D_i and D are preselected bounds (a nonnegative integer value) on the number of 0-1switches (or vice versa) in the entire of the updated mode vector with respect to the reference mode vector. Note that the mode vector constraints (4.54) and (4.55) can be recast as linear constraints by introducing some auxiliary variables to deal with the absolute values $|\bar{\delta}_m^i - \delta_m^i|$ for $i = 1, 2, \dots, I$ and $m = 1, 2, \dots, M_i$. Furthermore, the mode vector constraints can be seen as the Hamming distance between Δ and $\bar{\Delta}$ if we would expand Δ and $\bar{\Delta}$ into binary strings [68].

An important practical problem is to find a reference mode vector $\bar{\Delta}$. It is stated in [95] that physics or heuristic knowledge of the system can often be used to find $\bar{\Delta}$ that fulfills the physical constraints of the system. A good candidate for the reference mode vector can be obtained by solving the optimal control problem for multiple trains sequentially since the computation time of the solving I single-train MILP problems will be much less than solving the full I -trains MILP problem all at once.

Table 4.1: Parameters of train and line path

Property	Symbol	Value
Train mass [kg]	m	$2.78 \cdot 10^5$
Basic resistance [N/kg]	R_b	$0.0142 + 1.0393 \cdot 10^{-4} v^2$
Mass factor [-]	ρ	1.06
Maximum velocity [m/s]	V_{\max}	22.2
Line length [m]	s_{end}	1332
Minimum kinetic energy [J]	E_{\min}	0.1
Maximum traction force [N]	u_{\max}	$2.224 \cdot 10^5$
Maximum braking force (regular) [N]	u_{\min}	$-2.224 \cdot 10^5$

Table 4.2: Parameters for the calculation of the minimum headway

Property	Symbol	Value
Train length [m]	L_t	90
Safety margin [m]	S_{SM}	30
Length of the secure section [m]	L_s	60
Initial acceleration [m/s^2]	a_{acc}^L	1
Braking deceleration [m/s^2]	a_b^F	0.9
Braking reaction time [s]	t_r	1
Station dwell time [s]	t_d	25

By using the MILP approach with mode vector constraints, the computation time can be reduced significantly. A case study illustrate this is presented in [131], see there for more information.

4.6 Case study

In order to illustrate the performance of the proposed greedy and simultaneous approaches for the optimal trajectory planning for multiple trains under an FBS system and an MBS system, a part of the Beijing Yizhuang subway line is used as a test case study.

4.6.1 Set-up

The performance of the MILP approach is compared with the widely used pseudospectral method and for both approaches we consider both the greedy and the simultaneous variant. For the sake of simplicity, we only consider two² stations in the Yizhuang subway line: Songjiazhuang station and Xiaocun station. The track length between these two stations is 1332 m and the speed limits and grade profile are shown in Figure 4.3. The parameters of the train and the line path are listed in Table 4.1. The rotating mass factor is often chosen as 1.06 in the literature [57] and therefore we also adopt this value. According to the assumptions

²Note however that the MILP approaches and the pseudospectral methods can also be applied if more than 2 stations are considered.

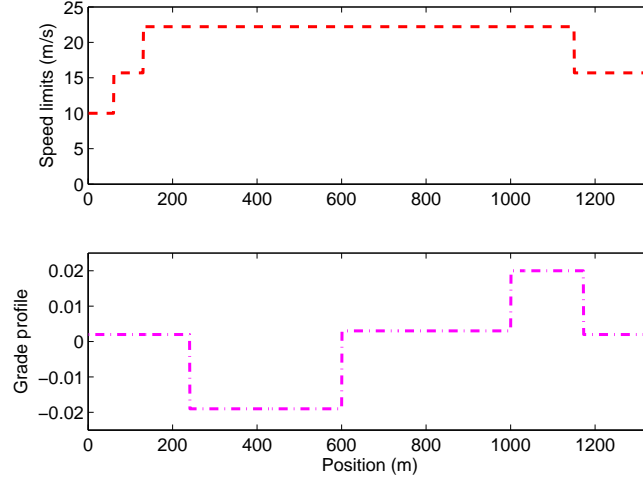


Figure 4.3: The speed limits and the grade profile between Songjiazhuang station and Xiaocun station

made in Section 3.3, the unit kinetic energy should be larger than some positive threshold E_{\min} . In this test case, the minimum kinetic energy is chosen as 0.1 J. The maximum traction force of the train is a nonlinear function of the train's velocity and the maximum value of this function is 315 kN. Moreover, the maximum acceleration and maximum deceleration used for trajectory planning along the line are assumed to be 0.8 m/s^2 and -0.8 m/s^2 respectively in order to make sure that the planned trajectory can be followed by the train controlled by the lower level controller. Since the train mass here is $2.78 \cdot 10^5 \text{ kg}$, the maximum traction force and maximum braking force are 222.4 kN and -222.4 kN, respectively. The objective function of the optimal train control problem considered in this chapter is the energy consumption of the train operation without regenerative braking (cf. (4.21)).

In this case study, two trains are scheduled to run from Songjiazhuang station with a headway of 75 s to Xiaocun station. We consider two cases: the FBS system and the MBS system. Moreover, we assume that the leading train has a malfunction during the whole simulation and as a consequence its maximum speed is reduced to 40 km/h, i.e. 11.1 m/s. In addition, the leading train and the following train will arrive at different platforms in Xiaocun station and the following train will overtake the leading train at Xiaocun station. The parameters for the calculation of the minimum headway are given in Table 4.2. The length of the train is 90 m and the reaction time of the driver is 1 s. For the FBS system, we assume that there exist four fixed block sections between Songjiazhuang station and Xiaocun station and all fixed block sections are of equal length, i.e. 333 m. The minimum headway of the FBS system can be calculated according to (4.3) and it is equal to 98.4 s. Based on the parameters of Table 4.2, the run-in/run-out time $t_{\text{in-out}}$ in (4.19) is equal to 44.6 s and the minimum headway of the MBS system equals 69.6 s. Note that the headway 75 s is smaller than the minimum headway of the FBS system (i.e., 98.4 s) and is larger than the minimum headway of the MBS system (i.e., 69.6 s).

In the MILP approaches, the lengths Δs_k of the intervals $[s_k, s_{k+1}]$ depend on the speed limits, gradient profile, fixed block length, and so on. If the number of the space intervals N is large, then the accuracy will be better but the computation time of the MILP approaches will be larger. For this case study, the number of space intervals N is chosen as 20, 40, and 60, respectively. Moreover, the space intervals are taken to have equal length being 66.6 m, 33.3 m, and 22.2 m respectively for the different three values of N . In addition, the nonlinear terms in the trajectory planning problem, such as the nonlinear terms in the differential equations of the train model, are approximated by PWA functions (see Chapter 3 for more details). If we take PWA approximations with more subfunctions, the approximation accuracy will be better. Here, PWA functions with 2 subfunctions and 3 subfunctions are compared. We use the CPLEX solver via the Tomlab³ interface to Matlab for solving the MILP problems.

The pseudospectral method is a state-of-the-art method for solving optimal control problems (cf. Chapter 3, Appendix 3.A, [37], and [51]). The approximation error of the pseudospectral method can be reduced by taking more collocation points. The numbers of LGL points are taken as 20, 40, 80, and 120, respectively. There are several packages that implement the pseudospectral method (see Appendix 3.A for detailed information). One of them is PROPT, which supports the description of the differential algebraic equations and can call many solvers, such as MINOS and SNOPT, to solve the resulting nonlinear programming problem. We in our case study use PROPT solver through the Tomlab interface to Matlab and SNOPT is used to solve the resulting nonlinear programming problem. Note that both SNOPT and PROPT are implemented in object code.

4.6.2 Results for the fixed block signaling system

Table 4.3 shows the performance of the MILP approaches and the pseudospectral methods for the trajectory planning of two trains in the FBS system. The performance mentioned here, such as the energy consumption and the end time violation, is calculated by applying the obtained optimal control inputs into the nonlinear train model (3.1). The total energy consumption is the sum of the energy consumption of the leading train and the following train. The end time violation is the sum of the absolute values of the differences between the real running times and the planned running times of the leading train and the following train. The energy consumption for each train is influenced by the sign of the difference between the real running time and the planned running time. If the real running time is larger than the planned running time, e.g. 105 s for the following train, then the energy consumption usually

³Tomlab website: <http://tomopt.com>.

⁴For the greedy and simultaneous MILP approaches, the n in the notation n/m is the number of subfunctions used in the PWA approximations of the nonlinear terms in the differential equations of the nonlinear train model and m is the number of the space intervals. For the greedy and simultaneous pseudospectral methods, the n indicates the number of collocation points used.

⁵For all four approaches, the number n_1 in the notation $n_1/n_2/n_3$ is the number of real-valued variables, n_2 is the number of integer-valued variables, and n_3 is the number of constraints. Note that for the greedy and simultaneous MILP approach, all the constraints are linear. Furthermore, in the greedy MILP and greedy pseudospectral methods, two subproblems are solved. One is for the trajectory planning problem of the leading train, the size of which is shown as L: $n_1/n_2/n_3$. The other is for the trajectory planning problem for the following train considering the constraints caused by the leading train, the size of which is shown as F: $n_1/n_2/n_3$. For the simultaneous MILP and simultaneous pseudospectral approaches, the trajectories of the leading train and the following train are obtained by solving a combined optimization problem.

Table 4.3: Performance comparison of the greedy and simultaneous approach using the MILP and the pseudospectral method for the FBS system

Approach	Method	Variant ⁴	Problem size ⁵	Total energy consumption [MJ]	Total end time violation [s]	Total CPU time [s]
Greedy	MILP	2/20	L: 99/40/462 F: 199/140/1116	114.48	3.17	1.34
		2/40	L: 199/80/922 F: 399/280/2326	113.77	1.95	14.22
		2/60	L: 299/120/1382 F: 599/420/3486	110.60	1.48	24.26
		3/20	L: 179/120/962 F: 279/220/1686	114.79	2.82	6.87
		3/40	L: 359/240/1542 F: 559/440/3366	112.58	2.21	96.56
		3/60	L: 539/360/2702 F: 839/660/5046	110.03	1.41	229.72
	Pseudo-spectral	20	L: 60/0/1107 F: 60/0/1307	112.59	2.52	231.83
		40	L: 120/0/1207 F: 120/0/1607	110.04	1.43	1381.71
		80	L: 240/0/1407 F: 240/0/2207	109.48	0.76	1935.09
		120	L: 360/0/1607 F: 360/0/2807	109.17	0.45	3588.10
Simul-taneous	MILP	2/20	358/180/1837	109.65	4.07	2.76
		2/40	718/360/3697	108.06	2.91	78.40
		2/60	1078/540/5557	106.32	1.65	204.38
		3/20	518/340/2877	108.44	3.19	24.13
		3/40	1038/680/5777	106.58	1.73	184.76
		3/60	1558/1020/8677	106.19	1.08	349.31
	Pseudo-spectral	20	120/0/2414	109.53	3.58	445.52
		40	240/0/2814	106.44	1.68	1521.30
		80	480/0/3614	105.92	0.78	3005.71
		120	720/0/4414	105.68	0.54	4875.23

becomes less since a train can run with a smaller average speed. For the greedy approach, the CPU time is the sum of the time used to solve the optimal control problem for the leading train and the optimal control problem for the following train on a 1.8 GHz Intel Core 2 Duo CPU running a 64-bit Linux operating system. In the simultaneous approach, the total CPU time is equal to the time spent by solving the optimal control problem since the optimal control inputs for the leading train and the following train are obtained simultaneously.

Furthermore, Figures 4.4 and 4.5 show the optimal control inputs and the speed-position trajectories obtained by the MILP approaches and the pseudospectral methods for the FBS system, where the number of space intervals for the MILP approaches and the number of the collocation points for the pseudospectral methods are taken as 40 and the nonlinear terms in the nonlinear train model are approximated using PWA functions with 3 affine subfunctions for the MILP approach. The speed-position trajectories of the leading train and the following train were produced by applying the optimal control inputs obtained by solving the optimal control problems to the nonlinear train model. It is observed from Figures 4.4 and 4.5 that the operation of the following train is affected by the leading train in the FBS system, where the signal at position 666 m shows a yellow aspect to the following train. Because the headway between the leading train and the following train is taken as 75 s, which is less than the minimum headway of the FBS system, i.e. 98.4 s. Thus, the following train must slow down to satisfy the speed limit caused by the yellow signal aspect, which is 40 km/h, i.e. 11.1 m/s.

In the case study for the FBS system, the corresponding energy consumption, the end time violation, and the computation time of the greedy MILP approach are 112.58 MJ, 2.21 s, and 96.56 s, respectively. For the greedy pseudospectral method, the energy consumption, the end time violation, and the calculation time of both trains are 110.04 MJ, 1.43 s, and 1381.71 s, respectively. It can be seen that the energy consumption and the end time violation of the greedy MILP approach are a bit larger than those of the greedy pseudospectral method. However, the computation time of the greedy pseudospectral method is more than one order of magnitude longer than that of the greedy MILP approach.

The energy consumption, end time violation, and calculation time are 106.58 MJ, 1.73 s, and 184.76 s, respectively, using the simultaneous MILP approach. For the simultaneous pseudospectral method, they are 106.44 MJ, 1.68 s, and 1521.30 s, respectively. When compared with the greedy MILP approach, the energy consumption and the end time violation of the simultaneous MILP approach become smaller since the leading train's trajectory can also be optimized with respect to the following train. However, the computation time of the simultaneous MILP approach becomes longer than the greedy MILP approach because in the simultaneous MILP approach, the size of the optimization problem is almost doubled. In addition, the simultaneous pseudospectral method yields better performance, i.e., lower energy consumption and lower end time violation when comparing with the greedy pseudospectral method.

As can be observed in Table 4.3, the energy consumption and the end time violation of the greedy MILP approach are generally larger than that of the greedy pseudospectral method for the same number of discrete intervals and collocation points. However, the computation time of the greedy pseudospectral method is one to two orders of magnitude higher than that of the greedy MILP approach. In addition, the energy consumption and the end time violation become less if we take more space intervals for the greedy MILP approach and more collocation points for the greedy pseudospectral method. The results

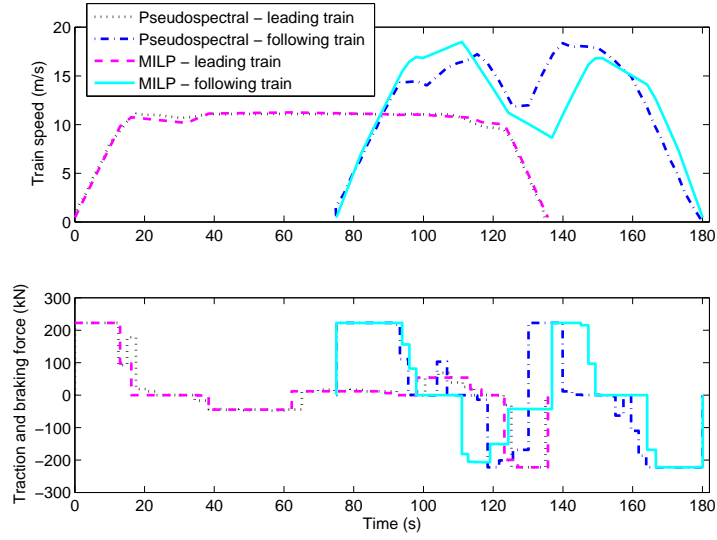


Figure 4.4: Trajectory planning for two trains under the FBS system with headway 75 s using the greedy MILP and the pseudospectral approach

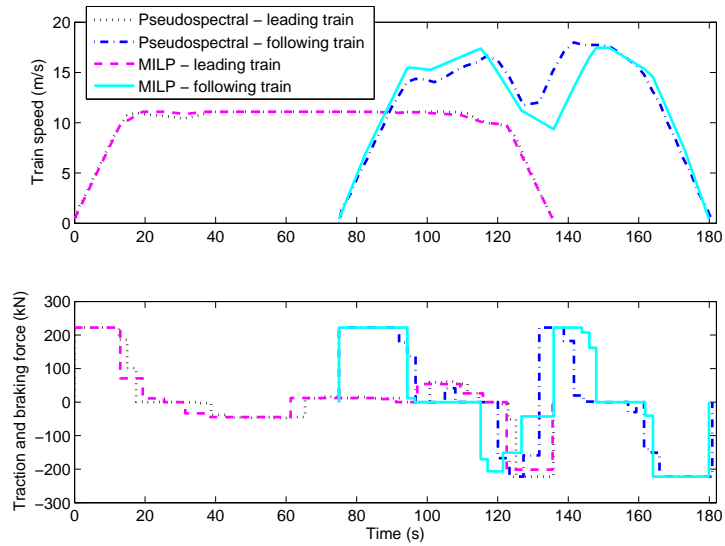


Figure 4.5: Trajectory planning for two trains under the FBS system with headway 75 s using the simultaneous MILP and the pseudospectral approaches

obtained for the greedy MILP approach and the greedy pseudospectral method also hold for the simultaneous MILP approach and the simultaneous pseudospectral method. Moreover, it can be observed that the energy consumption of the simultaneous MILP approach is less than that of the greedy MILP approach, while the computation times of the simultaneous MILP approach are higher than those of the greedy MILP approach. This also holds for the greedy and simultaneous pseudospectral method.

4.6.3 Results for the moving block signaling system

The performance of the greedy and simultaneous approaches is shown in Table 4.4 for the MBS system. In particular, Figures 4.6 and 4.7 show the speed-position trajectories and the optimal control inputs for the MBS system with the same set-up as Section 4.6.2. Similarly, the speed-position trajectories, the end time violation, etc. are obtained by applying the optimal control inputs to the nonlinear train model (3.1). As we can see from Figures 4.6 and 4.7, the operation of the following train is not affected by the leading train since the scheduled headway (75 s) is larger than the minimum headway of the MBS system (69.6 s).

In the case study for the MBS system, the energy consumption, the end time violation, and the computation time for the greedy MILP approach are 67.89 MJ, 1.68 s, and 75.98 s. In addition, the energy consumption, the end time violation, and the computation time for the greedy pseudospectral method are 67.42 MJ, 1.66 s, and 729.24 s. The energy consumption and the end time violation of the greedy MILP approach are slightly larger than those of the greedy pseudospectral method. However, the computation time of the greedy pseudospectral method is almost one order of magnitude larger than that of the greedy MILP approach. This also holds for the results obtained by the simultaneous MILP and pseudospectral approach. In general, the total energy consumption and the total end time violation decrease with the increase of the number of the space intervals in MILP approach and the number of collocation points in pseudospectral method. Nevertheless, the total CPU time increases quickly with respect to the number of the space intervals and collocation points. For the greedy and/or simultaneous MILP approach, if we take PWA approximations with more subfunctions, the end time violation also decreases. Furthermore, when compared with the greedy MILP (or pseudospectral) approach, the simultaneous MILP (or pseudospectral) approach in principle has a better control performance in principle but it is characterized by a much higher computational burden since the size of the optimization problem is almost doubled.

4.6.4 Discussion

The simulation results show that when compared with the greedy pseudospectral method, the energy consumption and the end time violation of the greedy MILP approach are inconsiderably larger, but the computation time is one to two orders of magnitude shorter. Similarly, the energy consumption and the end time violation of the simultaneous MILP approach are lightly larger than those of the simultaneous pseudospectral method. However, the computation time of the simultaneous MILP approach is much smaller than that of the simultaneous pseudospectral method. Moreover, the energy consumption of the greedy MILP approach is larger than that of the simultaneous MILP approach, but the computation time of the simultaneous MILP approach is longer in general. Furthermore, the energy

Table 4.4: Performance comparison of the greedy and simultaneous approach using the MILP and the pseudospectral method for the MBS system

Approach	Method	Variant	Problem size	Total energy consumption [MJ]	Total end time violation [s]	Total CPU time [s]
Greedy	MILP	2/20	L: 99/40/462 F: 99/40/632	69.16	3.45	1.41
		2/40	L: 199/80/922 F: 199/80/1292	68.34	2.47	12.03
		2/60	L: 299/120/1382 F: 299/120/1952	67.56	1.08	24.19
		3/20	L: 179/120/962 F: 179/120/1152	68.76	2.79	9.87
		3/40	L: 359/240/1542 F: 359/240/2332	67.89	1.68	75.98
		3/60	L: 539/360/2702 F: 539/360/3662	67.12	1.34	254.80
	Pseudo-spectral	20	L: 60/0/1107 F: 60/0/1157	68.88	3.94	89.71
		40	L: 120/0/1207 F: 120/0/1257	67.42	1.66	729.24
		80	L: 240/0/1407 F: 240/0/1457	67.03	0.75	1483.42
		120	L: 360/0/1607 F: 360/0/1657	66.85	0.29	4542.20
Simul-taneous	MILP	2/20	198/80/1094	68.14	2.58	2.45
		2/40	398/160/2214	67.52	1.53	56.17
		2/60	598/240/3334	66.62	1.26	150.67
		3/20	358/240/2134	67.73	1.71	13.49
		3/40	718/480/4294	67.03	1.04	121.29
		3/60	1078/720/6454	66.87	0.85	420.37
	Pseudo-spectral	20	120/0/2264	68.62	3.61	161.15
		40	240/0/2464	67.50	1.37	1051.06
		80	480/0/2864	67.07	0.66	2984.23
		120	720/0/3264	66.26	0.31	6954.37

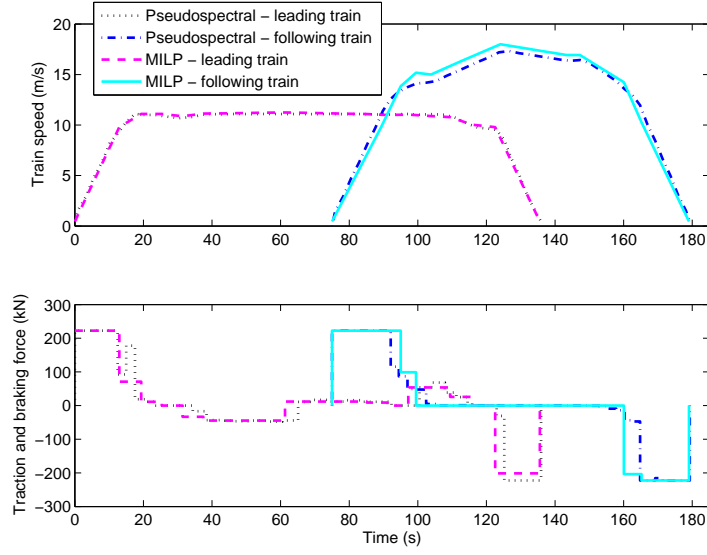


Figure 4.6: Trajectory planning for two trains under the MBS system with headway 75 s using the greedy MILP and the pseudospectral approaches

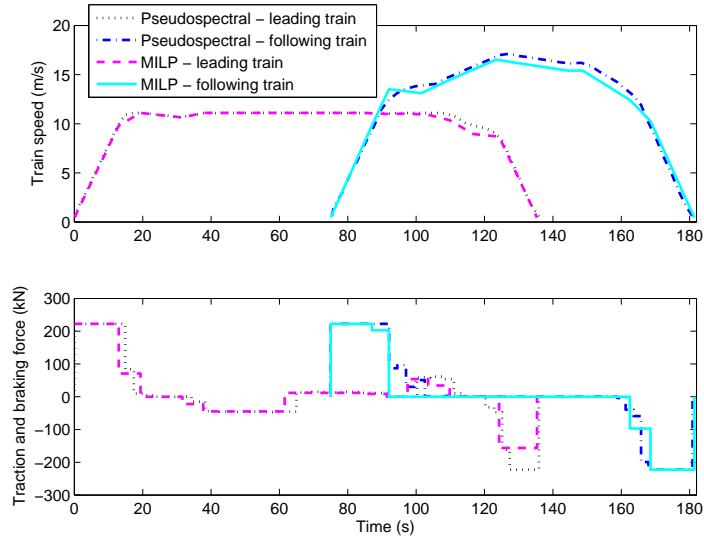


Figure 4.7: Trajectory planning for two trains under MBS system with headway 75 s using the simultaneous MILP and the pseudospectral approaches

consumption of the simultaneous pseudospectral method is lower than that of the greedy pseudospectral method, but it requires a much higher computation time.

4.7 Conclusions

In this chapter, we have proposed two approaches, namely the greedy approach and the simultaneous approach, to solve the optimal trajectory planning problem for multiple trains. In the greedy approach, the optimal trajectory planning problem of the leading train is solved first and then based on the optimal control inputs of the leading train, the trajectory planning problem for the following train is solved. For the simultaneous approach, the trajectories of the leading train and the following train are optimized at the same time. The constraints caused by the leading train in a fixed block signaling system and a moving block signaling system are included in the optimal trajectory planning problem for multiple trains. In the MILP approach, the nonlinear terms in the train model and constraints are approximated by piecewise affine functions. In this way, the optimal trajectory planning problem for multiple trains can then be recast as a mixed integer linear programming (MILP) problem. The performance of the greedy and the simultaneous MILP approach has been compared with the greedy and the simultaneous pseudospectral method in a case study. The simulation results have shown that the MILP approaches yield a similar control performance as the pseudospectral methods but they require a much less computation time.

A topic for future work will be an extensive comparison and assessment between the MILP approach, the pseudospectral method (also using other nonlinear programming sub-solvers, e.g., MINOS and KNITRO), a dynamic programming algorithm [58], and other approaches and frameworks (such as AMPL, APMonitor, and ASCEND) described in the literature for various case studies and a wide range of scenarios.

Chapter 5

OD-Independent Train Scheduling for an Urban Rail Transit Line

In the previous two chapters, we have discussed trajectory planning for trains in a railway network based on given train schedules. In this chapter, the scheduling problem based on origin-destination-independent (OD-independent) passenger demands for an urban rail transit line is considered with the aim of minimizing the total travel time of passengers and the energy consumption of trains. We propose a new iterative convex programming (ICP) approach to solve this train scheduling problem. The performance of the ICP approach is compared via a case study with other alternative approaches, such as nonlinear programming approaches, a mixed integer nonlinear programming (MINLP) approach, and a mixed integer linear programming (MILP) approach.

The research discussed in this chapter has been published in [132, 133, 135].

5.1 Introduction

As has been pointed out in Chapter 2, the train scheduling of urban rail transit systems becomes more and more important for reducing the operation costs of railway operators and for guaranteeing passenger satisfaction. In the urban rail transit systems considered here, the lines are assumed to be separated from each other and each direction of the line has a separate rail track. Therefore, trains do not overtake each other. In addition, for urban rail transit systems with high frequencies, it is not a major issue to the passengers anymore whether or not the train schedule is cyclic since new trains arrive at a station every 2 to 5 minutes. In practice, rail transport operators therefore do not announce the train schedule to passengers but only provide some information, such as that a train will arrive within 2 minutes. Hence, rail transport operators can schedule trains in real time based on the current situation, such as the number of waiting passengers at stations, the passenger arrival rates, and the number and position of running trains.

In this chapter, we propose a real-time scheduling approach to schedule trains in a receding horizon way based on the OD-independent passenger demands for an urban rail transit line, where a predefined timetable or service headway is not needed. The train scheduling problem is essentially a multi-objective optimization problem because it should consider both the benefits of the rail operators and the passengers [44, 59, 78, 124]. This chapter considers multi-objective optimization for the train scheduling problem, where the energy consumption of the trains and the total travel time of passengers are minimized. Since the train scheduling problem here is a non-smooth non-convex problem, we propose a new iterative convex programming (ICP) approach to solve the problem. The performance of the ICP approach is compared with a pattern search method, a mixed integer nonlinear programming (MINLP) approach, a mixed integer linear programming (MILP) approach, and a sequential quadratic programming (SQP) method.

The rest of this chapter is structured as follows. Section 5.2 formulates the evolution equations for the arrivals and departures of trains, the passenger demand characteristics, and the passenger/vehicle interaction. Section 5.3 describes the multi-objective cost function and the constraints of the real-time train scheduling problem. Section 5.4 proposes several solution approaches for the train scheduling problem, in particular, the new iterative convex programming approach. In addition, we show that the problem with stop-skipping can be solved using an MINLP approach or an MILP approach. Section 5.5 further extends the train scheduling model to a model with stop-skipping at small stations. Section 5.6 compares the performance of the proposed solution approaches in Section 5.4 with a case study. Finally, conclusions and recommendations are provided in Section 5.7.

5.2 Model formulation

This chapter considers one direction of an urban transit line consisting of J stations as shown in Figure 5.1. Station 1 is the origin station and station J is the final station of each trip.

We make the following assumptions when formulating the real-time scheduling model:

- A.1 Station j for $j \in \{2, 3, \dots, J-1\}$ can only accommodate one train at a time and no passing can occur at any point in the line.
- A.2 Passengers arrive at a constant rate¹ λ_j at station j .
- A.3 The number of passengers alighting from trains at station j for $j \in \{1, 2, \dots, J\}$ is a fixed proportion ρ_j of the number of passengers on board of trains when they arrive at station j .
- A.4 The number of passengers waiting at a station and the number of passengers on board immediately after a train's departure are approximated by real numbers.
- A.5 The operation of trains only consists of three phases: the acceleration phase, the speed holding phase, and the deceleration phase. Moreover, the acceleration and the deceleration are taken to be fixed constants.

¹Here we use a deterministic model to describe the passenger arrival process.

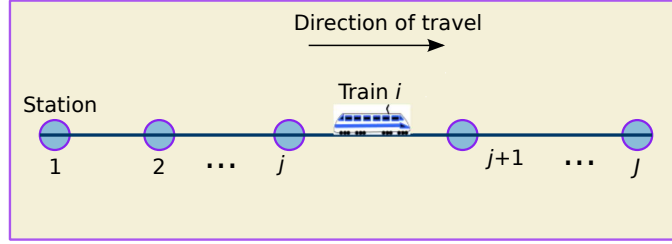


Figure 5.1: Illustration of a subway line

Assumption A.1 generally holds for most urban transit systems, which are usually operated in first-in first-out order from station 1 to J . Assumption A.2 is consistent with observed random passengers arrivals for short headway (less than 10 minutes) services [96]. An estimate of these passenger arrival rate λ_j at stations can be obtained by analyzing historical data of the passenger flow. Assumption A.3 is made according to [110]. Similar as the passenger arrival rate, the passenger alighting proportion ρ_j can be estimated by analyzing historical data. For Assumption A.4, if the number of passengers is high, then the error made by this assumption is small. Furthermore, this assumption simplifies the optimization problem later on. In order to simplify the operation model for the trains, the detailed dynamics are not included in the model formulation, but only the three phases mentioned in Assumption A.5 are considered. However, once the running times between consecutive stations are fixed, a more accurate speed profile for the operation of trains can be calculated as a lower level control problem (see Chapter 3 for more information).

5.2.1 Arrivals and departures

In the literature on train scheduling [28, 36, 57, 59, 81], the operation of trains is usually described by the departure times, arrival times, running times, and dwell times. As has been discussed in Chapter 2, the operation of trains is controlled through a multi-layer control framework. This chapter focus on the train scheduling. In the scheduling layer, we use an online model-based approach; this means the model needs to be simulated repeatedly. In order to obtain a balanced trade-off between the accuracy and the computation speed, a macroscopic model is used. The detailed train dynamics, the position of block signals, the detection of trains, etc. can then be taken into account by the lower level control layer.

The departure time $d_{i,j}$ as shown in Figure 5.2 of train i at station j is

$$d_{i,j} = a_{i,j} + \tau_{i,j}, \quad (5.1)$$

where $a_{i,j}$ and $\tau_{i,j}$ are the arrival time and the dwell time of train i at station j . In the literature, the dwell time is usually considered as a constant. However, in practice, it is influenced by the number of passengers boarding and alighting from a train. Therefore, we consider a variable dwell time, as will be explained in Section 5.2.3.

The track section between station j and station $j+1$ is denoted as segment j as illustrated in Figure 5.2. So the arrival time $a_{i,j+1}$ of train i at station $j+1$ equals the sum of the departure time $d_{i,j}$ at station j and the running time $r_{i,j}$ on segment j for train i :

$$a_{i,j+1} = d_{i,j} + r_{i,j}. \quad (5.2)$$

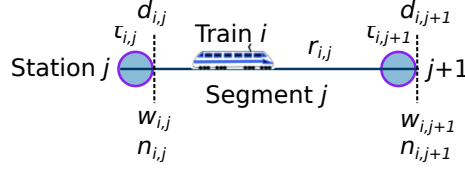


Figure 5.2: The variables for the operation of trains and passenger characteristics

Remark 5.1 Max-plus algebra has been adopted for the train rescheduling and timetable analysis [53, 70, 120]. However, it is not suitable for the problem proposed here because the max operation will not be used since there is no fixed timetable yet. Moreover, it would result in a bilinear varying max-plus model since both running times and dwell times are variables. \square

According to Assumption A.5, we denote the acceleration and the deceleration as $a_{i,j}^{\text{acc}}$ and $a_{i,j}^{\text{dec}}$, respectively. If we define the speed in the speed holding phase as $v_{i,j}$, then the running distance of these three phases can be calculated as

$$s_{i,j}^{\text{acc}} = \frac{v_{i,j}^2}{2a_{i,j}^{\text{acc}}}, \quad s_{i,j}^{\text{dec}} = -\frac{v_{i,j}^2}{2a_{i,j}^{\text{dec}}}, \quad s_{i,j}^{\text{hold}} = s_j - s_{i,j}^{\text{acc}} - s_{i,j}^{\text{dec}},$$

where s_j is the length of segment j . Therefore, the running time of train i for segment j can be written as $r_{i,j} = t_{i,j}^{\text{acc}} + t_{i,j}^{\text{hold}} + t_{i,j}^{\text{dec}}$, which can then be recast as

$$r_{i,j} = \frac{s_j}{v_{i,j}} + \frac{v_{i,j}}{2a_{i,j}^{\text{acc}}} - \frac{v_{i,j}}{2a_{i,j}^{\text{dec}}}. \quad (5.3)$$

Note that the speed $v_{i,j}$ of the holding phase should satisfy

$$v_{i,j} \in [v_{i,j,\min}, v_{i,j,\max}], \quad (5.4)$$

where $v_{i,j,\min}$ and $v_{i,j,\max}$ are the minimal and maximal running speed for the speed holding phase of train i at segment j , respectively. The maximum running speed is limited by the train characteristics and the condition of the line. The minimum running speed is introduced to ensure passenger satisfaction since if trains run too slow, the passengers may complain.

Remark 5.2 The coasting phase of the operation of trains can be included as follows (at the cost of an increased number of variables and an increased computational complexity). In the coasting phase, both the traction force and braking force are equal to zero, where the train speed slows down because of the resistance. With the model given above, the entering speed (i.e., the holding speed) of the coasting phase is $v_{i,j}$. We denote the speed at the end of the coasting phase as $v'_{i,j}$. The resistance varies with the train speed. Here, we approximate the resistance using the mean speed $v_{m,i,j} = (v_{i,j} + v'_{i,j})/2$ of the coasting phase:

$$R_i = m(k_{1i} + k_{2i}v_{m,i,j}) + k_{3i}v_{m,i,j}^2,$$

where m is the mass of the train itself and of the passengers on board of the train and k_{1i} , k_{2i} , and k_{3i} are the resistance coefficients of train i . In this way, the running distance and running time of the coasting phase can be calculated. The coasting phase can then be included in the model formulation for the optimization of train schedule. \square

The minimum headway is the minimum time interval between two successive trains so that they can enter and depart from a station safely [98]. A train cannot enter a station until a minimum headway h_0 after the preceding train's departure, which can be formulated as

$$a_{i,j} - d_{i-1,j} \geq h_0. \quad (5.5)$$

In addition, we select the order in which the trains run such that vehicle $i - 1$ always precedes train i for $i \in \{1, 2, 3, \dots, I\}$ with I the total number of trains.

5.2.2 Passenger demand characteristics

The number of passengers still remaining at the station after the departure of train $i - 1$ immediately after its departure at station j is defined as $w_{i-1,j}$. The number of passengers who want to get on train i at station j can then be formulated as

$$w_{i,j}^{\text{wait}} = w_{i-1,j} + \lambda_j(d_{i,j} - d_{i-1,j}), \quad (5.6)$$

where $\lambda_j(d_{i,j} - d_{i-1,j})$ is the number of the passengers that arrived during the departure of train $i - 1$ and the departure of train i .

By defining the number of passengers on train i immediately after its departure at station $j - 1$ as $n_{i,j-1}$, the remaining capacity of train i at station j immediately after the alighting process of the passengers is

$$n_{i,j}^{\text{remain}} = C_{i,\max} - n_{i,j-1}(1 - \rho_j), \quad (5.7)$$

where $C_{i,\max}$ is the effective maximal capacity of train i , and $n_{i,j-1}(1 - \rho_j)$ is the number of passengers remaining on train i immediately after all the passengers that wanted to leave the train have gotten off. Note that the effective maximal capacity can be estimated based on the data from the daily operations, where the distribution of onboard passengers and the effect of the distribution of waiting passengers on the platforms, etc. can be taken into account.

The number of passengers boarding train i at station j is equal to the minimum of the remaining capacity and the number of waiting passengers, i.e.

$$n_{i,j}^{\text{board}} = \min(n_{i,j}^{\text{remain}}, w_{i,j}^{\text{wait}}). \quad (5.8)$$

The number of passengers at station j immediately after the departure of train i , i.e. the passengers who cannot get on train i , is then given by

$$w_{i,j} = w_{i,j}^{\text{wait}} - n_{i,j}^{\text{board}}. \quad (5.9)$$

The number of passengers on train i when it departs from station j is equal to the sum of the passengers arriving but not getting off at station j and the passengers boarding on train i at station j , which can be formulated as

$$n_{i,j} = n_{i,j-1}(1 - \rho_j) + n_{i,j}^{\text{board}}. \quad (5.10)$$

5.2.3 Passenger/vehicle interaction

The dwell time is influenced by the number of alighting and boarding passengers, the number on waiting passengers at platform, and the distribution of passengers on board of a train and of passengers waiting at platform etc. Based on [85], the minimum dwell time can be described by a linear model

$$\tau_{i,j,\min} = \alpha_{1,d} + \alpha_{2,d}n_{i,j-1}\rho_j + \alpha_{3,d}n_{i,j}^{\text{board}} \quad (5.11)$$

or a nonlinear model

$$\tau_{i,j,\min} = \alpha_{1,d} + \alpha_{2,d}n_{i,j}^{\text{alight}} + \alpha_{3,d}n_{i,j}^{\text{board}} + \alpha_{4,d}\left(\frac{w_{i,j}^{\text{wait}}}{n^{\text{door}}}\right)^3 n_{i,j}^{\text{board}}, \quad (5.12)$$

where $\alpha_{1,d}$, $\alpha_{2,d}$, $\alpha_{3,d}$, and $\alpha_{4,d}$ are coefficients that can be estimated based on historical data, n^{door} is the number of doors of the train, and $w_{i,j}^{\text{wait}}/n^{\text{door}}$ is the average number of passengers waiting at each door. The dwell time $\tau_{i,j}$ should satisfy

$$\max(\tilde{\tau}_{\min}, \tau_{i,j,\min}) \leq \tau_{i,j} \leq \tau_{i,j,\max}, \quad (5.13)$$

where $\tilde{\tau}_{\min}$ is the minimum dwell time predefined by railway operator. The dwell time $\tau_{i,j}$ should be larger than the predefined minimum dwell time $\tilde{\tau}_{\min}$ and it should also be larger than the minimal dwell time $\tau_{i,j,\min}$ such that the passengers can get on and get off the train. In addition, $\tau_{i,j}$ should be less than a maximum dwell time $\tau_{i,j,\max}$ to ensure the passenger satisfaction.

5.3 Mathematical formulation of the train scheduling problem

Based on the model formulation in Section 5.2, we now formulate the real-time train scheduling problem. The total travel time of all passengers and the energy consumption caused by the operation of trains are minimized using a weighted-sum strategy for the real-time train scheduling problem. The total travel time is the sum of the passenger waiting time and the passenger in-vehicle time. The passenger waiting time $t_{i,j}^{\text{wait}}$ at station j for train i includes the waiting time of both passengers left by the previous train $i-1$ and the newly arrived passengers, and it can be calculated by

$$t_{i,j}^{\text{wait}} = w_{i-1,j}(d_{i,j} - d_{i-1,j}) + \frac{1}{2}\lambda_j(d_{i,j} - d_{i-1,j})^2, \quad (5.14)$$

where the first term represents the waiting time of the passengers left by train $i-1$ at station j , and the second term represents the waiting time of randomly arriving passengers between the departures of train $i-1$ and train i . The passenger in-vehicle time for train i running from station j to $j+1$ includes the running time for all passengers on train i after its departure from station j and the waiting time of the passengers who do not get off the train at station $j+1$, which can be formulated as

$$t_{i,j}^{\text{in-vehicle}} = n_{i,j}r_{i,j} + n_{i,j}(1 - \rho_{j+1})\tau_{i,j+1}. \quad (5.15)$$

The total travel time of passengers can be described as a weighted sum of the passenger waiting time and the passenger in-vehicle time, i.e.,

$$t_{\text{total}} = \sum_{i=1}^I \sum_{j=1}^{J-1} (\gamma_{\text{wait}} t_{i,j}^{\text{wait}} + t_{i,j}^{\text{in-vehicle}}). \quad (5.16)$$

Remark 5.3 Since the passengers usually feel that time goes slowly when they are waiting at the platform [122], a weight larger than one can be added to the passenger waiting time in the problem formulation. \square

In [48], the energy consumption is proportional to the resistance, which includes the rolling resistance, air resistance, and grade resistance. However, the computation of the energy consumption is more precise in this chapter because the operation model of a train considered here includes the acceleration phase, the speed holding phase, and the deceleration phase. The deceleration phase usually does not consume energy. In addition, if the electric motor on board of a train can be used as an electric generator (so-called regenerative braking), then the regenerative energy can be fed back into the power supply system. The energy consumptions for each phase are then calculated as follows:

- The energy consumption for the acceleration phase of train i on segment j is

$$E_{i,j}^{\text{acc}} = \int_0^{t_{i,j}^{\text{acc}}} \left((m_{e,i} + n_{i,j} m_p) (a_{i,j}^{\text{acc}} + k_{1i} + k_{2i} v(t) + g \sin(\theta_j)) + k_{3i} v^2(t) \right) v(t) dt, \quad (5.17)$$

where $m_{e,i}$ is the mass of train i itself, m_p is the mass of one passenger, $(m_{e,i} + n_{i,j} m_p)$ is the mass of train i and the passengers on board of train i at station j , k_{1i} , k_{2i} , and k_{3i} are the resistance coefficients of train i , $v(t)$ is equal to $a_{i,j}^{\text{acc}} t$, and θ_j is the grade profile of segment j .

- The energy consumption for the speed holding phase of train i on segment j is

$$E_{i,j}^{\text{hold}} = \int_{t_{i,j}^{\text{acc}}}^{t_{i,j}^{\text{acc}} + t_{i,j}^{\text{hold}}} \left((m_{e,i} + n_{i,j} m_p) (k_{1i} + k_{2i} v_{i,j} + g \sin(\theta_j)) + k_{3i} v_{i,j}^2 \right) v_{i,j} dt. \quad (5.18)$$

- The energy generated by the regenerative braking in the deceleration phase can be calculated as

$$E_{i,j}^{\text{dec}} = \beta_i^{\text{dec}} \int_{t_{i,j}^{\text{acc}} + t_{i,j}^{\text{hold}}}^{t_{i,j}} \left((m_{e,i} + n_{i,j} m_p) (|a_{i,j}^{\text{dec}}| - k_{1i} - k_{2i} v(t) - g \sin(\theta_j)) - k_{3i} v^2(t) \right) v(t) dt, \quad (5.19)$$

where β_i^{dec} is the ratio for regenerative energy fed back to the power supply system since there is power loss in electric generator of train i .

The total energy consumption for all I trains running with J stations can then be formulated as

$$E_{\text{total}} = \sum_{i=1}^I \sum_{j=1}^J (E_{i,j}^{\text{acc}} + E_{i,j}^{\text{hold}} - E_{i,j}^{\text{dec}}). \quad (5.20)$$

We apply a weighted-sum strategy to solve the multi-objective optimization of the train scheduling problem, i.e., we consider the following objective function

$$f_{\text{opt}} = \frac{E_{\text{total}}}{E_{\text{total,nom}}} + \gamma \frac{t_{\text{total}}}{t_{\text{total,nom}}}, \quad (5.21)$$

where γ is a non-negative weight, and the normalization factors $E_{\text{total,nom}}$ and $t_{\text{total,nom}}$ are the nominal values of the total energy consumption and the total travel time of passengers, respectively. These nominal values can e.g. be determined by running trains using a feasible initial schedule.

The constraints of the real-time scheduling problem consist of the running time constraints, dwell time constraints, headway constraints, and capacity of trains, given as (5.1)-(5.13) in Section 5.2.

Remark 5.4 The train scheduling problem presented here can be solved in a rolling horizon way. A detailed description on how to do this for train scheduling is given in Section 7.5. See there for more details. \square

5.4 Solution approaches

The resulting train scheduling problem with objective function (5.21) and constraints (5.1)-(5.13) is a non-smooth non-convex problem, where the non-smoothness is caused by the min function in (5.8), and the non-convexity is due to the nonlinear non-convex objective function and the non-convex set defined by constraints. We solve the train scheduling problem using several alternative approaches in this section. One is a gradient-free non-smooth optimization approach, e.g., pattern search. Another one is a gradient-based nonlinear programming approach, e.g., sequential quadratic programming. Furthermore, a general purpose nonlinear integer programming approach, e.g., branch-and-bound algorithm, is also used. By approximating the nonlinear objective function using PWA functions, the train scheduling problem can be recast into an MILP problem. Furthermore, we propose a new iterative convex programming (ICP) approach to solve the train scheduling problem.

5.4.1 Gradient-free nonlinear programming

Nonlinear programming approaches can be grouped in gradient-free approaches and gradient-based approaches. The gradient-free approaches do not explicitly require gradient and Hessian information but only require that the values of the objective function can be ranked. Moreover, gradient-free methods are suitable for non-smooth problems. Since the real-time train scheduling problem is non-smooth due to the min function, our first choice is to use a gradient-free method. Here, in particular we propose to use the pattern search method, since it can handle optimization problems with nonlinear, linear, and bound constraints, and does not require objective functions to be differentiable or continuous. The pattern search method was first proposed by Hooke and Jeeves [61], and it has been proved successful in practice even for objective functions with many local minima, in particular when used in combination with a multi-start method [91], in order to improve the probability of obtaining a solution that is close to a globally optimal solution.

When solving the scheduling problem using the pattern search method, the variables are the departure times $d_{i,j}$, the running times $r_{i,j}$, and the dwell times $\tau_{i,j}$. The other variables, such as the number of passengers $w_{i,j}$ waiting at stations, the number of passengers $n_{i,j}$ on board of the trains, the passenger waiting times $t_{\text{wait},i,j}$, and the passenger onboard times $t_{\text{in-vehicle},i,j}$, can be eliminated. The pattern search method has been implemented in the global optimization toolbox of Matlab [118].

5.4.2 Gradient-based nonlinear programming

Gradient-based nonlinear programming methods rely on gradient or Hessian information. If this information is not available, it can be approximated numerically. We consider the gradient-based sequential quadratic programming (SQP) algorithm here since it is widely used to solve nonlinear programming problems. A requirement for the SQP algorithm is that the objective function and the constraints of the nonlinear programming problem should be continuously differentiable [15]. In the SQP method, the nonlinear programming problem is recast as a sequence of quadratic programming problems, which can be solved easily and efficiently. The train scheduling problem considered in this chapter is non-differentiable because of the min function. Even though the SQP algorithm is a gradient-based method, we also apply it to our problem setting² (since it yields good solutions in the case study in practice in Section 5.6).

When solving the real-time scheduling problem using the SQP algorithm, the variables are the same as those in the pattern search method. The SQP algorithm has been implemented in many packages, such as SNOPT and the optimization toolbox of MATLAB [117].

5.4.3 Mixed integer nonlinear programming

In the MINLP approach, we introduce auxiliary binary variables and auxiliary real variables to deal with the non-smooth min function in (5.8). By introducing a binary variable $\delta_{i,j} \in \{0, 1\}$ and defining

$$\tilde{f}_{i,j} = w_{i-1,j} + \lambda_j(d_{i,j} - d_{i-1,j}) - [C_{i,\max} - n_{i,j-1}(1 - \rho_j)], \quad (5.22)$$

the following equivalence holds [140]:

$$[\tilde{f}_{i,j} \leq 0] \Leftrightarrow [\delta = 1] \quad (5.23)$$

which is true iff

$$\begin{cases} \tilde{f}_{i,j} \leq \tilde{M}_{i,j}(1 - \delta_{i,j}) \\ \tilde{f}_{i,j} \geq \varepsilon + (\tilde{m} - \varepsilon)\delta_{i,j} \end{cases}, \quad (5.24)$$

where ε is a small positive number that is introduced to transform a strict equality into a non-strict inequality, and $\tilde{M}_{i,j}$ and $\tilde{m}_{i,j}$ are the maximum value and the minimum value of $\tilde{f}_{i,j}$, respectively. Equation (5.8) can now be rewritten as

$$n_{i,j}^{\text{board}} = \delta_{i,j}[w_{i-1,j} + \lambda_j(d_{i,j} - d_{i-1,j})] + (1 - \delta_{i,j})[C_{i,\max} - n_{i,j-1}(1 - \rho_j)]. \quad (5.25)$$

²When the SQP algorithm is applied to a nonlinear programming problem with a non-differentiable objective function, it might get stuck in a local solution. In the nonlinear programming problem proposed in this chapter, the minimum value of the objective function is usually not obtained at the points where the objective function is non-differentiable, so the SQP algorithm will jump over these points. Therefore, the SQP approach with multiple initial points works well in this case.

Note that in (5.25) there are four nonlinear terms (i.e., $\delta_{i,j}w_{i-1,j}$, $\delta_{i,j}d_{i,j}$, $\delta_{i,j}d_{i-1,j}$, $\delta_{i,j}n_{i,j-1}$), which are the products of the binary variable $\delta_{i,j}$ and real variables. Four auxiliary real variables can then be introduced to transform these four nonlinear terms into linear terms with linear constraints. The detailed information about this transformation is given in Section 3.4.2.

The variables of the resulting MINLP problem involve the departure times $d_{i,j}$, the running times $r_{i,j}$, and the dwell times $\tau_{i,j}$, the passengers waiting at stations $w_{i,j}$, the passengers on board of the trains $n_{i,j}$, the binary variables $\delta_{i,j}$, and the auxiliary variables. The other variables like the passenger waiting times $t_{\text{wait},i,j}$ and the passenger onboard times $t_{\text{in-vehicle},i,j}$ can be eliminated. The MINLP problem can be solved using branch-and-bound method, such as the MINLP BB solver³ and SCIP (Solving Constraint Integer Programs) [11]. Because the computation time of the MINLP BB solver are too long, so we now propose a bi-level optimization method to solve the MINLP problem. This method consists of two levels of optimization. The high-level optimization optimizes the binary variables, where a brute force approach can be used to find all the combinations for the binary variables when the size of the problem is small. Alternatively, integer programming approaches, such as genetic algorithms, can be applied in the high-level optimization. For each combination of binary variables, the nonlinear optimization problem in the lower level is now a smooth non-convex optimization problem with real-valued variables only, which can be solved using gradient-based approaches, such as an interior point algorithm.

5.4.4 Mixed integer linear programming

In Chapter 3, we have shown that the mixed integer linear programming (MILP) approach can be very efficient for train trajectory planning problems. Therefore, we also apply the MILP approach to the real-time train scheduling problem. In this approach, we approximate the nonlinear terms in the objective function by piecewise affine (PWA) approximations and then transform the non-smooth non-convex problem into an MILP problem. The MILP approach deals with the min function of (5.8) in the same way as the MINLP approach. In the MINLP problem in Section 5.4.3, the constraints are linear, but the objective function is nonlinear and non-convex. Therefore, in order to solve the real-time rescheduling problem as an MILP problem, we need to approximate the nonlinear terms, such as $w_{i-1,j}d_{i,j}$, $n_{i,j}r_{i,j}$, and $n_{i,j}\tau_{i,j}$, as PWA functions. These nonlinear terms are products of two real-valued variables. Here, we use the general form xy to denote such a nonlinear term, which can be rewritten as [77, 139]

$$xy = \frac{1}{4}(x+y)^2 - \frac{1}{4}(x-y)^2. \quad (5.26)$$

Define $\phi = x + y$ and $\xi = x - y$. Then we have $xy = \frac{1}{4}\phi^2 + \frac{1}{4}\xi^2$, where the quadratic terms ϕ^2 and ξ^2 , or z^2 in general notation, can be approximated by a PWA function of the following form:

$$f_{\text{PWA}}(z) = \begin{cases} \alpha_1 z + \beta_1 & \text{for } z \leq Z_1, \\ \alpha_2 z + \beta_2 & \text{for } z > Z_1. \end{cases} \quad (5.27)$$

For each nonlinear term in the objective function, the values of α , β , and Z_1 are optimized based on least-squares optimization (see Section 3.4.2 for more details). By introducing the

³For more details about MINLP BB solver, see <http://tomopt.com>.

binary variables and auxiliary variables, the product xy can be recast as a linear expression with linear constraints (see Section 3.4.2 for more information).

The variables of the resulting MILP problem include the variables in the MINLP problem and the binary variables and the auxiliary variables introduced by the approximations of nonlinear terms in the objective function. The MILP problem can be solved by branch-and-bound algorithms implemented in several existing commercial and free solvers, such as CPLEX, Xpress-MP, or GLPK [5, 86].

5.4.5 A new approach: iterative convex programming

The non-differentiability of the train scheduling problem is introduced by the min function in (5.8). In addition, the non-convexity of the problem is caused by the variables $w_{i,j}$ and $n_{i,j}$. Therefore, we propose the iterative convex programming (ICP) approach, where we use estimated values $\hat{w}_{i,j}$ and $\hat{n}_{i,j}$ for $w_{i,j}$ and $n_{i,j}$, respectively. This eliminates the min function and the nonlinear terms $w_{i,j}d_{i,j}$, $w_{i,j}d_{i-1,j}$, $n_{i,j}r_{i,j}$ and $n_{i,j}\tau_{i,j+1}$ in the objective function. Hence, the resulting optimization problem is a smooth and convex problem, which can be solved for the global optimum using e.g. interior point algorithms [16] and ellipsoid algorithm [109], which are implemented in the Matlab software CVX for convex programming⁴. Based on the optimum of the convex problem, the new estimated values for $w_{i,j}$ and $n_{i,j}$ can be calculated. By solving the convex problems iteratively, an approximation of the global optimum of the original non-smooth non-convex problem can be obtained. The procedure of the ICP method is given in Algorithm 1.

Algorithm 1 ICP method

- 1: **Input** : a feasible initial solution of departure times, running times, and dwell times, i.e., $d_{i,j}(0)$, $r_{i,j}(0)$, and $\tau_{i,j}(0)$ for $i = 1, \dots, I$ and $j = 1, \dots, J$, p_{\max} , convergence tolerance ς , maximum number of iterations p_{\max} ;
 - 2: iteration index $p \leftarrow 0$;
 - 3: calculate initial estimates $\hat{w}_{i,j}(p)$ and $\hat{n}_{i,j}(p)$ using (5.9) and (5.10) based on $d_{i,j}(p)$, $r_{i,j}(p)$, and $\tau_{i,j}(p)$;
 - 4: calculate the initial objective $f_{\text{opt}}(p)$ using (5.21) ;
 - 5: **Repeat**
 - 6: $p = p + 1$;
 - 7: substitute the estimates $\hat{w}_{i,j}(p-1)$ and $\hat{n}_{i,j}(p-1)$ into the original problem to get a convex problem;
 - 8: obtain optimal departure time $d_{i,j}^*(p)$, running time $r_{i,j}^*(p)$, and dwell time $\tau_{i,j}^*(p)$ by solving the convex problem;
 - 9: compute $\hat{w}_{i,j}(p)$ and $\hat{n}_{i,j}(p)$ using (5.9) and (5.10) based on $d_{i,j}^*(p)$, $r_{i,j}^*(p)$, and $\tau_{i,j}^*(p)$;
 - 10: calculate the objective $f_{\text{opt}}(p)$ using (5.21);
 - 11: **Until** $p = p_{\max}$ or $|f_{\text{opt}}(p) - f_{\text{opt}}(p-1)| \leq \varsigma$
 - 12: $p^* = \arg \min_p f_{\text{opt}}(p)$;
 - 13: **Output** : $d_{i,j}^*(p^*)$, $r_{i,j}^*(p^*)$, $\tau_{i,j}^*(p^*)$, $f_{\text{opt}}(p^*)$.
-

⁴More information about Matlab software CVX, see <http://cvxr.com/cvx>.

For the ICP approach, the variables of the real-time scheduling problem are the departure times $d_{i,j}$, the running times $r_{i,j}$, and the dwell times $\tau_{i,j}$. The number of passengers $w_{i,j}$ waiting at stations and the number of passengers $n_{i,j}$ on board of the trains are estimated by $\hat{w}_{i,j}$ and $\hat{n}_{i,j}$, respectively. The other variables, such as the passenger waiting time $t_{\text{wait},i,j}$ and the passenger onboard times $t_{\text{in-vehicle},i,j}$, are functions of the decision variables and hence are not explicitly represented in the solution process. The solution obtained by the ICP approach is not necessarily the global minimum of the formulated scheduling problem since the ICP approach solves a sequence of convex approximations of the formulated nonlinear non-convex problem. For the ICP approach, we should in general use multiple starting points. However, for the case study in Section 5.6, we found that one random feasible initial point yields comparable results with respect to other alternative approaches.

5.5 Extension: stop-skipping at small stations

In order to reduce the passenger travel time and energy consumption further, a stop-skipping scheme can be adopted, which has already been applied in practice, such as the SEPTA line in Philadelphia and the urban rail transit system in Santiago, Chile (see Chapter 2 for more detailed information). The stop-skipping strategy is beneficial for both the rail operator (e.g., less energy consumption and higher operation speed) and the passengers (e.g., shorter travel time and lower train occupation). With the help of the information provided via personal digital assistant (PDA) devices and the real-time displays and announcements at stations, we assume that passengers can obtain enough information and board the right train. However, the passengers at the skipped stations experience longer waiting time and thus a longer total travel time. Therefore, the skipping of trains at stations should be carefully coordinated. For example, additional constraints can be considered when scheduling the trains, such as: two successive trains should not skip the same station. In this way, the waiting time of passengers can be limited to an acceptable value.

In practice, if the passenger alighting proportion and the passenger arrival rate are high at a station, a train will not skip that station. Therefore, we assume here that trains may only skip small stations with low passenger alighting proportion ρ_j and that the effect of stop-skipping on passengers is then negligible for these small stations. Hence we define a skipping set $S_{\text{skip}} = \{(i, j) | \text{train } i \text{ may potentially skip station } j\}$.

We introduce a binary variable $y_{i,j}$ to indicate whether train i stops at station j or not:

$$y_{i,j} = \begin{cases} 1 & \text{if train } i \text{ will stop at station } j, \\ 0 & \text{if train } i \text{ will skip station } j. \end{cases} \quad (5.28)$$

For the sake of simplicity of the expression, we consider here the variables $y_{i,j}$ to be defined for pairs $(i, j) \in \{1, 2, \dots, I\} \times \{1, 2, \dots, J\}$. However, actually $y_{i,j}$ is only a free variable if $(i, j) \in S_{\text{skip}}$ and otherwise $y_{i,j} = 1$ by definition. Hence, instead of (5.1) we get

$$d_{i,j} = a_{i,j} + y_{i,j} \tau_{i,j}. \quad (5.29)$$

Since train i may skip station j or station $j+1$, the running distance of the speed holding phase is then rewritten as

$$s_{i,j}^{\text{hold}} = s_j - y_{i,j} \frac{v_{i,j}^2}{2a_{i,j}^{\text{acc}}} + y_{i,j+1} \frac{v_{i,j}^2}{2a_{i,j}^{\text{dec}}}, \quad (5.30)$$

which means that if train i skips station j , then train i will run with the holding speed $v_{i,j}$ in the running distance of the acceleration phase. Similarly, if train i skips station $j+1$, train i will run with the holding speed $v_{i,j}$ in the distance of the deceleration phase. Note that we have

$$(1 - y_{i,j+1})(v_{i,j+1} - v_{i,j}) = 0, \quad (5.31)$$

since when train i skips station $j+1$, i.e., $y_{i,j+1} = 0$, the operation of the train between station j and station $j+2$ only contains three phases.

The running time of train i for segment j can be written as

$$r_{i,j} = \frac{s_j}{v_{i,j}} + y_{i,j} \frac{v_{i,j}}{2a_{i,j}^{\text{acc}}} - y_{i,j+1} \frac{v_{i,j}}{2a_{i,j}^{\text{dec}}}. \quad (5.32)$$

The remaining capacity of train i at station j immediately after the passengers alight is reformulated as

$$n_{i,j}^{\text{remain}} = C_{i,\text{max}} - n_{i,j-1}(1 - y_{i,j}p_j)$$

instead of (5.7). Instead of (5.8), the number of passengers boarding train i at station j can be calculated using

$$n_{i,j}^{\text{board}} = \min(n_{i,j}^{\text{remain}}, y_{i,j}w_{i,j}^{\text{wait}}), \quad (5.33)$$

where $y_{i,j}w_{i,j}^{\text{wait}}$ is the number of passengers who want to get on train i at station j . Furthermore, the number of passengers at station j immediately after the departure of train i can be computed by (5.9). Instead of (5.10), the number of passengers on train i when it departs from station j is now reformulated as

$$n_{i,j} = n_{i,j-1}(1 - y_{i,j}p_j) + n_{i,j}^{\text{board}}. \quad (5.34)$$

For the train scheduling problem with stop-skipping, the total energy consumption should be calculated as

$$E_{\text{total}} = \sum_{i=1}^I \sum_{j=1}^J (y_{i,j}E_{i,j}^{\text{acc}} + E_{i,j}^{\text{hold}} - y_{i,j+1}E_{i,j}^{\text{dec}}) \quad (5.35)$$

instead of (5.20). In addition, the passenger in-vehicle time for train i running from station j to $j+1$ is reformulated as

$$t_{\text{in-vehicle},i,j} = n_{i,j}r_{i,j} + y_{i,j+1}n_{i,j}(1 - p_{j+1})\tau_{i,j+1} \quad (5.36)$$

instead of (5.15). The total travel time for the stop-skipping problem can then be calculated by (5.16).

The train scheduling problem with stop-skipping has objective function (5.21) and constraints (5.2), (5.4)-(5.9), and (5.29)-(5.14). This optimization problem is an MINLP problem, where the stopping variables $y_{i,j}$ are the binary variables and the objective function are nonlinear and nonconvex. The MINLP approach and the MILP approach in Section 5.4.3 and Section 5.4.4 can be directly applied to solve the train scheduling problem with stop-skipping since they can easily deal with binary variables.

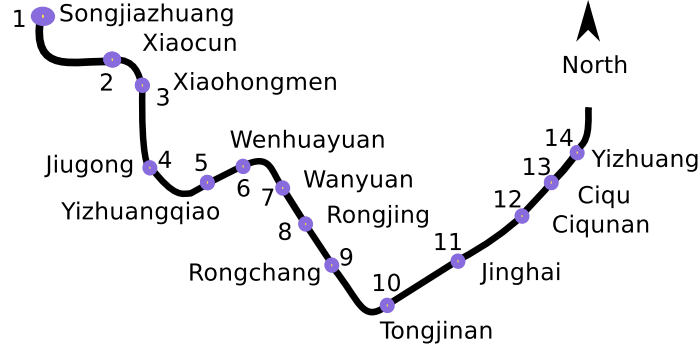


Figure 5.3: The layout of the Yizhuang subway line

5.6 Case Study

5.6.1 Set-up

In order to demonstrate the approaches proposed in Section 5.4 for the real-time train scheduling problem (with stop-skipping) and to compare their performance, the train characteristics and the line data of the Yizhuang subway line in Beijing are used as a test case study. The Yizhuang line has 14 stations as shown in Figure 5.3, and the speed limit for the line is 80 km/h (i.e., 22.2 m/s). The detailed information about these 14 stations is listed in Table 5.1. The minimum running time in Table 5.1 is calculated by taking a fixed acceleration of 0.8 m/s^2 and a fixed deceleration of -0.8 m/s^2 . Furthermore, the speed $v_{i,j}$ of the holding phase in (5.3) is taken to be equal to the maximum speed 22.2 m/s for computing the minimum running time. We assume the maximum running time is $r_{i,j,\max} = \zeta r_{i,j,\min}$, where ζ is larger than 1. We have chosen $\zeta = 1.2$ in order to ensure that the passengers do not complain that the train is too slow. Based on the maximum running time, the minimal holding speed can be calculated.

The mass of the train itself and the standard mass of one passenger are given in Table 5.2. In addition, we choose the linear model (5.11) for the minimum dwell time, the coefficients of which given in Table 5.2 are chosen according to [145]. The maximal dwell time is chosen as 150 s. The effective capacity of each train is 1468 passengers. A communication-based train control system (i.e., a moving block signaling scheme) is implemented in the Beijing Yizhuang subway line and the minimum headway h_0 between two successive trains is 90 s.

The initial state of the network at the start of the simulation is defined as follows: the train scheduling at the beginning of a day is considered. The schedule of train 0, i.e., the first train entering the urban rail transit line, is fixed. There are no passengers left by train 0 because not too many passengers wait for the first train in the morning. The schedule for the following trains should be optimized in this case study. Furthermore, in order to compare the performance of the schedules obtained by different approaches proposed in this chapter, a reference schedule with a fixed departure headway defined, which is a state-of-the-art method used in practice, where the departure headway could change several times a day based on the peak hours and off-peak hours. For the scheduling period considered in this

Table 5.1: Information of the Yizhuang subway line

Station number	Distance to next station [m]	Passenger arrival rate [passenger/s]	Passenger alighting proportion[-]	Minimum running time [s]
1	1332	3	0	87.721
2	1286	0.5	0.05	85.651
3	2086	3	0.3	121.654
4	2265	4	0.38	129.710
5	2331	0.4	0.04	132.680
6	1354	4	0.32	88.711
7	1280	4	0.38	85.380
8	1544	3	0.7	97.260
9	992	3	0.6	72.420
10	1975	3	0.7	116.659
11	2369	3	0.7	134.391
12	1349	2	0.5	88.486
13	2610	2	0.5	145.237
14	-	0	1	-

Table 5.2: parameters of the trains and the passengers

Property		Symbol	Value
Train mass	[kg]	$m_{e,i}$	$199 \cdot 10^3$
Mass of one passenger	[kg]	m_p	60
Capacity of trains	[passengers]	$C_{i,max}$	1468
Minimum dwell time	[s]	$\tilde{\tau}_{min}$	30
Maximum dwell time	[s]	τ_{max}	150
Coefficients of the minimal dwell time	[s]	$\alpha_{1,d}$	4.002
	[s/passengers]	$\alpha_{2,d}$	0.047
	[s/passengers]	$\alpha_{3,d}$	0.051
Coefficients of resistance	[m/s ²]	k_{1i}	0.012
	[s ⁻¹]	k_{2i}	$5.049 \cdot 10^{-4}$
	[m ⁻¹]	k_{3i}	$2.053 \cdot 10^{-5}$

case study, the schedule of trains is taken the same as train 0 and the constant departure headway is taken as 210 s (i.e., the sum of the minimum headway and the maximum dwell time).

Two cases will be considered here:

- Case A: the train scheduling problem without stop-skipping.
- Case B: the train scheduling problem with stop-skipping.

For Case A, we have applied the approaches proposed in Section 5.4 to solve the train scheduling problem. For the pattern search method, we used the patternsearch function in

Table 5.3: The nominal values of the energy consumption and the passenger travel time

Scenario index	I & J	Nominal passenger travel time [s]	Nominal energy consumption [J]
1	$I = 2, J = 3$	$6.402 \cdot 10^5$	$1.216 \cdot 10^8$
2	$I = 3, J = 4$	$1.954 \cdot 10^6$	$3.285 \cdot 10^8$
3	$I = 4, J = 5$	$6.457 \cdot 10^6$	$4.780 \cdot 10^8$
4	$I = 5, J = 6$	$7.211 \cdot 10^6$	$1.402 \cdot 10^9$
5	$I = 6, J = 7$	$1.582 \cdot 10^7$	$1.992 \cdot 10^9$
6	$I = 7, J = 8$	$2.537 \cdot 10^7$	$1.943 \cdot 10^9$
7	$I = 7, J = 10$	$2.943 \cdot 10^7$	$2.859 \cdot 10^9$
8	$I = 7, J = 12$	$3.523 \cdot 10^7$	$2.557 \cdot 10^9$
9	$I = 7, J = 14$	$3.298 \cdot 10^7$	$4.926 \cdot 10^9$

the global optimization toolbox of Matlab. The SNOPT solver implemented in the Matlab Tomlab toolbox is adopted for the SQP algorithm to solve the nonlinear non-convex train scheduling problem. In the ICP approach, the resulting smooth and convex problem is also solved by the SNOPT. The MINLP BB solver is used for the MINLP approach. In addition, for the bi-level approach for MINLP problem, the low-level optimization problem is also solved using SNOPT and the ga function in the global optimization toolbox of Matlab is applied for the high-level optimization. Furthermore, we use CPLEX, implemented through the cplex interface function of the Matlab Tomlab toolbox as MILP solver.

Remark 5.5 SNOPT, MINLP BB and CPLEX are implemented in object code while the patternsearch and ga functions are implemented in pure m files, i.e., they need to be interpreted and thus usually seem slower than the functions implemented in object code. So, in principle, the current computation time comparison is not fair. However, since we limited ourselves to the methods and functions available to us, we did our best to compare the results of these approaches in the fairest possible way. Note however that as explained above the computation time for the patternsearch and ga functions would be improved if they would also be implemented in object code. \square

In order to illustrate the performance of the solution approaches proposed for different-sized problems, we considered 9 scenarios with different problem sizes as shown in Table 5.3, where the values of I and J are the number of trains and stations involved. For the scenarios with J less than 14, the passenger arrival rate λ_J and the passenger alighting rate ρ_J at station J is set to 0 and 1, respectively, because we assume that station J is the last station of the trip. The weight γ in the multi-objective function (5.21) is chosen as 1. In addition, the nominal values of the passenger travel time and the energy consumption shown in Table 5.3 are obtained using feasible train schedules, where the running times of trains are uniformly distributed random values in $[r_{\min}, 1.2 \cdot r_{\min}]$ and the dwell times are constants.

In Case B, we have applied the MINLP approach and the MILP approach to solve the train scheduling problem with stop-skipping based on Section 5.4. The resulting MINLP problem is solved using a direct MINLP approach, the brute force bi-level method, and the bi-level method with a genetic algorithm for the high-level binary optimization. The corresponding solvers and settings of these methods are chosen the same as those mentioned

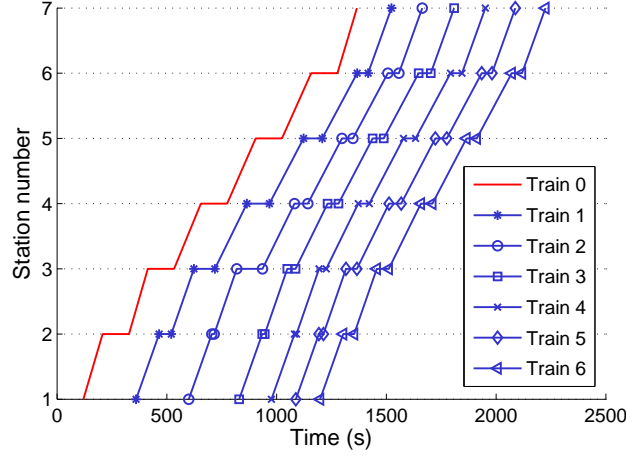


Figure 5.4: The train schedule of scenario 5 with 6 trains and 7 stations for Case A obtained by the sequential quadratic programming approach

for Case A. According to the information given in Table 5.1, station 2 and station 5 are small stations since the passenger arrival rate and the passenger alighting proportion are smaller compared with other stations. In this case study, we allow trains to skip station 2 or/and station 5.

5.6.2 Results and discussion

Results of Case A: the train scheduling problem

The schedule of trains for scenario 5 obtained by solving the train scheduling problem for 6 trains (i.e., train $i \in \{1, 2, \dots, 6\}$) and 7 stations (i.e., station $j \in \{1, 2, \dots, 7\}$) using the SQP approach is shown in Figure 5.4. The running times, dwell times, and critical headways corresponding to Figure 5.4 are shown in Table 5.4. The model formulation in this chapter allows the presence of waiting passengers at platforms at the beginning of the scheduling period and allows trains to be running somewhere on the transit line. As we can observe from Figure 5.4, the departure headways between train 1, train 2, and train 3 at station 1 are larger than those between the later trains. This is because of the schedule of train 0, which stops at each station with a dwell time of 120 s. Therefore, in order to satisfy the headway constraints at all stations, the departure headway at the station 1 must be much larger than the minimum headway 90 s.

We applied a multi-start pattern search and a multi-start SQP here and we started the calculation with 10 feasible random initial points. As regards the ICP algorithm, we should also solve it using multiple initial points. However, we saw that the random feasible initial points yielded a comparable result with respect to each other. In addition, this result is also quite close to the solutions obtained by the other approaches. Therefore, we use one feasible random initial point for the ICP approach in this case study. When we solve the MINLP problem using the bi-level optimization approach, the `fmincon` function in the lower

Table 5.4: The total running times, dwell times, and critical minimum headways of each train for the train schedule of scenario 5 with 6 trains and 7 stations for Case A obtained by the sequential quadratic programming approach

Train	0	1	2	3	4	5	6
Total running time [s]	646.1	775.4	775.4	775.4	775.4	775.4	775.4
Total dwell time [s]	720	538.1	438.3	343.4	256.3	246.0	272.1
Minimum headway [s]	-	90	90	90	90	90	90

optimization is executed for 10 feasible random initial points. For the MILP approach, only one feasible random initial point is needed to obtain the global minimum of the MILP problem.

The control performance f_{opt} and the computation time of these methods for the 9 scenarios are shown in Figures 5.5 and 5.6. The value of the objective function is influenced by the nominal values and weights in (5.21). A smaller value of the objective function means a better performance since we solve a minimization problem. Note that the control performance of the MILP approach is calculated using the original nonlinear objective function based on the obtained optimal results. We set the upper bound of the computation time as 5 hours on a 1.8 GHz Intel Core2 Duo CPU running a 64-bit Linux operating system. If the computation cannot finish within 5 hours, no results are reported, so we cannot determine the control performance index. In order to visualize the scenarios of which the computation cannot finish within 5 hours, we have set the total performance index of these scenarios larger than 3.5 as shown in Figure 5.5 and set the computation time larger than 4×10^4 s as shown in Figure 5.6. It is noted that only the performance indices of the SQP algorithm and the ICP algorithm are reported for scenario 9 in Figure 5.5, and the calculation of other algorithms for scenario 9 cannot finish within 5 hours. It is observed that the reference schedule has the worst control performance but also the lowest computation time. In addition, the performance of the MILP approach is worse compared with the other solution approaches that yield similar control performance. The computation time of the ICP approach and the SQP approach is less than 15 s and 120 s for scenario 9, respectively. In addition, we can observe that the function values of the ICP approach are mostly quite close to those of the solutions of the SQP approach, where for 7 scenarios the relative performance difference is less than 5% and for the other two scenarios the relative difference is around 10%.

Results for Case B: the scheduling problem with stop-skipping at small stations

The schedule of trains with stop-skipping for scenario 5 with 6 trains and 7 stations obtained by the bi-level method with a genetic algorithm for the high-level optimization is shown in Figure 5.7. The corresponding running times, dwell times, and critical headways are shown in Table 5.5. Note that the normalize rate in (5.21) is the same for both Case A and Case B. It can be observed that train 3 and train 5 skip station 2 and train 2 and train 5 skip station 5 since trains are allowed to skip the small stations 2 and 5. Furthermore, the headways between trains at stations are influenced by the skipping of trains at station 2 and station 5 in order to satisfy the headway constraints at all stations.

The comparison of the performance index for Case A and Case B, obtained by the bi-

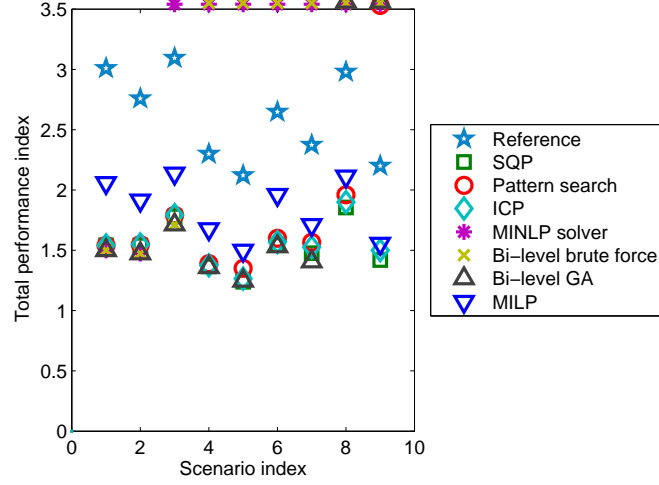


Figure 5.5: Performance comparison of the solution approaches for the real-time train scheduling problem for Case A. In order to visualize the scenarios of which the computation cannot finish within 5 hours, the performance index f_{opt} calculated using (5.21) of these scenarios is set larger than 3.5.

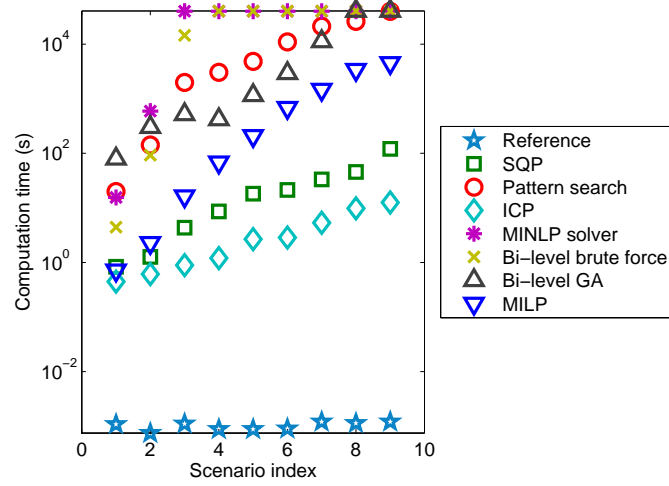


Figure 5.6: The computation time of the solution approaches for the real-time train scheduling problem for Case A. In order to visualize the scenarios of which the computation cannot finish within 5 hours, the computation time of these scenarios is set larger than 4×10^4 s.

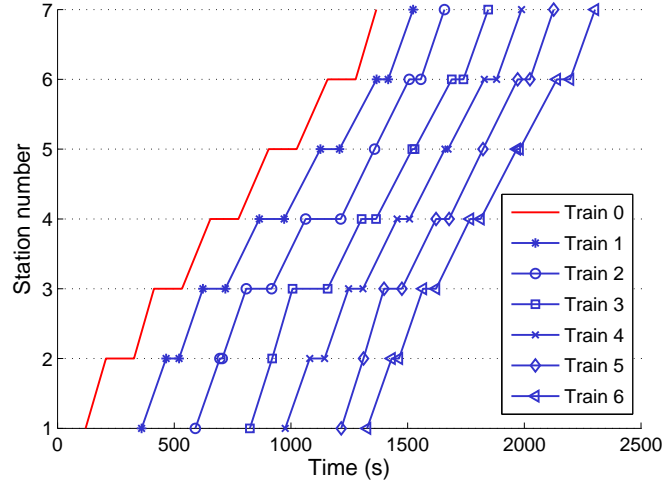


Figure 5.7: The train schedule of scenario 5 with 6 trains and 7 stations for Case B obtained by the bi-level optimization method with a genetic algorithm for the high-level optimization

Table 5.5: The total running times, dwell times, and critical minimum headways of each train for the train schedule of scenario 5 with 6 trains and 7 stations for Case B obtained by the bi-level optimization method with a genetic algorithm for the high-level optimization

Train	0	1	2	3	4	5	6
Total running time [s]	646.1	775.4	749.2	750.5	767.9	724.3	775.4
Total dwell time [s]	720	538.1	465.5	420.0	306.1	347.9	221.8
Minimum headway [s]	-	90	90	90	90	90	90

level optimization approach with a genetic algorithm for the high-level integer optimization, is given in Table 5.6. It is observed that the performance indices of Case B, i.e. the train scheduling problem with stop-skipping, are better (i.e., a lower performance index means a better performance) than those of Case A, i.e. the train scheduling problem without stop-skipping, in general. The energy consumption is reduced since some trains do not need to decelerate and accelerate again at those small stations. Skipping some small stations may reduce the travel time of most passengers due to the zero dwell time at small stations and lower running times, however, it will increase the travel time of these passengers influenced by the skipping of trains. Therefore, the skipping of trains should be carefully coordinated.

The performance index and the computation time of the MINLP approach and the MILP approach for the 9 scenarios with stop-skipping are shown in Figures 5.8 and 5.9. The reference schedule is same as Case A and it has the lowest computation time and the worst performance. The performance index of the MILP approach is calculated using the original nonlinear objective function based on the obtained optimal results. Due to the errors introduced by the PWA approximations of nonlinear terms in the objective function, the performance index of the MILP approach is much higher than other solution methods. The performance of the direct MINLP approach, the brute force bi-level optimization, and the bi-level optimization with a genetic algorithm for the real-time train scheduling problem with stop-skipping is similar to each other. However, we only report the results of scenario 1 and 2 for the brute force bi-level approach and the results of scenarios 1-3 for the direct MINLP approach since the computation for other scenarios did not finish within 5 hours. In addition, the bi-level optimization approach with a genetic algorithm for the high-level optimization did not finish the calculations within 5 hours for scenarios 8 and 9. Therefore, the performance indices of the MINLP approach and the MILP approach are set higher than 3.5 in Figure 5.8. It is observed that the MILP approach needs less computation effort but at the cost of much less optimal performance. The bi-level optimization methods with a genetic algorithm require a longer computation time, but they yield a better performance.

Discussion

For the train scheduling problem in Case A, the performance of the results obtained by the ICP approach is close to the best results obtained by other alternative approaches. In addition, the computation time of the ICP approach is smaller than that of other alternative approaches except the reference schedule that has a bad performance. Therefore, for the given cases studies the ICP approach produces the best trade-off between performance and computational complexity.

Based on the simulation results for Case B, the bi-level optimization with a genetic algorithm is recommended for solving the train scheduling problem with stop-skipping. Using parallel processing, the computation of this approach would still be tractable for small-sized problems (up to, say 20 stations and 10 trains). However, this approach is too slow for large-scale real-time applications. So in the future, new approaches need to be investigated to solve the train scheduling problem with stop-skipping efficiently.

Remark 5.6 Stop-skipping strategy could result in shorter circulation time of trains; so the turn-around operation at terminal stations may become critical for the operation of trains. However, the turn-around time can be reduced further with the increasing automation of the operation of trains, e.g., automatic turn-around of trains. \square

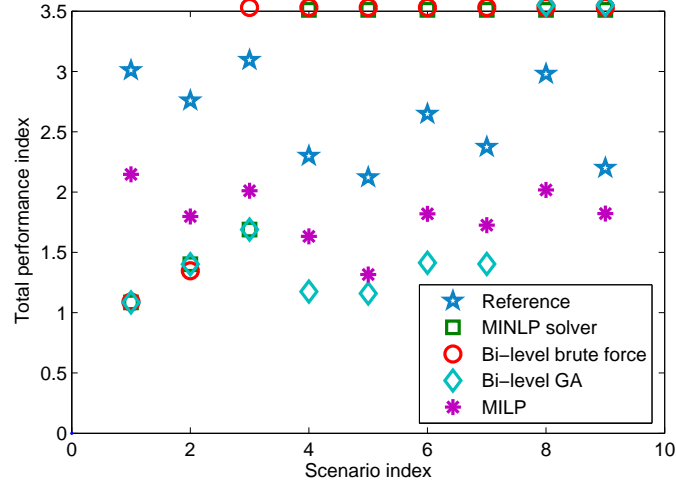


Figure 5.8: Performance comparison of the solution approaches for the real-time train scheduling problem with stop-skipping for Case B. In order to visualize the scenarios of which the computation cannot finish within 5 hours, the performance index f_{opt} calculated using (5.21) of these scenarios is set larger than 3.5.

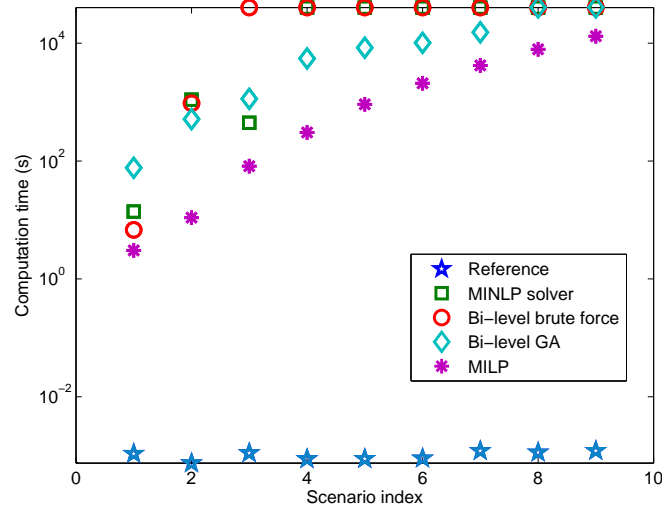


Figure 5.9: The computation time of the solution approaches for the real-time train scheduling problem with stop-skipping for Case B. In order to visualize the scenarios of which the computation cannot finish within 5 hours, the computation time of these scenarios is set larger than 4×10^4 s.

Table 5.6: Comparison of the total performance index for train scheduling with and without stop-skipping using the bi-level optimization with a genetic algorithm for the high-level integer optimization

Scenario		1	2	3	4	5	6	7	8	9
Performance index [-]	Case A	1.500	1.473	1.713	1.360	1.246	1.532	1.407	–	–
	Case B	1.086	1.402	1.689	1.175	1.159	1.413	1.404	–	–
Total energy consumption [J]	Case A	$1.328 \cdot 10^8$	$3.379 \cdot 10^8$	$6.589 \cdot 10^8$	$1.102 \cdot 10^9$	$1.602 \cdot 10^9$	$2.165 \cdot 10^9$	$2.673 \cdot 10^9$	–	–
	Case B	$8.421 \cdot 10^7$	$2.933 \cdot 10^8$	$6.231 \cdot 10^8$	$8.665 \cdot 10^8$	$1.396 \cdot 10^9$	$1.902 \cdot 10^9$	$2.455 \cdot 10^9$	–	–
Total travel time [s]	Case A	$2.610 \cdot 10^5$	$8.683 \cdot 10^5$	$2.161 \cdot 10^6$	$4.137 \cdot 10^6$	$6.993 \cdot 10^6$	$1.060 \cdot 10^7$	$1.388 \cdot 10^7$	–	–
	Case B	$2.515 \cdot 10^5$	$9.951 \cdot 10^5$	$2.487 \cdot 10^6$	$4.017 \cdot 10^6$	$7.247 \cdot 10^6$	$1.104 \cdot 10^7$	$1.605 \cdot 10^7$	–	–

5.7 Conclusions

In this chapter, the train scheduling problem with OD-independent passenger demands for an urban rail transit line has been investigated. We have proposed a new iterative convex programming (ICP) approach to solve this train scheduling problem. In addition, we have also considered other solution approaches, i.e., a gradient-free nonlinear programming approach (in particular pattern search method), a gradient-based nonlinear programming approach (in particular sequential quadratic programming (SQP)), a mixed integer nonlinear programming (MINLP) approach (which includes 3 submethods, i.e., an MINLP approach, brute force bi-level optimization, and a bi-level optimization approach with a genetic algorithm), and a mixed integer linear programming (MILP) approach. Furthermore, the train scheduling model with stop-skipping has been formulated by introducing binary variables to indicate whether a train stops at a station or not. The MINLP approach and the MILP approach are applied to solve this scheduling problem since they can handle integer variables.

The simulation results for train scheduling without stop-skipping have shown that the optimal solutions obtained by the ICP approach, the pattern search method, the SQP approach, and the MINLP approach are close to each other. However, the ICP approach can provide a better trade-off between performance and computational complexity. Furthermore, for the train scheduling problem with stop-skipping, the experiment results have shown that the control performance of the MILP approach is worse than that of the MINLP approach. Among the 3 submethods of the MINLP approach, the bi-level approach with a genetic algorithm offers the best trade-off between performance and computational efficiency.

In the future, one could investigate the effect of more detailed models (modeling the operation of trains in terminals, short turns, the stochastic variability in passenger flow, the distribution of onboard passengers and waiting passengers at platforms, the passengers appearing at platforms after the last train has passed, the passenger flows as described by origin-destination matrices) on the trade-off between performance and computational complexity. In addition, decomposition approaches can be developed to solve large-size instance of the train scheduling problem. Moreover, an extensive comparison and assessment of the approaches proposed in this chapter for a wide range of set-ups and scenarios could also be a topic for future work.

Chapter 6

OD-Dependent Train Scheduling for an Urban Rail Transit Line

In this chapter, in order to capture more detailed information about passengers we consider train scheduling with origin-destination-dependent (OD-dependent) passenger demands for an urban rail transit line. A stop-skipping strategy is adopted to reduce the total passenger travel time and the energy consumption. The resulting train scheduling problem is a mixed integer nonlinear programming problem. A bi-level approach and an efficient bi-level approach are proposed to solve this problem. These two approaches are compared through a case study inspired by real data from the Beijing Yizhuang line.

The results discussed in this chapter have been presented in [136, 137].

6.1 Introduction

In Section 5.5, the train scheduling model has been extended such that trains can skip small stations with low passenger alighting proportions while assuming that the effect of skipping is negligible. Because *OD-independent* passenger demands are used in Chapter 5, the effect on the passengers who have the skipped stations as their destination is not taken into account in Section 5.5. Therefore, in this chapter, a train scheduling model based on *OD-dependent* passenger demands is presented. Since OD-dependent passenger demands vary significantly along the urban rail transit line and the time of the day, e.g., some stations (e.g., those in the central business district) may have a relatively large number of passengers boarding and alighting and others may have few passengers, fixed all-stop train schedules cannot efficiently satisfy such OD-dependent demand patterns. Hence, a dynamic stop-skipping strategy (i.e., the stop-skipping stations for each train are not fixed before the optimization process) based on OD-dependent passenger demands is important for both passengers and rail operators.

This chapter is structured as follows. Section 6.2 formulates the operation of trains at the terminal station, at stations, and in between stations, and the passenger demand characteristics. Section 6.3 describes the multi-objective cost function of the train scheduling problem with stop-skipping. Section 6.4 proposes a bi-level optimization approach and an

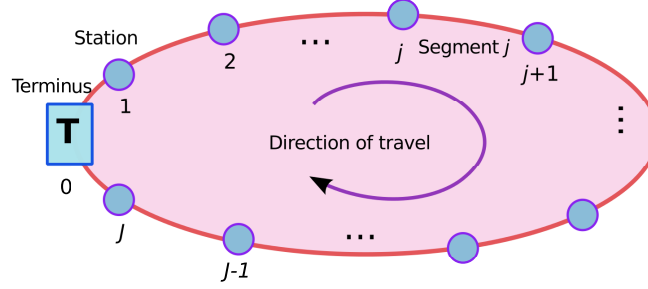


Figure 6.1: Illustration of the urban rail transit line

efficient bi-level approach for the train scheduling problem with stop-skipping. Section 6.5 illustrates the performance of the proposed solution approaches with a case study. Finally, Section 6.6 concludes the chapter.

6.2 Model formulation

This chapter considers an urban rail transit line as shown in Figure 6.1, where the terminal station and stations in the line are numbered increasingly. Let J denote the total number of stations (terminal station not included). The index of the terminal station is set equal to 0. The track section between station j and station $j+1$ is denoted as segment j . The scheduling time period for the train scheduling problem is denoted as $[t_0, t_{\text{end}}]$. In order to distinguish the different running cycles of the physical trains, so called train services are introduced, where the train service number in a unique way identifies a train and its current cycle. After the arrival of a physical train at the terminal station, its service number will be augmented when the train departs. More specifically, the transit line has I physical trains in total, which are numbered as train $1, 2, \dots, I$. However, the service number of trains increases with the cycle of the operation of trains. During the scheduling period, the service number of trains is $1, 2, \dots, I, I+1, I+2, \dots, 2I, \dots, N_{\text{cyc}}I$, where N_{cyc} is number of the cycles of the operation of trains for the given time period $[t_0, t_{\text{end}}]$. The service number of a train is increased with I when the train departs from the terminal station. Therefore, train service i corresponds to physical train $[(i-1) \bmod I] + 1$. For the sake of simplicity, we use “train i ” as a short-hand for “train service i ” from now on.

We make the following assumptions¹ for the terminal station and the stations:

- A.1 Multiple trains can be present at terminal station 0, which has a maximum capacity C_0^{ter} . In addition, the trains in terminal station 0 will depart from the terminal station in a first-in-first-out manner.
- A.2 Station j for $j \in \{1, 2, \dots, J\}$ can only accommodate one train at a time and no overtaking can occur at any point of the line.
- A.3 Trains can skip some stations in the urban transit line, where we define the skipping set $S_{\text{skip}} = \{(i, j) \mid \text{train } i \text{ may potentially}^2 \text{ skip station } j\}$.

¹Assumption A.2, A.5, and A.7 are also considered in Chapter 5.

²A binary variable will be introduced to indicate whether train i stops at station j or skips station j (see Section

- A.4 In view of pre-announcement for passengers about the stop-skipping schedule, there is detailed stop-skipping information displayed in the station and/or the urban rail transit operator provides this information to passengers through mobile devices.
- A.5 The operation of trains only consists of three phases: the acceleration phase, the speed holding phase, and the deceleration phase. Moreover, the acceleration and the deceleration are taken to be fixed constants.
- A.6 Each passenger only takes one train to arrive at his/her destination, i.e., the transfer between different trains along the line is not allowed.
- A.7 The number of passengers waiting at a station and the number of passengers on board immediately after a train's departure are approximated by real numbers.

Assumption A.1 can be motivated as follows: multiple trains can be present at the terminal station since we assume that there are multiple track sections in the terminal station. Furthermore, a first-in-first-out operation for trains in the terminal station is not difficult to realize in practice since it depends on the dispatching of trains in the terminal station and it is a matter of renumbering of trains. Assumption A.2 generally holds for most urban transit systems, which are usually operated in this way [36, 125]. Even though the stop-skipping strategy is not yet widely used in urban rail transit networks throughout the world, there are several lines which apply it as mentioned before, e.g., the SEPTA line in Philadelphia. Therefore, Assumption A.3 is possible in practice. With the recent development in information and communication technologies, it is possible to provide personalized transportation information to passengers via their mobile devices (like smart phone, PDAs, etc.). In addition, many stations in urban rail transit systems have screens to display travel information to passengers at platforms. Hence, Assumption A.4 is reasonable. In order to simplify the operation model for the trains, the detailed dynamics are not included in the model formulation, but only the three phases mentioned in Assumption A.5 are considered. However, once the running times between consecutive stations are fixed, a more accurate speed profile for the operation of trains can be calculated as a lower level control problem (see Chapter 3 for more information). Since we only consider a single line and since in Assumption A.2 we assume that no overtaking can happen at any point of the line, the transfer between different trains for passengers is useless. Therefore, it is reasonable to assume that passengers will wait at their origin for the right train to get to their destination as is stated in Assumption A.6. For Assumption A.7, the number of passengers is usually large, so the error made by this assumption is small. Furthermore, this assumption simplifies the optimization problem later on.

Remark 6.1 When realizing the stop-skipping strategy in practice, rail transit operators should be aware that some passengers may not like it since they want to keep their journey as simple as possible, e.g. when a train arrives at the platform they want to board the train. In addition, passenger could get on wrong trains which will not stop at their destinations, then they would feel frustrated. Furthermore, the passenger satisfaction to urban rail transit systems would go down because passengers waiting at the skipping stations will watch a

6.2.1 for more details). The values of these binary variables will be optimized to minimize the objective function of the train scheduling problem.

train passing away without stop. Therefore, a sophisticated passenger information system is essential for the stop-skipping strategy. \square

6.2.1 Arrivals and departures with stop-skipping

As has been indicated in Section 5.2.1, in order to obtain a balanced trade-off between the accuracy and the computation speed, a macroscopic model is used for the train scheduling. Section 5.2.1 has derived the arrival and departure equations for the operation of trains at stations. Here, the operation of trains in the terminal station is also included. We will first formulate the operation of trains at the terminal station and then at the stations.

Operation of trains in the terminal station

A train can depart from the terminal station only after it has arrived. Moreover, the train number is increased with I when it departs from the terminal station. So we have

$$d_{i,0} \geq a_{i-I,0} + \tau_{0,\min}, \quad (6.1)$$

where $d_{i,0}$ is the departure time of train i at the terminal station, a_{i-I} is the arrival time of train $i - I$, and $\tau_{0,\min}$ is the minimum dwell time for the trains at the terminal station. The minimum dwell time could be equal to the minimum turn-around time or the minimum running time at a terminal station. In addition, there is no upper bound for the dwell time of trains at the terminal station. Since we assume that there are multiple tracks in terminal station to accommodate trains, the running distance for trains between stations and the terminal station varies and depends on the route setting in the terminal station. However, the layout of the terminal station and the scheduling of trains in the terminal station are out of the scope of this chapter. Here, we assume an average distance s_0 for trains running between the terminal station and station 1 and an average distance s_J for the trains running between station J and the terminal station. The arrival time of train i at the terminal station can then be written as

$$a_{i,0} = d_{i,J} + r_{i,J}, \quad (6.2)$$

where $r_{i,J}$ is the running time on segment J .

The headway constraints in the terminal station can be formulated as

$$d_{i,0} \geq d_{i-1,0} + h_{0,\text{dep}}, \quad (6.3)$$

where $h_{0,\text{dep}}$ is the minimum departure headway at the terminal station. In addition, the minimum arrival headway at the terminal station should also be taken into account, which can be formulated as

$$a_{i,0} \geq a_{i-1,0} + h_{0,\text{arr}}, \quad (6.4)$$

where $h_{0,\text{arr}}$ is the minimum arrival headway at the terminal station.

As mentioned in assumption A.2, the capacity of the terminal station is C_0^{ter} . Therefore, at any time t the number of trains in the terminal station should be less than the capacity, which can be formulated as

$$\sum_{i \in S_{\text{trains}}} \mathcal{I}(a_{i,0} \leq t) - \sum_{i \in S_{\text{trains}}} \mathcal{I}(t \geq d_{i,0}) \leq C_0^{\text{ter}}, \quad \text{for all } t$$

where S_{trains} is the set of trains considered in the scheduling problem and the indicator function $\mathcal{I}(\cdot)$ is defined as

$$\mathcal{I}(x) = \begin{cases} 1 & \text{if } x \text{ is true,} \\ 0 & \text{if } x \text{ is false.} \end{cases}$$

The number of trains in the terminal station only increases when a train arrives at the terminal station. Therefore, we should only check the capacity constraint when a train arrives at a terminal station; so the constraints can be reformulated as

$$\sum_{i \in S_{\text{trains}}} \mathcal{I}(a_{i,0} \leq a_{\ell,0}) - \sum_{i \in S_{\text{trains}}} \mathcal{I}(a_{\ell,0} \geq d_{i,0}) \leq C_0^{\text{ter}}, \quad (6.5)$$

for each $\ell \in S_{\text{trains}}$.

Operation of trains at stations

By assumption A.3, trains can skip any station in the skipping set S_{skip} . If a train skips a station, then the train passes that station without stopping and the dwell time is then equal to zero. A binary variable has been introduced in Section 5.5 to indicate whether a train will stop at a station or not (see (5.28)):

$$y_{i,j} = \begin{cases} 1 & \text{if train } i \text{ will stop at station } j, \\ 0 & \text{if train } i \text{ will skip station } j. \end{cases}$$

In Section 5.2.1, the departure time $d_{i,j}$ of train i at station j is equal to the sum of the arrival time $a_{i,j}$ and the dwell time $\tau_{i,j}$ of train i at station j as shown in (5.1). Here, we reformulate (5.1) as the following two inequalities, i.e., the departure time $d_{i,j}$ should satisfy

$$d_{i,j} \geq a_{i,j} + y_{i,j} \tau_{i,j,\min} \quad (6.6)$$

and

$$d_{i,j} \leq a_{i,j} + y_{i,j} \tau_{i,j,\max}, \quad (6.7)$$

where $a_{i,j}$ is the arrival time of train i at station j , the minimum dwell time $\tau_{i,j,\min}$ is influenced by the number of passengers boarding and alighting from the train (see Section 5.2.3 for more details), and the maximum dwell time $\tau_{i,j,\max}$ is introduced to ensure passenger satisfaction. The arrival time $a_{i,j+1}$ of train i at station $j+1$ can be calculated by (5.2) in Section 5.2.1. The running time $r_{i,j}$ should be calculated by (5.32) instead of (5.3). Furthermore, the minimum headway is the same as (5.5). Moreover, if train i skips station $j+1$, i.e., $y_{i,j+1} = 0$, then the operation of the train between station j and station $j+2$ only contains three phases and the holding speed should satisfy (5.31) in Section 5.5.

6.2.2 OD-dependent passenger demand characteristics

Section 5.2.2 has described OD-independent passenger demand characteristics. Passenger demands with origin and destination are more complex than that in Section 5.2.2. The relationship between the variables used for describing the OD-dependent passenger characteristics is illustrated in Figure 6.2. The number of waiting passengers $w_{i,j}^{\text{wait}}$ for train i at

station j is equal to the sum of the number of waiting passengers $w_{i,j}^{\text{wait}}$ with destination m for all $m \in \{j+1, j+2, \dots, m, \dots, J\}$, i.e.,

$$w_{i,j}^{\text{wait}} = \sum_{m=j+1}^J w_{i,j,m}^{\text{wait}}. \quad (6.8)$$

The number of waiting passengers $w_{i,j}^{\text{wait}}$ with destination m can be calculated by

$$w_{i,j,m}^{\text{wait}} = w_{i-1,j,m} + \lambda_{j,m}(d_{i,j} - d_{i-1,j}), \quad (6.9)$$

where $w_{i-1,j,m}$ is the number of passengers with destination station m remaining at station j immediately after the departure of train $i-1$, $\lambda_{j,m}(d_{i,j} - d_{i-1,j})$ is the number of newly arrived passengers in between the departures of train i and train $i-1$, and $\lambda_{j,m}$ is the passenger arrival rate at station j for passengers with destination m . Note that the passenger arrival rate at the final station J , is assumed to be zero since we only consider one direction of the line.

The number of passengers alighting from train i at station j is denoted as $n_{i,j}^{\text{alight}}$ and it can be computed using

$$n_{i,j}^{\text{alight}} = \sum_{\ell=1}^{j-1} n_{i,\ell,j}^{\text{board}}, \quad (6.10)$$

where $n_{i,\ell,j}^{\text{board}}$ is the number of passengers that have station j as their destination and have boarded train i at station ℓ , i.e.,

$$n_{i,\ell,j}^{\text{board}} = w_{i,\ell,j}^{\text{wait}} - w_{i,\ell,j}. \quad (6.11)$$

No passenger will get off the train if train i skips station j because the passengers at upstream stations were informed that train i would not stop at station j ; so in that case those passengers with station j as destination would not get on train i . The number of passengers who want to board train i at station j and have station m as their destination is denoted as $w_{i,j,m}^{\text{want-to-board}}$. The number of passengers $w_{i,j,m}^{\text{want-to-board}}$ depends on whether train i stops at station j and whether train i stops at station m for $m \in \{j+1, j+2, \dots, J\}$, i.e.,

$$w_{i,j,m}^{\text{want-to-board}} = y_{i,j} y_{i,m} w_{i,j,m}^{\text{wait}}. \quad (6.12)$$

So if train i skips station j , i.e., $y_{i,j} = 0$, then no passengers want to board train i , i.e., $w_{i,j}^{\text{want-to-board}} = 0$. If train i stops at station j , i.e., $y_{i,j} = 1$, then the number of passengers who want to board is decided by whether train i stops at their destination m , i.e., $y_{i,m} w_{i,j,m}^{\text{wait}}$. Note that all the trains stop at the terminal station, so $y_{i,0}$ is equal to 1 for $i \in \{1, 2, \dots, I\}$.

The number of passengers $w_{i,j}^{\text{want-to-board}}$ who want to board train i at station j is

$$w_{i,j}^{\text{want-to-board}} = \sum_{m=j+1}^J w_{i,j,m}^{\text{want-to-board}}. \quad (6.13)$$

The number of passengers on train i immediately after its departure at station j is defined as $n_{i,j}$, which can be computed as

$$n_{i,j} = n_{i,j-1} - n_{i,j}^{\text{alight}} + n_{i,j}^{\text{board}}, \quad (6.14)$$

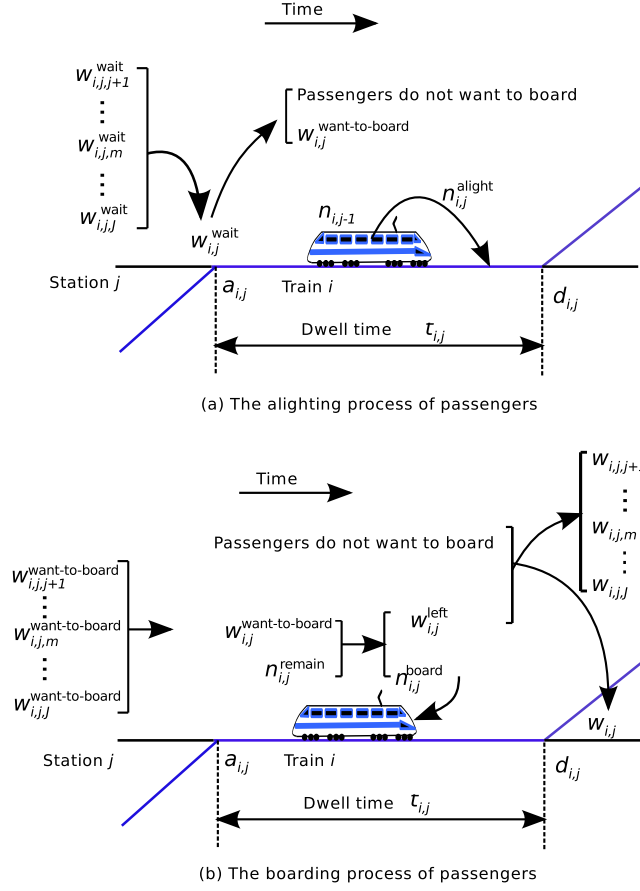


Figure 6.2: The variables for the passenger characteristics

where the number of boarding passengers $n_{i,j}^{\text{board}}$ equals the minimum of the number of passengers that want to board train i and the remaining capacity of the train:

$$n_{i,j}^{\text{board}} = \min(n_{i,j}^{\text{remain}}, w_{i,j}^{\text{want-to-board}}). \quad (6.15)$$

In addition, the number of passengers $n_{i,j}^{\text{board}}$ boarding train i at station j is also equal to

$$n_{i,j}^{\text{board}} = \sum_{m=j+1}^J n_{i,j,m}^{\text{board}}.$$

Moreover, the remaining capacity of train i at station j immediately after the alighting process is

$$n_{i,j}^{\text{remain}} = C_{\max} - n_{i,j-1} + n_{i,j}^{\text{alight}}. \quad (6.16)$$

The number of passengers $w_{i,j}^{\text{left}}$ left by train i depends on whether train i will stop at station j or not. We have the following two cases:

- Train i skips station j , i.e., $y_{i,j} = 0$

If train i will skip station j , then the number of boarding passengers $n_{i,j}^{\text{board}}$ is equal to zero. All the passengers waiting at station j will then be left by train i .

- Train i will stop at station j , i.e., $y_{i,j} = 1$

If $w_{i,j}^{\text{want-to-board}} \leq n_{i,j}^{\text{remain}}$, then all the passengers that want to board can get on train i . However, there will be passengers left by train i if $w_{i,j}^{\text{want-to-board}} > n_{i,j}^{\text{remain}}$. The number of passengers who want to board but cannot get on train i at station j immediately after the departure of train i is

$$w_{i,j}^{\text{left}} = w_{i,j}^{\text{want-to-board}} - \min(n_{i,j}^{\text{remain}}, w_{i,j}^{\text{want-to-board}}) \quad \text{if } y_{i,j} = 1.$$

In this case, if train i stops station m for $m \in \{j+1, j+2, \dots, J-1\}$, i.e., $y_{i,m} = 1$, we assume that the number of passengers that have station m as destination and are left by train i is proportional to the number of passengers who want to board. The number of passengers who have destination m and are left by train i can be formulated as

$$w_{i,j,m} = w_{i,j}^{\text{left}} \frac{w_{i,j,m}^{\text{want-to-board}}}{w_{i,j}^{\text{want-to-board}}} \quad \text{if } y_{i,j} = 1 \text{ and } y_{i,m} = 1.$$

However, if train i skips station m for $m \in \{j+1, j+2, \dots, J-1\}$, i.e., $y_{i,m} = 0$, then the number of passengers that have station m as destination will not board. So we have

$$w_{i,j,m} = w_{i,j,m}^{\text{wait}}, \quad \text{if } y_{i,j} = 1 \text{ and } y_{i,m} = 0.$$

Hence, the number of passengers who are left by train i and with destination m can be calculated as

$$w_{i,j,m} = y_{i,j} \left(y_{i,m} w_{i,j}^{\text{left}} \frac{w_{i,j,m}^{\text{want-to-board}}}{w_{i,j}^{\text{want-to-board}}} + (1 - y_{i,m}) w_{i,j,m}^{\text{wait}} \right) + (1 - y_{i,j}) w_{i,j,m}^{\text{wait}}. \quad (6.17)$$

Furthermore, the total number of waiting passengers at station j immediately after the departure of train i is

$$w_{i,j} = \sum_{m=j+1}^J w_{i,j,m}. \quad (6.18)$$

Remark 6.2 The minimum dwell time for the passengers to get on and get off trains can be calculated using the linear model (5.11) and the nonlinear model (5.12) presented in Section 5.2.3. See there for more information. \square

6.3 Mathematical formulation of the scheduling problem

The train scheduling problem with stop-skipping is a multi-objective optimization problem. In this chapter, the objective is similar as (5.21) defined in Section 5.3, where the energy consumption of trains and the travel time of passengers are considered.

For the train scheduling problem with stop-skipping, the energy consumption of trains can be calculated by (5.17), (5.18), and (5.35) in Section 5.3. The total travel time can be calculated by (5.16), which is repeated as follows:

$$t_{\text{total}} = \sum_{i=1}^I \sum_{j=1}^{J-1} (\gamma_{\text{wait}} t_{\text{wait},i,j} + t_{\text{in-vehicle},i,j}).$$

However, since we consider OD-dependent passenger demands in this chapter, instead of (5.14) and (5.15), the computation for the passenger waiting time and passenger in-vehicle time will be calculated as follows:

$$t_{\text{wait},i,j} = w_{i-1,j}(d_{i,j} - d_{i-1,j}) + \frac{1}{2} \sum_{m=j+1}^J \lambda_{j,m}(d_{i,j} - d_{i-1,j})^2, \quad (6.19)$$

and

$$t_{\text{in-vehicle},i,j} = n_{i,j} r_{i,j} + (n_{i,j} - n_{i,j}^{\text{alight}}) \tau_{i,j+1}. \quad (6.20)$$

In order to spread trains over the entire scheduling time period, we add a penalty term for the waiting time of the passengers left by the last train $N_{\text{cyc}}I$ during the scheduling period:

$$f_{\text{penalty},1} = \sum_{j=1}^J \left(w_{N_{\text{cyc}}I,j}(t_{\text{end}} - d_{N_{\text{cyc}}I,j}) + \frac{1}{2} \zeta \sum_{m=j+1}^J \lambda_{j,m}(t_{\text{end}} - d_{N_{\text{cyc}}I,j})^2 \right), \quad (6.21)$$

If $\zeta = 1$, the waiting time for the newly arrived passengers between the departure time $d_{N_{\text{cyc}}I,j}$ and the end time t_{end} is also considered. However, if $\zeta = 0$, the waiting time of the newly arrived passengers after the last train is not considered (e.g., the trains coming later will pick up these passengers). But in the latter case, we need to add a penalty term for the arrival time of the last train $N_{\text{cyc}}I$ at the terminal station to avoid all the trains operating close to each other at the start of the period $[t_0, t_{\text{end}}]$:

$$f_{\text{penalty},2} = |a_{N_{\text{cyc}}I,0} - t_{\text{end}}|. \quad (6.22)$$

The objective function of the train scheduling problem can be written as

$$f_{\text{opt}} = \gamma_1 \frac{E_{\text{total}}}{E_{\text{total,nom}}} + \gamma_2 \frac{t_{\text{total}}}{t_{\text{total,nom}}} + \mu_{\text{penalty},1} \frac{f_{\text{penalty},1}}{f_{\text{penalty},1,\text{nom}}} + \mu_{\text{penalty},2} \frac{f_{\text{penalty},2}}{f_{\text{penalty},2,\text{nom}}}, \quad (6.23)$$

where γ_1 , γ_2 , $\mu_{\text{penalty},1}$, and $\mu_{\text{penalty},2}$ are non-negative weights, and the normalization factors $E_{\text{total,nom}}$, $t_{\text{total,nom}}$, $f_{\text{penalty},1,\text{nom}}$, and $f_{\text{penalty},2,\text{nom}}$ are “nominal” values of the total energy consumption and the total travel time of passengers, respectively. These nominal values can e.g. be determined by running trains using a feasible initial schedule.

Remark 6.3 A rolling horizon approach can be adopted to solve the train scheduling problem. See Section 7.4 for more details on this. \square

6.4 Solution approaches

The resulting train scheduling problem is a mixed integer nonlinear programming (MINLP) problem with objective function (6.23) and constraints (6.1)-(6.18). The objective function (6.23) is nonlinear and nonconvex. In addition, we have the non-smooth min function

in constraints (6.14), (6.15), and (6.17). This MINLP problem can be solved using e.g. the branch-and-bound method, but the computation time is quite long even for small-sized problems in practice. We could also approximate the nonlinear terms in the MINLP problem using piecewise affine (PWA) functions and transform the MINLP problem into a mixed integer linear programming problem. However, we need to approximate many nonlinear terms of the scheduling problem; so the performance of the MILP approach is not optimal due to the big approximation error.

In this section, we propose a bi-level optimization approach to solve the optimization problem under consideration. However, the computation time of this bi-level optimization method is too long in practice. Therefore, we also propose an efficient bi-level solution approach for the MINLP problem, where the search space of the problem is limited and a threshold method is presented to obtain a good initial solution for the MINLP problem.

6.4.1 Bi-level optimization approach

The free variables in the real-time scheduling problem are the departure times $d_{i,j}$, the holding speeds $v_{i,j}$, and the binary variables $y_{i,j}$ for all trains and stations. The other variables like the number of passengers waiting at stations $w_{i,j}$ and the number of passengers on-board the trains $n_{i,j}$ can be eliminated using the model equations (6.8)-(6.18). The proposed bi-level optimization method consists of two levels of optimization:

- The high-level optimization optimizes the binary variables $y_{i,j}$ (only if $(i,j) \in S_{\text{skip}}$ $y_{i,j}$ could be equal to 0), where a brute force approach can be used to explore all the combinations for the binary variables in case the size of the problem is small. Alternatively, integer programming approaches, such as genetic algorithms, can be applied in the high-level optimization.
- For each combination of binary variables, the low-level optimization solves a nonlinear nonconvex problem using e.g., multi-start sequential quadratic programming (SQP) algorithm [15] or a pattern search method [61].

The procedure of the bi-level optimization method is given in Algorithm 2, where - for illustration purpose - the high level optimization problem is solved using a genetic algorithm and a multi-start optimization algorithm is used in the lower level. The feasibility of the low-level optimization problem depends on the value of the binary variables. If the low-level optimization problem is infeasible, we introduce a large penalty value F for the fitness function as shown in Algorithm 2. Furthermore, in order to steer infeasible binary variables towards feasible ones, we also add the norm of $\delta - \delta_f$ in the fitness function where δ_f is a feasible value of δ and can be decided using the known information about the urban rail transit line, e.g., passenger flows and line conditions.

6.4.2 Efficient bi-level optimization approach

In the efficient bi-level optimization problem, we propose a threshold method to obtain good initial solutions for the MINLP problem. In addition, we limit the search space of train scheduling problem in the neighborhood of these initial solutions.

Algorithm 2 Bi-level optimization approach for the train scheduling problem with stop-skipping

- 1: **Input** : maximum number of generations G , population size s_p , initial population P_0 of the binary variables, number of initial points k_{\max} used in the low-level optimization, a large value F for the fitness function to indicate infeasibility, feasible binary solution δ_f , positive weight λ_f ;
 - 2: **for** $g = 0, 1, \dots, G - 1$ **do**
 - 3: **for** $\ell = 1, 2, \dots, s_p$ **do**
 - 4: binary variables $\delta \leftarrow \ell$ -th parent $P_g(\ell)$ from the g -th generation;
 - 5: **for** $k = 1, 2, \dots, k_{\max}$ **do**
 - 6: generate an initial random feasible solution $d_{i,j}(\ell, 0)$, $r_{i,j}(\ell, 0)$, $\tau_{i,j}(\ell, 0)$, $w_{i,j}(\ell, 0)$, and $n_{i,j}(\ell, 0)$ for $i = 1, \dots, I$ and $j = 1, \dots, J$;
 - 7: **if** low-level optimization problem turns out to be feasible based on current values of δ and initial solutions **then**
 - 8: $d_{i,j}(\ell, k)$, $r_{i,j}(\ell, k)$, $\tau_{i,j}(\ell, k)$, $w_{i,j}(\ell, k)$, $n_{i,j}(\ell, k)$, and $f_{\text{opt}}(\ell, k) \leftarrow$ solution of the low-level optimization problem;
 - 9: **else**
 - 10: $f_{\text{opt}}(\ell, k) \leftarrow F + \lambda_f \|\delta - \delta_f\|_2$;
 - 11: **end if**
 - 12: **end for**
 - 13: value of fitness function $f_{\text{opt}}^*(\ell)$ for the ℓ -th parent $\leftarrow \min_{k=1, \dots, k_{\max}} f_{\text{opt}}(\ell, k)$;
 - 14: **end for**
 - 15: select new parents from the current population based on the fitness function f_{opt} ;
 - 16: generate a new generation population of binary variables through crossover and mutation;
 - 17: **end for**
 - 18: **Output** : choose the best offspring solution at generation G and calculate $d_{i,j}$, $r_{i,j}$, $\tau_{i,j}$, $w_{i,j}$, and $n_{i,j}$.
-

Threshold method for obtaining good initial solutions

In order to obtain a good initial solution for the train scheduling problem, we first introduce a threshold function to determine the value of the stopping variable as follows:

$$y_{i,j} = \mathcal{I}\left((w_{i,j}^{\text{want-to-board}} \geq \theta_{i,j}^{\text{th,in}}) \wedge (n_{i,j}^{\text{alight}} \geq \theta_{i,j}^{\text{th,out}})\right), \quad (6.24)$$

where $\mathcal{I}(\cdot)$ is the indicator function (see Section 6.2.1 for more information), $\theta_{i,j}^{\text{th,in}}$ and $\theta_{i,j}^{\text{th,out}}$ are the thresholds, which are free variables and determined by the optimization procedure. In this case, the value of the stopping variables depends on the passenger flows in the urban rail transit line. By introducing the threshold function, we can reformulate the MINLP problem as a real-valued nonlinear programming problem, which can be solved by e.g. a sequential quadratic programming method. The resulting solution then yields an initial solution of the stopping variables, the departure times, and the holding speeds for the original, full MINLP problem.

Limiting the search space

After a initial solution is obtained using the threshold method, we can limit the search space of the train scheduling problem within a neighborhood of the initial solution to reduce the computation time. For the high-level optimization, the search space of the integer variables $y = [y_{1,1}, y_{1,2}, \dots, y_{2,1}, \dots, y_{N_{\text{cycl}}, J}]^T$ can be limited by the 1-norm constraint:

$$\|y - y_{\text{init}}\|_1 \leq \chi_0, \quad (6.25)$$

which means that only a limited number, i.e., χ_0 , of binary variables can change their values in the bi-level optimization approach. If χ_0 is small, the search space can be reduced dramatically. A brute force method could be applied for the high-level optimization if χ_0 is chosen as 1 or 2 if the number of trains and stations is not too large. Otherwise, a genetic algorithm could be applied.

In addition, we can also limit the search space of the departure times and holding speeds as follows [45, 73]:

$$\|d - d_{\text{init}}\|_2 \leq \chi_2 + \chi_3\beta, \quad (6.26)$$

and

$$\|v - v_{\text{init}}\|_2 \leq \chi_3 + \chi_4\beta, \quad (6.27)$$

and the objective function (6.23) is revised as

$$f'_{\text{opt}} = f_{\text{opt}} + \gamma_4\beta, \quad (6.28)$$

where β is a slack variable which introduces an element of slackness into the problem to make sure the resulting optimization problem is always feasible. The relative degree of under- or over-achievement of the goals is controlled by the weights χ_3 , χ_4 and γ .

6.5 Case study

6.5.1 Set-up

In order to demonstrate the effectiveness of the proposed model formulation and the performance of the proposed efficient bi-level optimization approach, we consider a cyclic line with 1 terminal station and 12 stations following the structure shown in Figure 6.1. There are 6 physical trains in the cyclic line and the number of train services considered in the train scheduling problem is 10. The train characteristics and the line data are inspired by the data of Beijing Yizhuang subway line, and are given in Tables 6.1 and 6.2. In Table 6.2, station 0 represents the terminal station. The minimum running time in Table 6.2 is calculated by taking a fixed acceleration of 0.8 m/s^2 and a fixed deceleration of -0.8 m/s^2 ; furthermore, when calculating the minimum running time the trains are assumed to run at the maximum speed of 22.2 m/s during the holding phase. The maximum running time is assumed to be $r_{i,j,\text{max}} = \zeta r_{i,j,\text{min}}$, where ζ is larger than 1. We have chosen ζ as 1.2 to ensure that the passengers do not complain that the train is too slow. The mass of the train, the mass of one passenger, and the coefficients for the minimum dwell time in (5.12) are given in Table 6.1. Here, we use the nonlinear model (5.12) for the calculation of the minimum dwell time. In

Table 6.1: parameters of the trains and the passengers

Property		Symbol	Value
Train mass	[kg]	$m_{e,i}$	$199 \cdot 10^3$
Mass of one passenger	[kg]	m_p	60
Capacity of trains	[passengers]	$C_{i,max}$	1468
Minimum dwell time	[s]	$\tilde{\tau}_{min}$	30
Maximum dwell time	[s]	τ_{max}	150
Coefficients of the minimal dwell time	[s]	$\alpha_{1,d}$	4.002
	[s/passengers]	$\alpha_{2,d}$	0.047
	[s/passengers]	$\alpha_{3,d}$	0.051
	[s/passengers ⁻⁴]	$\alpha_{4,d}$	$1.0 \cdot 10^{-6}$
Coefficients of resistance	[m/s ²]	k_{1i}	0.012
	[s ⁻¹]	k_{2i}	$5.049 \cdot 10^{-4}$
	[m ⁻¹]	k_{3i}	$2.053 \cdot 10^{-5}$

addition, the minimum dwell time $\tilde{\tau}_{min}$ predefined by the rail operator in (5.13) is chosen as 30 s. The OD-dependent passenger arrival rates at stations are shown in Table 6.3.

The initial states at time t_0 (chosen as 1300 s for this case study) of the trains are as follows: train 1 and 2 are running to station 8 and 5, respectively. Since we assume that the schedule of a train can only be changed at stations, the arrival times of these two trains at station 3 and station 2 are fixed and they are 1400 s and 1340 s, respectively. Train 3 is stopped at station 3 and its arrival time is 1270 s. The numbers of passengers on train 1, 2, and 3 at time t_0 and their destination are given in Table 6.4. In addition, there are 3 trains at the terminal station, so the corresponding previous train services finished before t_0 . A communication-based train control system (a moving block signaling system) is implemented in Beijing Yizhuang subway line, where the minimum headway between two successive trains is 90 s. In addition, a maximum departure-departure headway is included to ensure the passenger satisfaction, which is chosen as 400 s. Furthermore, the numbers of passengers waiting at the various stations at t_0 and the destination of these passengers are shown in Table 6.5. The nominal values for the total travel time, the energy consumption, and the waiting time for the passengers who did not travel in the scheduling period are calculated based on a schedule with constant headway; they are $2.278 \cdot 10^7$ s, $7.013 \cdot 10^9$ J, and $1.387 \cdot 10^7$ s, respectively.

6.5.2 Results and discussion

The train scheduling problem is solved using the following three approaches:

- All-stop approach: Trains in the scheduling period stop at every station, i.e., there is no stop-skipping at all. In this case, the train scheduling problem is a nonlinear programming problem, which is solved here using the sequential quadratic programming (SQP) method implemented by the fmincon function of Matlab optimization toolbox.
- Bi-level approach with stop-skipping: The train scheduling problem with stop-skipping is a mixed integer nonlinear programming problem, which is solved using the bi-level

Table 6.2: Information of the cyclic line of the case study of Section 6.5

Station number	0	1	2	3	4	5	6	7	8	9	10	11	12
Distance to next station [m]	1050	1832	1786	2086	2265	1030	1354	1280	1544	992	1975	2369	1349
Minimal running time [s]	75.0	110.2	108.2	121.7	129.7	74.1	88.7	85.4	97.3	72.4	116.7	134.4	88.5

Table 6.3: Origin-destination-dependent passenger arrival rates at stations [passengers/s]

[illegible]

Table 6.4: Number of passengers on train 1, 2, and 3 at time t_0 and their destination

Destination station	1	2	3	4	5	6	7	8	9	10	11	12	Total number of passengers
Train 1	0	0	0	0	0	0	0	0	131	395	263	132	921
Train 2	0	0	0	0	0	89	33	111	44	333	166	22	798
Train 3	0	0	0	106	100	20	144	216	31	103	144	21	885

Table 6.5: Number of passengers waiting at stations at t_0 and their destination

Destination	1	2	3	4	5	6	7	8	9	10	11	12	Total number of passengers
Station 1	0	8	75	54	65	15	26	32	15	68	33	21	421
Station 2	0	0	19	19	19	14	16	13	15	13	11	14	153
Station 3	0	0	0	29	91	15	45	32	12	41	33	14	312
Station 4	0	0	0	0	57	22	52	43	22	11	24	35	266
Station 5	0	0	0	0	0	13	22	29	26	14	26	24	154
Station 6	0	0	0	0	0	0	25	23	26	5	30	14	123
Station 7	0	0	0	0	0	0	0	23	25	13	27	29	117
Station 8	0	0	0	0	0	0	0	0	25	9	19	18	71
Station 9	0	0	0	0	0	0	0	0	0	24	20	23	67
Station 10	0	0	0	0	0	0	0	0	0	0	21	25	46
Station 11	0	0	0	0	0	0	0	0	0	0	0	27	27
Station 12	0	0	0	0	0	0	0	0	0	0	0	0	0

approach. A genetic algorithm is applied for the integer optimization of the high level, where the `ga` function of the global optimization toolbox of Matlab is employed. The nonlinear optimization problem in the lower level is solved using the SQP algorithm of the `fmincon` function of the Matlab optimization toolbox.

- Efficient bi-level approach with stop-skipping: First, a good initial solution for the mixed integer nonlinear problem is obtained by solving the train scheduling problem with threshold function (6.24), which is a nonlinear programming problem and which is solved using the SQP algorithm of the `fmincon` function of the Matlab optimization toolbox. Based on an initial train schedule, we assign the value of χ_0 in (6.25) as 1, 2, and 3 to vary the search space. For $\chi_0 = 1$, we apply a brute force approach for the high-level optimization since the search space of the binary variables is small then. For $\chi_0 = 2$ or 3, the genetic algorithm function `ga` of the global optimization toolbox of Matlab is used to optimize the binary variables.

Results

The train schedules obtained by the all-stop approach, the bi-level approach, and the efficient bi-level approach are shown in Figures 6.3-6.8. These train schedules look similar to each other; however, there are some differences between them. In particular, for the all-stop approach (Figure 6.3) all trains stop at all stations, while several trains skip some stations in the train schedules obtained by the bi-level approach and the efficient bi-level approach (see Figures 6.4-6.8). In the train schedule obtained by the bi-level approach shown in Figure 6.4, trains 4, 6, 7, 8, 9, and 10 skip some stations. More specifically, train 4 skips stations 2, 5, 8, and 11; so the stopping variables of train 4 for these stations are equal to 0 as shown in Table 6.6. In addition, we can observe that the travel time for trains that skip some stations is smaller than that of the all-stop approach, e.g., train 4 arrives earlier at the terminal station in Figure 6.4 (stop-skipping approach) than in Figure 6.3 (all-stop approach). Trains 1, 2, and 3 have already departed from the terminal station at time t_0 and they are thus suppose to stop at all stations. So the stopping variables for these three trains are equal to 1. Figure 6.5 illustrates an initial train schedule that is obtained by the threshold method. The values of the stopping variables are shown in Table 6.6, which are different from those obtained via the bi-level approach. For example, train 4 only skips stations 5 and 8 but does not skip station 2 and 11. Figure 6.6 shows the train schedule obtained by the efficient bi-level approach with $\chi_0 = 1$, which means that one binary variable can change its value. From Figure 6.6 and Table 6.6, we can observe that train 7 skips station 5 when $\chi_0 = 1$, which is different from the initial schedule. Similarly, when $\chi_0 = 2$, the values of two binary variables are changed compared to the initial solution. As we can observe from Figure 6.7 and Table 6.6, train 4 skips station 11 and train 7 skips station 5, while in the initial train schedule shown in Figure 6.5, train 4 stops at station 11 and train 7 stops at station 5. Figure 6.8 presents the train schedule obtained by the efficient bi-level approach with $\chi_0 = 3$, where train 4 skips station 11 and both train 7 and train 9 skips station 5 compared with the initial schedule.

Table 6.7 lists the values of the objective function, the computation time (on a 64-bit Linux operation system running on a 1.8 GHz Intel Core2 Duo CPU), the total passenger travel time, the energy consumption of the train schedules, etc. for each of the three approaches. Note that the comparison here is fair since all the approaches are implemented in m-files in Matlab. The relative improvements of the bi-level approach and the efficient

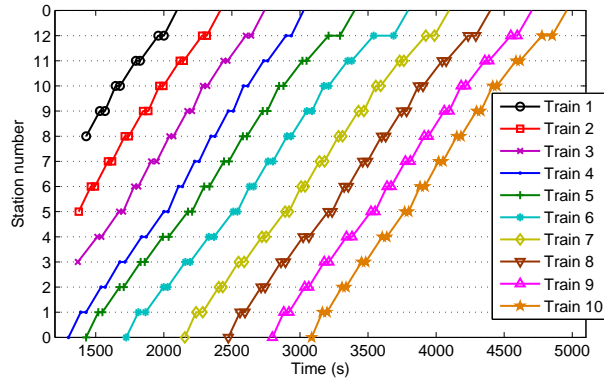


Figure 6.3: Train schedule obtained by the all-stop approach

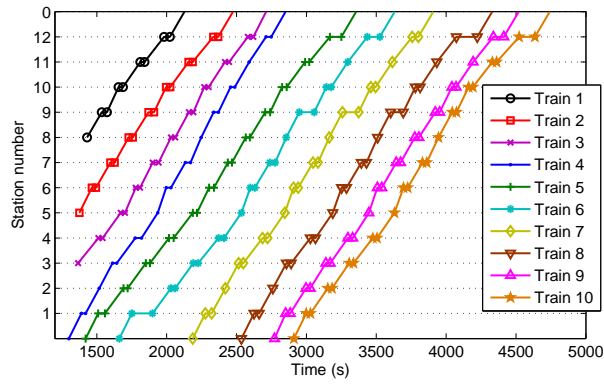


Figure 6.4: Train schedule obtained by the bi-level approach

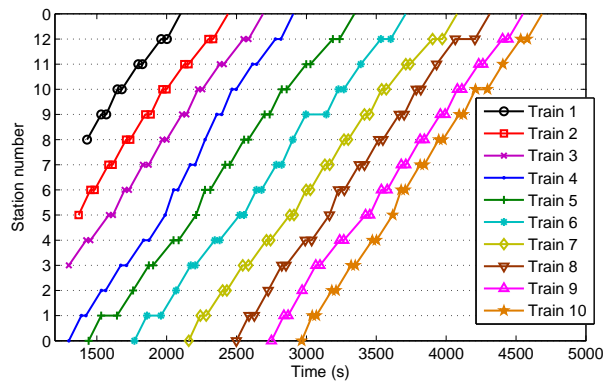


Figure 6.5: Train schedule obtained by the threshold method

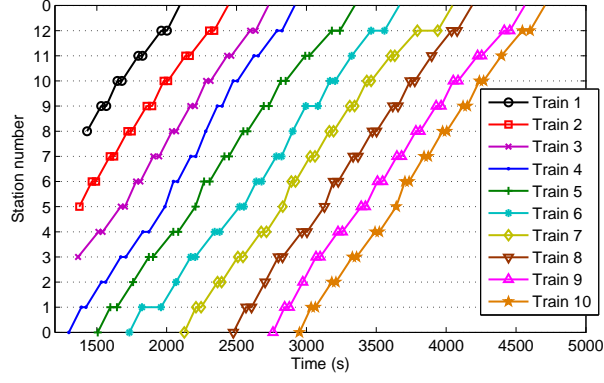


Figure 6.6: Train schedule obtained by the efficient bi-level approach with $\chi_0 = 1$

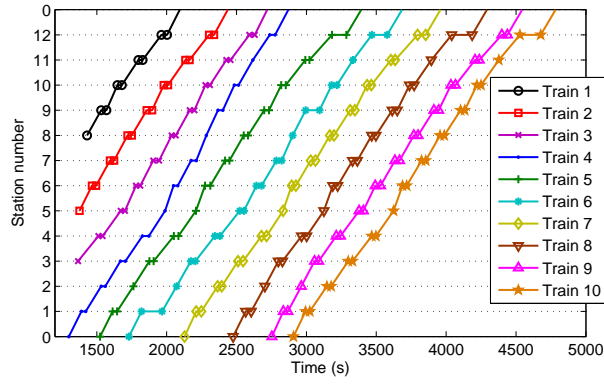


Figure 6.7: Train schedule obtained by the efficient bi-level approach with $\chi_0 = 2$

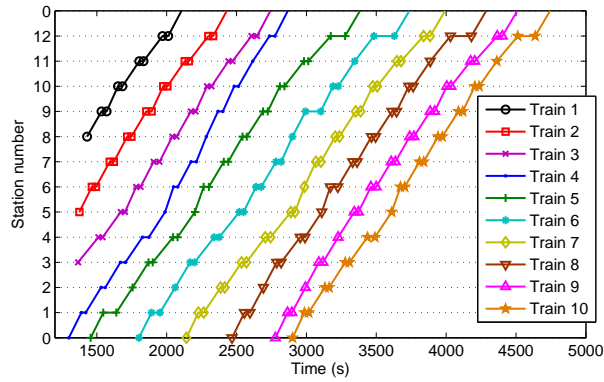


Figure 6.8: Train schedule obtained by the efficient bi-level approach with $\chi_0 = 3$

Table 6.6: Stopping variables of the all-stop approach, the bi-level approach, and the efficient bi-level approach at station 2, 5, 8, and 11 (The stopping variables for other stations are all equal to 1. Note that if the stopping variable equals 1, then the train stops at the station; otherwise, the train skips the station.)

Solution approaches	Solution options	Train	1	2	3	4	5	6	7	8	9	10
Bi-level	–	Station 2	1	1	1	0	1	1	0	0	1	1
		Station 5	1	1	1	0	1	0	0	0	0	0
		Station 8	1	1	1	0	1	0	0	0	1	0
		Station 11	1	1	1	0	1	0	0	0	0	1
Efficient bi-level	Solution (threshold)	Station 2	1	1	1	1	0	0	1	0	0	1
		Station 5	1	1	1	0	0	1	1	0	1	0
		Station 8	1	1	1	0	1	0	1	1	1	1
		Station 11	1	1	1	1	1	0	1	0	1	0
	Solution ($\chi_0 = 1$)	Station 2	1	1	1	1	0	0	1	0	0	1
		Station 5	1	1	1	0	0	1	0	0	1	0
		Station 8	1	1	1	0	1	0	1	1	1	1
		Station 11	1	1	1	1	1	0	1	0	1	0
	Solution ($\chi_0 = 2$)	Station 2	1	1	1	1	0	0	1	0	0	1
		Station 5	1	1	1	0	0	1	0	0	1	0
		Station 8	1	1	1	0	1	0	1	1	1	1
		Station 11	1	1	1	0	1	0	1	0	1	0
	Solution ($\chi_0 = 3$)	Station 2	1	1	1	1	0	0	1	0	0	1
		Station 5	1	1	1	0	0	1	0	0	0	0
		Station 8	1	1	1	0	1	0	1	1	1	1
		Station 11	1	1	1	0	1	0	1	0	1	0

Table 6.7: Performance comparison of the all-stop approach, the bi-level approach, and the efficient bi-level approach

Solution approach	All-stop	Bi-level	Efficient bi-level			
			Initial solution (threshold)	Solution ($\chi_0 = 1$)	Solution ($\chi_0 = 2$)	Solution ($\chi_0 = 3$)
Objective value [-]	2.917	2.617	2.708	2.681	2.661	2.653
Computation time [s]	$4.464 \cdot 10^2$	$1.672 \cdot 10^4$	$1.194 \cdot 10^3$	$1.705 \cdot 10^3$	$2.352 \cdot 10^3$	$4.567 \cdot 10^3$
Energy consumption [J]	$5.888 \cdot 10^9$	$4.994 \cdot 10^9$	$5.369 \cdot 10^9$	$5.301 \cdot 10^9$	$5.220 \cdot 10^9$	$5.111 \cdot 10^9$
Number of passengers that finished their trips [passengers]	$2.603 \cdot 10^4$	$2.451 \cdot 10^4$	$2.500 \cdot 10^4$	$2.505 \cdot 10^4$	$2.482 \cdot 10^4$	$2.473 \cdot 10^4$
Number of passengers that did not travel [passengers]	$1.324 \cdot 10^4$	$1.476 \cdot 10^4$	$1.427 \cdot 10^4$	$1.422 \cdot 10^4$	$1.445 \cdot 10^4$	$1.455 \cdot 10^4$
Travel time for passengers that finished their trips [s]	$1.944 \cdot 10^7$	$1.648 \cdot 10^7$	$1.724 \cdot 10^7$	$1.706 \cdot 10^7$	$1.680 \cdot 10^7$	$1.683 \cdot 10^7$
Waiting time of passengers that did not travel ³ [s]	$1.031 \cdot 10^7$	$1.270 \cdot 10^7$	$1.192 \cdot 10^7$	$1.188 \cdot 10^6$	$1.227 \cdot 10^6$	$1.240 \cdot 10^6$

³These remaining passengers, i.e., the passengers that did not travel, will be picked up by the trains that arrive later on. The waiting time of those passengers is also included in the objective function. So here the number and the waiting time of these passengers are also given for comparing the performance of these three approaches.

Table 6.8: Relative improvement with respect to the all-stop approach of the bi-level approach, and the efficient bi-level approach

Solution approach	Bi-level	Efficient bi-level			
		Initial solution (threshold)	Solution ($\chi_0 = 1$)	Solution ($\chi_0 = 2$)	Solution ($\chi_0 = 3$)
Objective value [-]	12.01%	7.16%	8.08%	8.77%	9.05%
Computation time [s]	-3646%	-168%	-282%	-427%	-923%
Energy consumption [J]	15.18%	8.81%	9.97%	11.35%	13.20%
Number of passengers that finished their trips [passengers]	5.83%	3.94%	3.77%	4.64%	5.01%
Number of passengers that did not travel [passengers]	-11.46%	-7.75%	-7.41%	-9.12%	-9.85%
Travel time for passengers that finished their trips [s]	15.19%	11.32%	12.24%	13.55%	13.41%
Waiting time of passengers that did not travel [s]	-23.26%	-15.64%	-15.29%	-19.04%	-20.35%

bi-level approach with respect to the all-stop approach are given as in Table 6.8, and they are calculated as

$$x_{\text{relative-difference}} = 1 - \frac{x_{\text{stop-skipping}}}{x_{\text{all-stop}}},$$

where x is the value of the objective function, the computation time, etc. in the table and the stop-skipping involves the value obtained for the bi-level approach or the efficient bi-level approach. Note that in Table 6.7, a solution approach has a better performance when the number of passengers finished their trips is larger since more passengers have finished their trip during the scheduling period. For the other criteria in Table 6.7, the performance of the approach is better if these criteria have a smaller value. In Table 6.8, if the value of relative improvement of the number of passengers that finished their trip is smaller, then the approach has a better performance. For the other criteria in Table 6.8, a bigger value represents a better performance.

Discussion

For the given case study, the overall performance improvement of the stop-skipping strategy is about 8-12% better compared to the all-stop approach. With the stop-skipping strategy, the total travel time is reduced with 12-15% and the total energy consumption is reduced with 10-15%. The number of passengers who did not travel increases with 7-11%; however, note that the trains coming later will pick up these passengers anyway; so those passengers will not be left at the platform. Since we solving the train scheduling problem in a rolling horizon way (cf. Section 7.4), the passengers who did not travel in the current time period will be taken into account in the next period. The efficient bi-level approach yields an acceptable performance when compared with the bi-level approach. However, the computation time of the bi-level approach is about 10 and 4 times longer than that of the efficient bi-level approach with $\chi_0 = 1$ and $\chi_0 = 3$, respectively. Note that the computation time of the approaches proposed in this chapter can be reduced by implementing the approaches in object code and by using faster processors and/or parallel processing.

6.6 Conclusions

We have considered the train scheduling problem with OD-dependent passenger demands for an urban rail transit line, where the operation of trains at both the stations and the terminal station are included in the model. Since the resulting train scheduling problem is a mixed integer nonlinear programming (MINLP) problem, an efficient bi-level approach has been proposed, where a threshold method is applied to obtain a good initial solution for the full problem and where the search space for the variables can next be limited to enhance the efficiency. For a case study, the efficient bi-level approach with a limited search space provides the best solution within the time that is typically available for the computations (e.g., half an hour). In particular, the overall performance improved with about 8-12% compared to the all-stop approach.

In the future, we will investigate other solution approaches to solve the resulting MINLP problem efficiently, especially for cases with a large number of trains and stations, and we will compare these approaches with the threshold approach and the efficient bi-level approach for large-scale real-life case studies. In addition, we will investigate the effect of

more detailed models (including short turns, the distribution of on-board passengers and waiting passengers at platforms, etc.) on the trade-off between performance and computational complexity. In future work, we will also consider the travel time uncertainty, dwell time uncertainty, etc. in the train scheduling and investigate robust train scheduling approaches.

Chapter 7

OD-Dependent Train Scheduling for an Urban Rail Transit Network

In the previous two chapters, we have discussed the train scheduling problem for an urban rail transit line with OD-independent and OD-dependent passenger demands. In the current chapter, we consider the train scheduling problem for an urban rail transit *network*. An event-driven model is built for train scheduling, which involves three types of events, i.e., departure events, arrival events, and passenger arrival rates change events. The routing of the arriving passengers at transfer stations is formulated in the train scheduling model. Moreover, the passenger transfer behavior (i.e., passengers' walking time and transfer duration time) is also taken into account in the model formulation. The resulting optimization problem is a real-valued nonlinear nonconvex problem. The effectiveness of the event-driven model is evaluated through a case study.

7.1 Introduction

In Chapters 5 and 6, we have studied the train scheduling problem for an urban rail transit line which takes the passenger demands into account. As in general urban rail transit lines are separated from each other, passengers may need to make several interchanges between different lines to arrive at their destination. Therefore, when scheduling trains for an urban rail network, it is important to take the passenger transfers into account to shorten the total travel time of passengers.

In order to model the time-varying passenger arrival rates, we propose an event-driven model for the train scheduling in this chapter. The event-driven model includes departure events, arrival events, and passenger arrival rate change events at platforms. At transfer stations, if there exist multiple route choices for passengers to arrive at their destinations, the arriving passengers will distribute themselves and go to different platforms of different lines. For the passengers arriving at transfer stations by trains, some of them will get off the train and transfer to other lines to arrive at their destination. Both the changes of splitting

rates and the passenger transfers result in passenger arrival rate change events at transfer stations. This behavior can be captured by the event-driven model. The resulting train scheduling problem is a real-valued nonlinear nonconvex programming problem.

The rest of the chapter is structured as follows. Section 7.2 introduces the three types of events and formulates the event-driven dynamics of train scheduling. Section 7.3 describes the performance criteria and constraints of the train scheduling problem. In Section 7.4, we discuss how to solve the scheduling problem in a rolling horizon way and how to define the initial conditions for the scheduling problem. Several solution approaches, e.g., the SQP method and genetic algorithm, are introduced to solve the resulting nonlinear nonconvex optimization problem in Section 7.5. In Section 7.6, the performance of the proposed event-driven model is evaluated via a case study. Finally, conclusions and recommendations are provided in Section 7.7.

7.2 Model formulation

Consider an urban rail transit network with L lines and J stations. Let S_{ln} and S_{sta} be the sets of lines and station indices, respectively. In practice, a station could have several platforms and we denote the set of platforms as S_{pla} . Note that a physical line with two directions is defined as two separate lines in this chapter. We make the following assumptions:

- A.1 There is no shared platform for different lines in the urban rail transit network. If passengers want to transfer from one line to the other, they need to walk from one platform to the other.
- A.2 A platform can only accommodate one train at a time and no overtaking can occur at any point of the line.

Assumption A.1 holds for most urban transit systems, e.g., the subway networks in Beijing, Paris, and Rome. With Assumption A.1, a platform is uniquely identified to a specific line, i.e., a line can be defined by a subset of platforms. Assumption A.2 generally holds for most urban transit systems. Furthermore, in practice, the trains of different lines are operated separately, which means that trains cannot be shared between different lines.

If platform p at station j is on line ℓ , we denote the predecessor of platform p on line ℓ as $\text{p}^{\text{pla}}(p)$ and the successor of platform p on line ℓ as $\text{s}^{\text{pla}}(p)$. In order to distinguish the different running cycles of the physical trains, train services are introduced, where each train service in the network has a unique service number which uniquely identifies a train and its current cycle. After the arrival of a physical train at the terminal station, its service number will be augmented the total number of trains in the network when the train departs. Let I_ℓ be the total number of physical trains in transit line ℓ , then the total number of physical trains in the network is $I_{\text{net}} = \sum_{\ell \in S_{\text{ln}}} I_\ell$. In addition, the set of indices of all train services is defined as S_{tra} . The predecessor and successor of train i on line ℓ are denoted as $\text{p}^{\text{tra}}(i)$ and $\text{s}^{\text{tra}}(i)$, respectively. The start time and end time of the scheduling period are denoted as t_0 and t_{end} .

7.2.1 Three types of events

We model the train scheduling problem with consideration of passenger demands using three types of events:

- Departure events: representing the departure of a train at a station,
- Arrival events: representing the arrival of a train at a station,
- λ -change events: representing the change of passenger arrival rates at a platform.

To describe the operation of trains, we now propose an event-driven model consisting of a continuous part describing the movement of trains running from one station to another station through the network, and of the discrete events listed above. The k -th event e_k occurring in the event-driven system is denoted as

$$e_k = (t_k, Y_{\text{type},k}, i_k, p_k), \quad (7.1)$$

where k is the event counter, t_k is the time instant at which event e_k occurs, $Y_{\text{type},k}$ is the event type, which can have three possible values, i.e., ‘d’, ‘a’, or ‘ λ ’ corresponding to a departure event, an arrival event, or a λ -change event, i_k is the train number, and p_k is the platform number.

In particular, λ -change events can be caused by the change of passenger arrivals at stations, the change of splitting rates at transfer stations, and the passenger transfers at transfer stations. Note that the passenger arrival rate stays the same between two subsequent events. The train scheduling model requires the real-time assessment of passenger arrival rates for different origins and destinations during the scheduling period. In the case of full state information, the passenger arrival rates can be obtained. Unfortunately, this is not the case in practice, where we need to e.g. use the information collected by the advanced fare collection systems and estimate the passenger arrival rates based on the historical data and the current passenger flows [143]. A typical profile for passenger arrivals at stations on work-days is given as the line in Figure 7.1, where the passenger arrival rate during the peak hours is much higher than that during the off-peak hours. The continuous passenger arrival rate can be approximated using a piecewise constant function as indicated by the dashed line in Figure 7.1. Piecewise constant functions $\lambda_{j,m}^{\text{station}}(\cdot)$ for each $j, m \in S_{\text{sta}}$ denote the passenger arrival rates at station j of passengers with station m as their final destination. These piecewise functions are the inputs to the event-driven model and we describe these piecewise constant functions via so-called base profiles. The base profiles are left-hand side continuous piecewise constant functions, which can be specified by a list of corner points as shown in Figure 7.2, where the corner points are marked with big dots. Hence, the base profile shown in Figure 7.2 can be described by these three corner points:

$$\{(t_1, \lambda_1), (t_2, \lambda_2), (t_3, \lambda_3)\}.$$

We introduce a λ -profile query module for each platform (see also Figure 7.3) [80]. If platform p is not at a transfer station, then the query module only contains the base profile $\lambda_{j,m}^{\text{station}}(\cdot)$. However, if platform p is at a transfer station, then the query module for a platform stores the base profile and possibly additional update profiles due to splitting rate changes and passenger transfers:

1. Splitting rates change at stations: At a transfer station, passengers can choose to go to the platforms of different lines since there could be multiple routes available to go to their destination. The splitting of passenger flows at transfer stations can be influenced or controlled by the rail operator by providing route information and

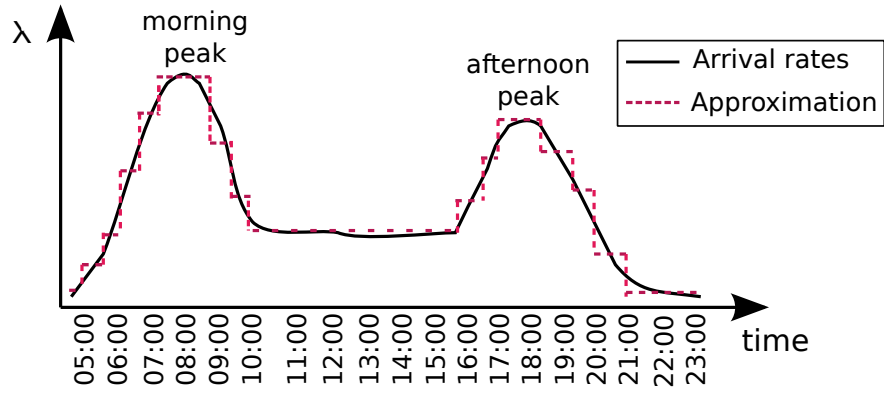


Figure 7.1: Typical passenger arrival rate profile at a station for a working day

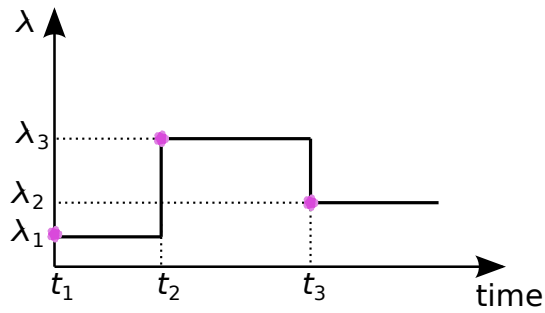


Figure 7.2: Example of base profile with corner points for passenger arrival rates

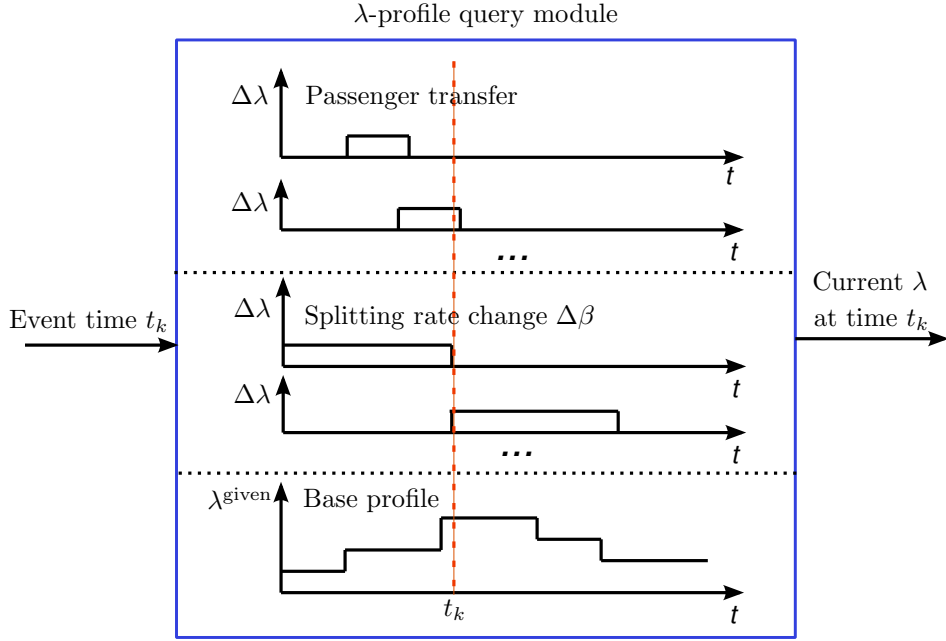
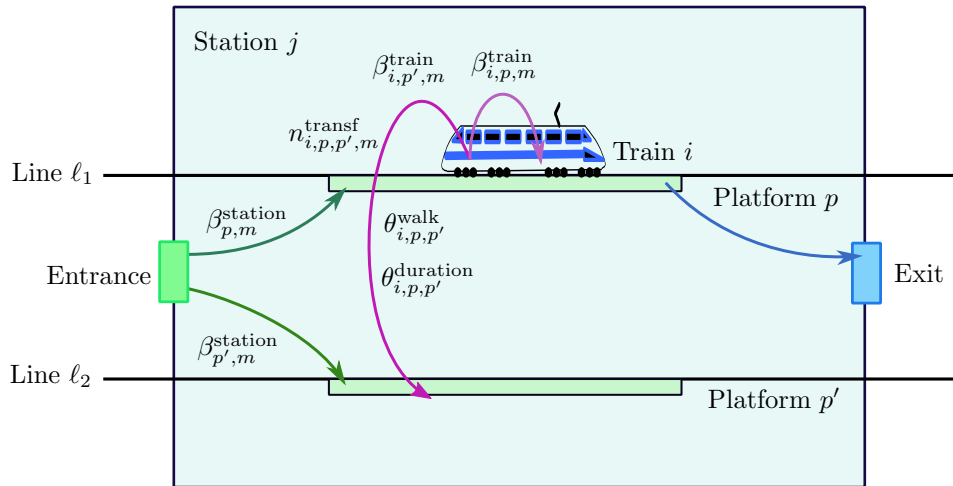
Figure 7.3: λ -profile query module

Figure 7.4: Variables for the splitting of passenger flows and passenger transfers

suggestions to passengers through PDA devices or through information panels at the entrance of stations. Let $\beta_{p,m}^{\text{station}}(\cdot)$ denote the splitting rate of the passengers flows that arrive at station j , have destination m , and go to platform p (see Figure 7.4). The function $\beta_{p,m}^{\text{station}}(\cdot)$ is also a left-hand side continuous piecewise constant function. In order to provide a consistent service to the passengers, the splitting rate should not change too often, e.g., 15 minutes. The passenger arrival rate at platform p of station j can be calculated as follows:

$$\lambda_{p,m}(t_k) = \beta_{p,m}^{\text{station}}(t_k) \lambda_{j,m}^{\text{station}}(t_k), \quad \forall p \in P_j, \forall m \in S_{\text{sta}}, \quad (7.2)$$

where t_k is one of the corner points of the base profile or of the splitting rate change profiles and P_j is the set of platforms at transfer station j . Furthermore, the sum of all the splitting rates at transfer station j is always equal to 1, i.e.,

$$\sum_{p \in P_j} \beta_{p,m}^{\text{station}}(t_k) = 1, \quad \forall m \in S_{\text{sta}}, \forall j \in S_{\text{sta}}^{\text{transf}}, \quad (7.3)$$

where $S_{\text{sta}}^{\text{transf}}$ is the set of transfer stations in the network. The splitting rates at the transfer stations are the control variables for the train scheduling problem. The change of the splitting rates of passenger flows results in λ -change events at the platforms of a transfer station. The $\Delta\lambda$ update profiles caused by splitting rate changes are also piecewise constant functions, which can be described by a list of corner points in a similar way as for base profiles.

The average walking time for passengers from entrances of station j to platform p at time instant t can be calculated by

$$\theta_p^{\text{walk-in}}(t) = a_{0,p}^{\text{walk}} \left(\sum_m \beta_{p,m}^{\text{station}}(t) \lambda_{j,m}^{\text{station}}(t) \right) + b_{0,p}^{\text{walk}}, \quad (7.4)$$

where $a_{0,p}^{\text{walk}}$ and $b_{0,p}^{\text{walk}}$ are the coefficients for the average walking time from the entrance of the station to platform p , which depend on the layout of the station, the walking distance, etc., and which can e.g. be determined based on historical data. The total walking time for passengers from entrances of station j to platform p during the scheduling time period $[t_0, t_{\text{end}}]$ can be calculated by

$$t_p^{\text{walk-in}} = \sum_{c=1}^C \theta_p^{\text{walk-in}}(t_{p,c}^{\text{station}}) \sum_m \beta_{p,m}^{\text{station}}(t_{p,c}^{\text{station}}) \lambda_{j,m}^{\text{station}}(t_{p,c}^{\text{station}}) (t_{p,c+1}^{\text{station}} - t_{p,c}^{\text{station}}), \quad (7.5)$$

where $t_{p,c}^{\text{station}}$ is the time instant at which PWA constant functions $\beta_{p,m}^{\text{station}}(\cdot)$ and/or $\lambda_{j,m}(\cdot)$ change. Note that $t_{p,1}^{\text{station}} = t_0$ and $t_{p,C}^{\text{station}} = t_{\text{end}}$.

2. Passenger transfers triggered by arrival events: If a train arrives at a transfer station, there could be several possible routes for the onboard passengers to arrive at their destinations. They could choose to stay on the train or to get off the train and transfer to a train on another line. At transfer station j , the splitting rate of the passengers that are on board of train i and have destination m to platform p' can be denoted as $\beta_{i,p',m}^{\text{train}}$ for $p' \in P_j$. For train i that stops at platform p of transfer station j , the sum of all the splitting rates has to be equal to 1, i.e.,

$$\sum_{p' \in P_j} \beta_{i,p',m}^{\text{train}} = 1, \quad \forall i \in S_{\text{tra}}, \forall m \in S_{\text{sta}}. \quad (7.6)$$

Note that the passengers with destination j , i.e., the ones for which $m = j$, will not choose to transfer to other platforms but they will exit the transit network at station j ; so if train i arrives at platform p , we set $\beta_{i,p,j}^{\text{train}} = 1$ for and $\beta_{i,p',j}^{\text{train}} = 0$ for $p' \in P_j \setminus \{p\}$ where p is the platform of station j at which train i arrives. The total walking time for passengers from platform p to exit station j during the scheduling time period $[t_0, t_{\text{end}}]$ can be calculated as

$$t_p^{\text{walk-out}} = \sum_{i \in S_{p,t_0,t_{\text{end}}}^{\text{tra}}} a_{p,0}^{\text{walk}} n_{i,p,j}^{\text{alight}} + b_{p,0}^{\text{walk}}, \quad (7.7)$$

where $S_{p,t_0,t_{\text{end}}}^{\text{tra}}$ is the subset of indices of trains that stop at platform p during the scheduling period $[t_0, t_{\text{end}}]$, $n_{i,p,j}^{\text{alight}}$ is the number of passengers who get off train i , have destination j , and exit the urban rail network from platform p . The coefficients $a_{p,0}^{\text{walk}}$ and $b_{p,0}^{\text{walk}}$ can be determined in a similar way as $a_{0,p}^{\text{walk}}$ and $b_{0,p}^{\text{walk}}$.

The walking time for transfer passengers depends on the walking distance between two platforms and the number of transfer passengers. In practice, the walking time could be distributed as shown by the line in Figure 7.5. For the sake of simplicity, we approximate the relationship between the passenger walking time and number of transfer passengers by a rectangular signal as represented by the dashed line in Figure 7.5. Hence, we can calculate the average walking time of the transfer passengers from platform p to the other platforms $p' \in P_j \setminus \{p\}$ as

$$\theta_{i,p,p'}^{\text{walk}} = a_{p,p'}^{\text{walk}} n_{i,p,p'}^{\text{transf}} + b_{p,p'}^{\text{walk}}, \quad \forall i \in S_{\text{tra}}, \forall p' \in P_j \setminus \{p\}, \quad (7.8)$$

where $n_{i,p,p'}^{\text{transf}}$ is the number of transfer passengers from train i to platform p' of line ℓ' , $a_{p,p'}^{\text{walk}}$ and $b_{p,p'}^{\text{walk}}$ are the coefficients for the average walking time, which depend on the layout of transfer station, the walking distance, etc., and which can e.g. be determined based on historical data. The total transfer time $t_{i,p}^{\text{transf}}$ for transferring passengers getting off from train i is

$$t_{i,p}^{\text{transf}} = \sum_{p' \in P_j \setminus \{p\}} \theta_{i,p,p'}^{\text{walk}} n_{i,p,p'}^{\text{transf}}. \quad (7.9)$$

Similar as the average walking time, the duration time for the transfer process can be approximated using

$$\theta_{i,p,p'}^{\text{duration}} = a_{p,p'}^{\text{duration}} n_{i,p,p'}^{\text{transf}} + b_{p,p'}^{\text{duration}}, \quad \forall i \in S_{\text{tra}}, \forall p' \in P_j \setminus \{p\}. \quad (7.10)$$

Similar as $a_{p,p'}^{\text{walk}}$ and $b_{p,p'}^{\text{walk}}$, $a_{p,p'}^{\text{duration}}$ and $b_{p,p'}^{\text{duration}}$ can be determined based on historical data. The updates for the λ -profile due to passenger transfers can be described by a list of corner points

$$\{(a_{i,j}, 0), (a_{i,j} + \theta_{i,p,p'}^{\text{walk}}, \Delta\lambda_{i,p,p'}), (a_{i,j} + \theta_{i,p,p'}^{\text{walk}} + \theta_{i,p,p'}^{\text{duration}}, 0)\}, \quad (7.11)$$

where $\Delta\lambda_{i,p,p'}$ is calculated by

$$\Delta\lambda_{i,p,p'} = \frac{n_{i,p,p'}^{\text{transf}}}{\theta_{i,p,p'}^{\text{duration}}}. \quad (7.12)$$

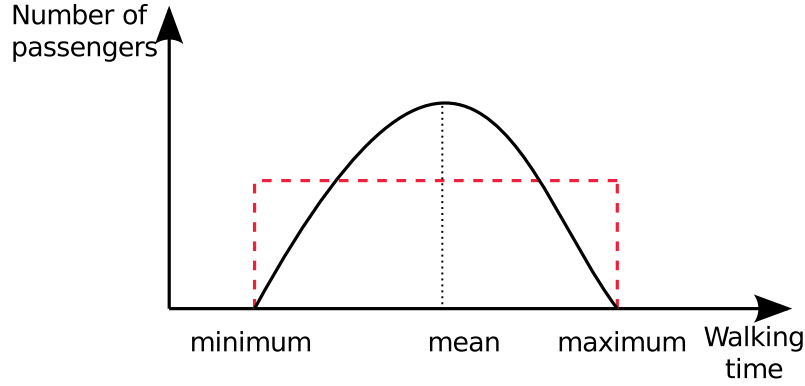


Figure 7.5: Typical walking time profile for the transfer passengers

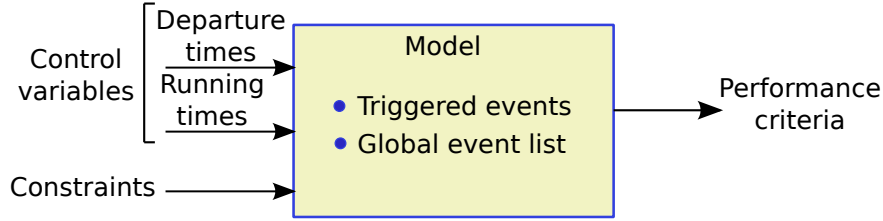


Figure 7.6: Model structure of the event-driven system

7.2.2 Event-driven dynamics

In this event-driven system, there are two classes of events: autonomous events and controlled events. Autonomous (or triggered) events are triggered by other events or by the environment, and their event times cannot be controlled directly. All the λ -change events are autonomous events. The departure and arrival events at stations are the controlled events of the given system. The event times of the controlled events are directly influenced by the inputs to the system. The control inputs of the system are the departure times, the running times of trains, and the splitting rates of passenger flows at transfer stations. The model structure of the event-driven system is illustrated in Figure 7.6.

Furthermore, we introduce a global event list for the event-driven system (see also Figure 7.6). At any time, this list contains all the possible next events for all the trains and stations in the urban rail network. The next event of the system will be the event in the global event list with the smallest value of t_k , i.e. the event that will occur first. As a starting point, the global event list should be initialized based on the initial state of the system. We denote the current time as t_{current} (see also Figure 7.7). Let τ_{process} be the processing time for data preparation for the train scheduling. All the events that happened in the past, i.e., for which the event time is smaller than $t_{\text{current}} + \tau_{\text{process}}$, are known to the event-driven system and the set of these events is denoted as S_{known} . The set of events that will happen in the future is denoted as S_{unknown} .

When an event happens, the state of the system should be updated and some other events

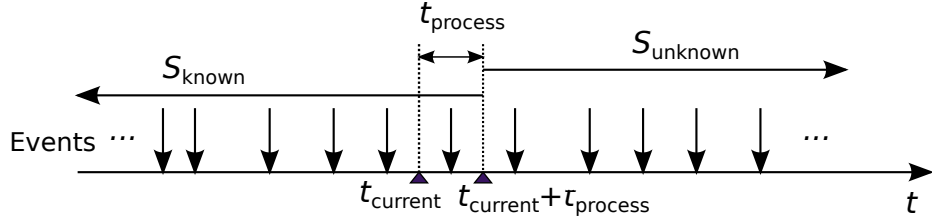
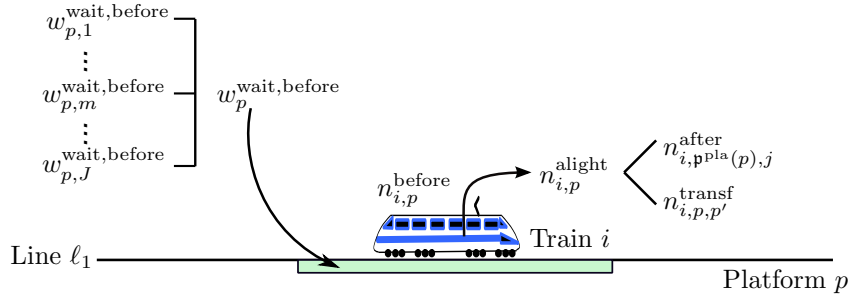
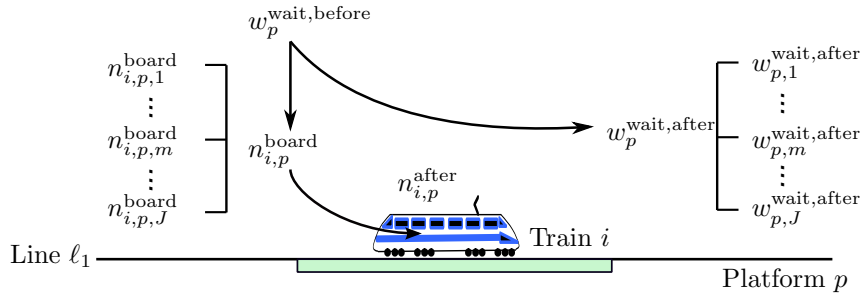


Figure 7.7: Definition of known events and unknown events



(a) Variables immediately before the boarding process of passengers



(b) Variables immediately after the boarding process of passengers

Figure 7.8: Variables for the passenger characteristics before and after the boarding process

may be triggered. For all the events occurring in the given system, the number of passengers waiting at platforms need to be updated. It is important to note that the passenger arrival rate stays the same between two subsequent events. Immediately before event e_k happens, the number of passengers $w_{p_k, m}^{\text{wait, before}}(t_k)$ with destination m that are waiting at platform p_k is updated as follows (see Figure 7.8):

$$w_{p_k, m}^{\text{wait, before}}(t_k) = w_{p_k, m}^{\text{wait, after}}(t_{k'}) + \lambda_{p_k, m}(t_{k'})(t_k - t_{k'}), \quad (7.13)$$

where $t_{k'}$ is the event time of the previous event $e_{k'} = (t_{k'}, Y_{\text{type}, k'}, i_{k'}, p_{k'})$ happening at platform p_k of line ℓ_k (i.e., $p_{k'} = p_k$), $w_{p_k, m}^{\text{wait, after}}(t_{k'})$ is the number of passengers at the platform immediately after event $e_{k'}$, and $\lambda_{p_k, m}(t_{k'})(t_k - t_{k'})$ is the number of passengers that arrive at this platform between $t_{k'}$ and t_k . The total number of waiting passengers $w_{p_k}^{\text{wait, before}}(t_k)$ at platform p_k of line ℓ_k immediately before the event e_k can be calculated as

$$w_{p_k}^{\text{wait, before}}(t_k) = \sum_{m \in S_{\text{sta}}} w_{p_k, m}^{\text{wait, before}}(t_k). \quad (7.14)$$

The waiting time of passengers at a platform is updated when an event occurs. We use $t_{p_k}^{\text{wait}}(t_k)$ to denote the waiting time of the passengers at platform p_k when event e_k occurs, which can be calculated by

$$t_{p_k}^{\text{wait}}(t_k) = t_{p_k}^{\text{wait}}(t_{k'}) + \sum_{m \in S_{\text{sta}}} \left(w_{p_k, m}^{\text{wait, after}}(t_{k'})(t_k - t_{k'}) + \frac{1}{2} \lambda_{p_k, m}(t_{k'})(t_k - t_{k'})^2 \right), \quad (7.15)$$

where $t_{k'}$ is the event time of the previous event $e_{k'}$ that occurred at platform p_k .

In general, the updates of other states and the triggered events caused by the current event depend on the event type of the current event. For λ -change events, only the number of waiting passengers at platforms and the waiting time of these passengers need to be updated. For departure events and arrival events, a detailed description of the updates of other states and the triggered events is given as follows.

Departure events

When a departure event occurs, denoted as e_k , then we have $Y_{\text{type}, k} = \text{'d'}$. Let i_k be the train involved in the event. The number of passengers boarding train i_k at platform p_k on line ℓ_k is equal to the minimum of the number of waiting passengers $w_{p_k}^{\text{wait, before}}(t_k)$ and the remaining space $n_{i_k, p_k}^{\text{remain}}$ on the train after the alighting process of passengers, i.e.

$$n_{i_k, p_k}^{\text{board}} = \min(n_{i_k, p_k}^{\text{remain}}, w_{p_k}^{\text{wait, before}}(t_k)). \quad (7.16)$$

The remaining space $n_{i_k, p_k}^{\text{remain}}$ on train i_k for passengers is

$$n_{i_k, p_k}^{\text{remain}} = C_{\text{max}, i_k} - n_{i_k, \text{p}^{\text{pla}}(p_k)} - n_{i_k, p_k}^{\text{alight}}, \quad (7.17)$$

where C_{max, i_k} is the capacity of train i_k , $\text{p}^{\text{pla}}(p_k)$ is the predecessor platform¹ of platform p_k , $n_{\alpha, \beta}$ is the number of passengers on train α when it departs from platform β , and $n_{\alpha, \beta}^{\text{alight}}$

¹Recall that $\text{p}^{\text{pla}}(p_k)$ is the previous platform on the line to which platform p_k belongs.

is the number of passengers getting off train α at platform β . The calculation for $n_{i_k, p_k}^{\text{alight}}$ will be given in (7.30).

The number of passengers $w_{p_k}^{\text{wait,after}}(t_k)$ left by train i_k at platform p_k , i.e., the number of passengers waiting at the platform immediately after event e_k , is

$$w_{p_k}^{\text{wait,after}}(t_k) = w_{p_k}^{\text{wait,before}}(t_k) - n_{i_k, p_k}^{\text{board}}. \quad (7.18)$$

In addition, we assume that the number of passengers with destination m that are left by train i_k is proportional to the number of waiting passengers. Hence, the number of passengers with destination m left by train i_k at platform p_k can be calculated as

$$w_{p_k, m}^{\text{wait,after}}(t_k) = w_{p_k}^{\text{wait,after}}(t_k) \frac{w_{p_k, m}^{\text{wait,before}}(t_k)}{w_{p_k}^{\text{wait,before}}(t_k)}. \quad (7.19)$$

The number of passengers with destination m that board train i_k at platform p_k is

$$n_{i_k, p_k, m}^{\text{board}} = w_{p_k, m}^{\text{wait,before}}(t_k) - w_{p_k, m}^{\text{wait,after}}(t_k). \quad (7.20)$$

After the boarding process, the number of passengers $n_{i_k, p_k, m}^{\text{after}}$ with destination m that are on board of train i_k is updated as

$$n_{i_k, p_k, m}^{\text{after}} = n_{i_k, p_k, m}^{\text{before}} + n_{i_k, p_k, m}^{\text{board}}, \quad (7.21)$$

and the total number of passengers $n_{i_k, p_k}^{\text{after}}$ on board of train i_k at platform p_k after the boarding process is

$$n_{i_k, p_k}^{\text{after}} = n_{i_k, p_k}^{\text{before}} + n_{i_k, p_k}^{\text{board}}, \quad (7.22)$$

where $n_{i_k, p_k, m}^{\text{before}}$ and $n_{i_k, p_k}^{\text{before}}$ are the number of passengers with destination m and the total number of passengers on board train i before the boarding process of passengers (see more details in the description of arrival events).

The departure event e_k at platform p_k will generate an arrival event at the next platform of the line to which platform p_k belongs, which is described as follows:

$$(a_{i_k, \mathfrak{s}^{\text{pla}}(p_k)}, 'a', i_k, \mathfrak{s}^{\text{pla}}(p_k)),$$

where $a_{i_k, \mathfrak{s}^{\text{pla}}(p_k)}$ is the arrival time of train i_k at platform $\mathfrak{s}^{\text{pla}}(p_k)$. The arrival time $a_{i_k, \mathfrak{s}^{\text{pla}}(p_k)}$ can be calculated by

$$a_{i_k, \mathfrak{s}^{\text{pla}}(p_k)} = d_{i_k, p_k} + r_{i_k, p_k}, \quad (7.23)$$

where d_{i_k, j_k} is equal to t_k and r_{i_k, p_k} is the running time on the track segment between platform p_k and platform $\mathfrak{s}^{\text{pla}}(p_k)$. This arrival event should then be added to the global event list.

Arrival events

If $Y_{\text{type}, k} = 'a'$, i.e., event e_k is the arrival event of train i_k at platform p_k on ℓ_k , the number of passengers $w_{p_k}^{\text{wait,before}}(t_k)$ waiting at platform p_k immediately before this arrival event should be updated by (7.13) and (7.14).

The number of passengers getting off train i_k depends on platform p_k :

- If platform p_k is at the first station of line ℓ_k , then there are no passengers getting off train i_k , i.e.

$$n_{i_k, p_k}^{\text{alight}} = 0. \quad (7.24)$$

In addition, the number of passengers $n_{i_k, p_k, m}^{\text{before}}$ that have destination m and are on board of train i_k immediately before the boarding process is also equal to zero, i.e.

$$n_{i_k, p_k, m}^{\text{before}} = 0, \quad \forall m \in S_{\text{sta}}. \quad (7.25)$$

- If the station j_k to which platform p_k belongs, is not the first station and not a transfer station, then the passengers with destination j_k will get off train i_k . The number of these passengers can be computed as follows:

$$n_{i_k, p_k}^{\text{alight}} = n_{i_k, \text{p}^{\text{pla}}(p_k), j_k}^{\text{after}}, \quad (7.26)$$

where $n_{i_k, \text{p}^{\text{pla}}(p_k), j_k}^{\text{after}}$ is the number of onboard passengers with destination j_k after the boarding process at predecessor platform $\text{p}^{\text{pla}}(p_k)$. Furthermore, we calculate the number of passengers $n_{i_k, p_k, m}^{\text{before}}$ as follows:

$$n_{i_k, p_k, m}^{\text{before}} = n_{i_k, \text{p}^{\text{pla}}(p_k), m}^{\text{after}}, \quad \forall m \in S_{\text{sta}} \setminus \{j_k\}. \quad (7.27)$$

Therefore, the total number of passengers on board train i_k before the boarding process is

$$n_{i_k, p_k}^{\text{before}} = \sum_{m \in S_{\text{sta}} \setminus \{j_k\}} n_{i_k, p_k, m}^{\text{before}}. \quad (7.28)$$

- If the station j_k to which platform p_k belongs is a transfer station, then not only the passengers with destination j_k will get off train i_k , but the passengers with other destinations may also get off train i_k . The splitting rates for the passengers with destination m staying on or getting off train i_k are denoted as $\beta_{p', i_k, m}^{\text{train}}$ for $p' \in P_{j_k}$. The number of passengers $n_{i_k, p_k, m}^{\text{before}}$ that have destination m and are on board of train i_k immediately before the boarding process can be calculated by

$$n_{i_k, p_k, m}^{\text{before}} = \beta_{p_k, i_k, m}^{\text{train}} n_{i_k, \text{p}^{\text{pla}}(p_k), m}^{\text{after}}, \quad \forall m \in S_{\text{sta}}, \quad (7.29)$$

where $n_{i_k, \text{p}^{\text{pla}}(p_k), m}^{\text{after}}$ is the number of onboard passengers with destination m immediately after the boarding process at predecessor platform $\text{p}^{\text{pla}}(p_k)$.

As mentioned in Section 7.2.1, the splitting rate $\beta_{p_k, i_k, j_k}^{\text{train}}$ equals 1 for the passenger flow with destination j_k . All these passengers will get off the train and exit the network from station j_k . For the passengers with destination m with $m \neq j_k$, the passengers staying on line ℓ_k also stays on train i_k . Hence, the number of alighting passengers can be calculated by

$$n_{i_k, p_k}^{\text{alight}} = n_{i_k, \text{p}^{\text{pla}}(p_k)}^{\text{after}} - \sum_{m \in S_{\text{sta}} \setminus \{j_k\}} n_{i_k, p_k, m}^{\text{before}}. \quad (7.30)$$

where $n_{i_k, \text{p}^{\text{pla}}(p_k)}^{\text{after}}$ is in fact equal to the number of passengers on board of train i_k when it arrives at platform p_k and $\sum_{m \in S_{\text{sta}} \setminus \{j_k\}} n_{i_k, j_k, m}^{\text{before}}$ is the total number of passengers

staying on train i_k after the alighting process. The number of transferring passengers $n_{i_k, p_k, p', m}^{\text{transf}}$ that have destination m and transfer from platform p_k to some other platform p' can be calculated by

$$n_{i_k, p_k, p', m}^{\text{transf}} = \beta_{i_k, p', m}^{\text{train}} n_{i_k, \text{ppla}(p_k), m}^{\text{after}}, \quad \forall p' \in P_{j_k} \setminus \{p_k\}. \quad (7.31)$$

The total number of transfer passengers from train i_k is then

$$n_{i_k, p_k}^{\text{transf}} = \sum_{p' \in P_{j_k} \setminus \{p_k\}} \sum_{m \in S_{\text{sta}} \setminus \{j_k\}} n_{i_k, p_k, p', m}^{\text{transf}}. \quad (7.32)$$

The passenger in-vehicle time for trains, denoted as $t_{i_k, p_k}^{\text{in-vehicle}}$ should be updated when an arrival event happens. When arrival event e_k happens, the passenger in-vehicle time, including the running time of train i_k and the dwell time at platform p_k , can be calculated by

$$t_{i_k, p_k}^{\text{in-vehicle}} = n_{i_k, \text{ppla}(p_k)}^{\text{after}} r_{i_k, p_k} + (n_{i_k, \text{ppla}(p_k)}^{\text{after}} - n_{i_k, p_k}^{\text{alight}})(d_{i_k, p_k} - a_{i_k, p_k}), \quad (7.33)$$

where r_{i_k, p_k} , d_{i_k, p_k} , a_{i_k, p_k} are the running time, departure time, and arrival time of train i_k at platform p_k .

In addition, λ -change events will be triggered to increase and decrease the passenger arrival rates. These events can be written as

$$(a_{i_k, p_k} + \theta_{i_k, p_k, p'}^{\text{walk}}, \lambda', -, p_k), \quad \forall p' \in P_{j_k} \setminus \{p_k\}, \quad (7.34)$$

$$(a_{i_k, p_k} + \theta_{i_k, p_k, p'}^{\text{walk}} + \theta_{i_k, p_k, p'}^{\text{duration}}, \lambda', -, p_k), \quad \forall p' \in P_{j_k} \setminus \{p_k\}, \quad (7.35)$$

where ‘-’ is a dummy place holder as there is no train included in these events. The above λ -change events should be added to the global event list.

7.3 Mathematical formulation for the scheduling problem

7.3.1 Performance criteria

In this chapter, we minimize the total travel time of all passengers and the energy consumption of the trains using a weighted sum strategy. Note that we can accommodate other performance criteria all well.

The total energy consumption for all I trains running with J stations can then be formulated as

$$E_{\text{total}} = \sum_{i \in S_{\text{tra}}} \sum_{p \in S_{\text{pla}}} (E_{i, p}^{\text{acc}} + E_{i, p}^{\text{hold}}), \quad (7.36)$$

where $E_{i, p}^{\text{acc}}$ and $E_{i, p}^{\text{hold}}$ can be calculated by (5.17) and (5.18). The total travel time of all passengers includes the passenger waiting time, the passenger in-vehicle time, and the passenger transfer time, which can then be formulated as

$$t_{\text{total}} = \sum_{p \in S_{\text{pla}}} t_p^{\text{wait}} + \sum_{i \in S_{\text{tra}}} \sum_{p \in S_{\text{pla}}} t_{i, p}^{\text{in-vehicle}} + \sum_{i \in S_{\text{tra}}} \sum_{p \in S_{\text{pla}}} t_{i, p}^{\text{transf}} + \sum_{p \in S_{\text{pla}}} t_p^{\text{walk-in}} + \sum_{p \in S_{\text{pla}}} t_p^{\text{walk-out}}. \quad (7.37)$$

Similar as the performance criteria in Section 5.3, a weighted sum strategy can be applied to solve the multi-objective optimization of the train scheduling problem, i.e., we consider

$$f_{\text{opt}} = \frac{E_{\text{total}}}{E_{\text{total,nom}}} + \lambda \frac{t_{\text{total}}}{t_{\text{total,nom}}}, \quad (7.38)$$

where λ is a non-negative weight, and the normalization factors $E_{\text{total,nom}}$ and $t_{\text{total,nom}}$ are the nominal values of the total energy consumption and the total travel time of passengers, respectively. These nominal values can e.g. be determined by a feasible initial schedule. Just as mentioned in Section 6.3, a penalty term of the waiting time of the passengers left by the last train during the scheduling period can be introduced to spread trains over the entire scheduling period.

7.3.2 Constraints

The event times of all the events in the future event set S_{unknown} should satisfy

$$t_k \geq t_{\text{current}} + \tau_{\text{process}}, \quad \forall e_k \in S_{\text{unknown}}. \quad (7.39)$$

In addition, the event times of the departure events and the arrival events should satisfy the operational constraints as follows. The event time t_k for arrival event e_k , i.e., the arrival time a_{i_k,p_k} of train i_k at platform p_k , should satisfy the headway constraints:

$$a_{i_k,p_k} - d_{p^{\text{tra}}(p_k),p_k} \geq h_{p_k,\min}, \quad (7.40)$$

where $d_{p^{\text{tra}}(p_k),p_k}$ is the departure time of the previous train at platform p_k and $h_{p_k,\min}$ is the minimum headway at platform p_k to ensure the safe operation of trains. Furthermore, the event time of the departure event, i.e., the departure time d_{i_k,p_k} of train i_k at platform p_k , should satisfy

$$d_{i_k,p_k} \geq a_{i_k,p_k} + \tau_{i_k,p_k,\min}, \quad (7.41)$$

$$d_{i_k,p_k} \leq a_{i_k,p_k} + \tau_{i_k,p_k,\max}, \quad (7.42)$$

where $\tau_{i_k,p_k,\min}$ and $\tau_{i_k,p_k,\max}$ are the minimal and maximal dwell time for train i_k at platform p_k . The minimal dwell time is affected by the number of passengers getting off and getting on the train, which can be calculated as

$$\tau_{i_k,p_k,\min} = \min \left(\tilde{\tau}_{\min}, \alpha_{1,d} + \alpha_{2,d} n_{i_k,p_k}^{\text{alight}} + \alpha_{3,d} n_{i_k,p_k}^{\text{board}} + \alpha_{4,d} \left(\frac{w_{p_k}^{\text{wait}}(t_k)}{n^{\text{door}}} \right)^3 n_{i_k,p_k}^{\text{board}} \right), \quad (7.43)$$

where $\tilde{\tau}_{\min}$ is the minimum dwell time predefined by railway operator, $\alpha_{1,d}$, $\alpha_{2,d}$, $\alpha_{3,d}$, and $\alpha_{4,d}$ are coefficients that can e.g. be estimated based on historical data, n^{door} is the number of doors of the train, and $w_{p_k}^{\text{wait}}/n^{\text{door}}$ is the number of passengers waiting at each door. Moreover, the departure time d_{i_k,p_k} should satisfy the headway constraint as follows:

$$d_{i_k,p_k} - d_{p^{\text{tra}}(p_k),p_k} \leq h_{p_k,\max}, \quad (7.44)$$

where $h_{p_k,\max}$ is the maximum departure-departure headway between trains at platform p_k to ensure the passenger satisfaction.

Note that the running time $r_{i,p}$ should satisfy

$$r_{i_k,p_k,\min} \leq r_{i_k,p_k} \leq r_{i_k,p_k,\max}, \quad (7.45)$$

where $r_{i_k,p_k,\min}$ and $r_{i_k,p_k,\max}$ are the minimal and maximal running time of train i_k between platform p_k and successor platform $\mathfrak{p}^{\text{pla}}(p_k)$, respectively. The minimum running time is limited by the train characteristics and the condition of the line. The maximum running time is introduced to ensure the passenger satisfaction since if trains run too slow, the passengers may complain.

7.4 Rolling horizon approach and initial conditions

A rolling horizon approach can be adopted to solve the train scheduling problem. In this section, we will discuss the rolling horizon approach in detail and we define the initial conditions for the scheduling problem.

Since passenger demands vary with the time in a daily operation, the train scheduling problem can be solved in a rolling horizon way, by solving the scheduling problem, e.g., every half an hour, so as to adapt the train schedule to passenger demands in real time. This works as follows. First, the train scheduling problem is solved for some period $[t_0, t_{\text{end}}]$ and the trains will be operated according to the resulting optimal schedule. After some period of time t_p , e.g., half an hour, we will run the optimization process again, but now for the period $[t_0 + t_p, t_{\text{end}} + t_p]$ using the known, measured, or estimated states of the system at time $t_0 + t_p$. Once the new optimal schedule is computed, it is executed for t_p time units, and next the whole process is repeated again for the period $[t_0 + 2t_p, t_{\text{end}} + 2t_p]$ and so on, until the end of the daily operation of the urban rail transit system.

When solving the train scheduling problem in a rolling horizon way, some of the variables² will no longer be free variables but will have fixed, known values. Assuming that t_0 is the start time instant of the scheduling period, we now discuss the fixed variables for a line in an urban rail network:

- If train i is in the terminal station at time t_0 , i.e., the arrival time $a_{i-I_{\text{net}},0}$ of train $i - I_{\text{net}}$ at the terminal station will be a known value with $a_{i-I_{\text{net}},0} < t_0$. So $a_{i-I_{\text{net}},0}$ is no longer an unknown variable.
- If train i is at a platform of a station at time t_0 , we use p_{i,t_0} to denote that platform. The arrival time $a_{i,p_{i,t_0}}$ of train i at platform p_{i,t_0} is known. In addition, the departure times, the arrival times, and the running times before platform p_{i,t_0} are also known.
- If train i is running on a segment at t_0 , we use p_{i,t_0} to denote the segment at which train i is running on at t_0 . The departure time $d_{i,p_{i,t_0}}$ of train i at platform p_{i,t_0} is a known time value with $d_{i,p_{i,t_0}} < t_0$. In addition, all the departure times, arrival times, and running times before segment p_{i,t_0} are known. Furthermore, the running time $r_{i,p_{i,t_0}}$ on segment p_{i,t_0} is also fixed since we assume that the schedule of a train can only be changed at platforms. Therefore, the arrival time of train i at platform $\mathfrak{p}^{\text{pla}}(p_{i,t_0})$ is also known.

²When the stop-skipping strategy is included in the train scheduling, the stopping variables should be fixed for trains that are already on their way in order to make sure all the passengers on these trains can arrive at their destinations.

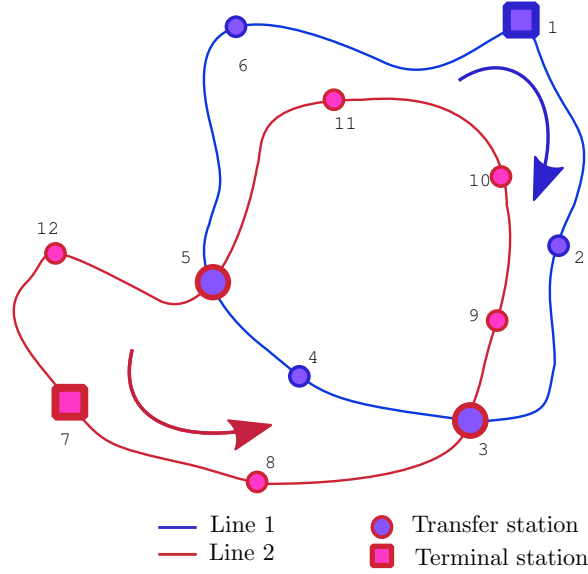


Figure 7.9: Layout of a small urban rail transit network

The number of passengers on the train and the number of passengers waiting at the platform are also known at time t_0 .

7.5 Solution approaches

The train scheduling problem for an urban rail transit network is a nonlinear non-convex programming problem with objective function (7.38) and constraints (7.39)-(7.45). The train scheduling problem in Chapter 5 is also a nonlinear nonconvex programming problem, where several approaches are proposed, such as the gradient-based SQP approach, the gradient-free pattern search method, mixed integer (non)linear programming approach, and the iterative convex programming approach. These approaches can also be applied to the train scheduling problem for an urban rail transit network in this chapter. In addition, the evolutionary algorithms, such as genetic algorithms, can also be applied to this train scheduling problem [14, 31, 144].

7.6 Case study

7.6.1 Set-up

In order to illustrate the event-driven model for urban rail transit networks proposed in this chapter, a small network with two cyclic lines as shown in Figure 7.9 is considered as a test case study. Line 1, i.e., the blue line, has 1 terminal (station 1) and 5 normal stations and line 2, i.e., the red line, has 1 terminal (station 7) and 7 normal stations. The line data of these two lines are given in Table 7.1, where the calculation of the minimum running time is

Table 7.1: Information of the two cyclic lines

Station number (Line 1)	1	2	3	4	5	6
Distance to next station [m]	700	1500	1700	2200	1900	800
Minimal running time [s]	59.3	95.3	104.3	126.8	113.3	63.8

Station number (Line 2)	7	8	3	9	10	11	5	12
Distance to next station [m]	860	1400	1500	1300	1600	1200	1100	730
Minimal running time [s]	66.5	90.8	95.3	86.3	99.8	81.8	77.3	66.5

Table 7.2: Parameters of the trains and the passengers

Property		Symbol	Value
Train mass	[kg]	$m_{e,i}$	$199 \cdot 10^3$
Mass of one passenger	[kg]	m_p	60
Capacity of trains	[passengers]	$C_{i,max}$	1500
Minimum dwell time	[s]	$\tilde{\tau}_{min}$	30
Maximum dwell time	[s]	τ_{max}	150
Coefficients of the minimal dwell time	[s]	$\alpha_{1,d}$	4.002
	[s/passengers]	$\alpha_{2,d}$	0.047
	[s/passengers]	$\alpha_{3,d}$	0.051
	[s/passengers ⁻⁴]	$\alpha_{4,d}$	$1.0 \cdot 10^{-6}$
Coefficients of resistance	[m/s ²]	k_{1i}	0.012
	[s ⁻¹]	k_{2i}	$5.049 \cdot 10^{-4}$
	[m ⁻¹]	k_{3i}	$2.053 \cdot 10^{-5}$

the same as that presented in Section 6.5. For each cyclic line, there are 5 physical trains and the number of train services considered in the train scheduling problem is taken as 7. The parameters of trains and passengers are chosen as in Table 7.2. The passenger arrival rates at stations are given in Table 7.3, where the passenger arrival rates are piecewise constant functions that can change with time in a scheduling period and the passenger arrival rates at terminals, i.e., station 1 and station 7, are equal to 0. In addition, since we only consider one direction of the cyclic lines, no passenger is arriving at the last stations of these two lines, i.e., station 6 and station 12.

At time t_0 (chosen as 2500 s), the initial states of trains for line 1 are as follows: train 1 and train 2 are running from station 4 and 2 to station 5 and station 3, respectively, and their arrival times are fixed, at 2530 s and 2550 s respectively. The number of passengers on train 1 and 2 at time t_0 is given in Table 7.4 and the number of passengers waiting at the platforms of line 1 is shown in Table 7.5. For line 2, the initial states at time t_0 are as follows: train 21, 22, and 23 are running from station 11, 9, and 8 to station 5, 10, and 3, respectively and their arrival times are 2520 s, 2540 s, and 2560 s. The number of passengers on these trains at t_0 is given in Table 7.4 and the number of passengers waiting at platforms of line 2 is shown in Table 7.5. In addition, there are 3 and 2 trains stopping at terminal stations 1 and 7, respectively. We choose the scheduling period as 2500 s, where the schedule of 7

train services is optimized for each line. To ensure the passenger satisfaction, the maximum departure-departure headway at stations is chosen as 400 s. The nominal values for the total travel time, the energy consumption, and the waiting time for the passengers who did not travel in the scheduling period is calculated based on a random feasible schedule, which are $1.454 \cdot 10^7$ s, $3.436 \cdot 10^9$ J, and $7.434 \cdot 10^6$ s respectively.

The model formulation in Section 7.2 distinguishes the splitting rates $\beta_{p,m}^{\text{station}}$ of the passengers just entering the rail network and the splitting rates $\beta_{i,p,m}^{\text{train}}$ of the passenger arriving the transfer stations by trains. For this case study, we simplify the train scheduling model by making³ $\beta_{i,p,m}^{\text{train}}$ equal to $\beta_{p,m}^{\text{station}}(a_{i,j})$ with $a_{i,j}$ the arrival time of train i at station j . After this simplification, the number of decision variables of the problem can be reduced significantly, especially for cases with a large number of trains. In addition, the walk-in and walk-out time for passengers in (7.37) are also taken as zero for the sake of simplicity. and Two solution approaches are proposed to solve the train scheduling problem, i.e., the multistart SQP approach and the genetic algorithm. The SQP method implemented by the `fmincon` function of Matlab optimization toolbox is employed and five feasible initial points are used to solve the optimization problem. For the genetic algorithm, the `ga` function of the global optimization toolbox of Matlab is used.

7.6.2 Results and discussion

The train schedules obtained by the SQP method for line 1 and line 2 are shown in Figures 7.10 and 7.11. The number of passengers on board of trains is shown in Figures 7.12 and 7.13. Since there are no passenger arrivals at terminal stations (station 1 and station 7 for line 1 and line 2), the number of passengers on board of trains should be equal to 0 when trains depart or arrive at terminal stations, which is illustrated in Figures 7.12 and 7.13. In Figure 7.12, it is shown that when train 3 departs from station 2, the number of onboard passengers has already reached the maximum capacity, i.e., 1500 passengers. Similarly, train 2 and train 4 reach their maximum capacity at station 4 and station 3, respectively. So the trains are operated with high passenger load and the operation costs of these trains are smaller when compared with the operation of trains with lower passenger load.

As mentioned before, station 3 and station 5 are transfer stations in the small rail network shown in Figure 7.9. Some passengers need to transfer to arrive at their destinations, e.g., passengers that enter the network at station 2 but have destination 9, 10, or 11 need to transfer at station 3. When trains of line 1 arriving at the platform of transfer station 3, the number of onboard passengers with different destinations is shown in Figure 7.14, where the number of onboard passengers with destination 1, 2, 7, 8 is equal to 0. The passengers with destination 3 (dark purple bars) will get off the train at this station and the passengers with destination 9, 10, and 11 (pink bars) will also get off the train and transfer to line 2. The other passengers will stay on board. It is noted that the passengers with destination 12 choose to stay on the train instead of transferring to line 2 at station 3. This is because these passengers can also transfer at station 5 as shown in Figure 7.15 and this will lead to a shorter travel time. Furthermore, the number of passengers at station 3 for train 1 is zero in Figure 7.14, since train 1 has already passed station 3 at time t_0 . Similarly, the number

³Recall that the splitting rates $\beta_{i,p,m}^{\text{train}}$ for trains are numbers and that the splitting rates $\beta_{p,m}^{\text{station}}$ are piecewise constant functions that have the time as their argument.

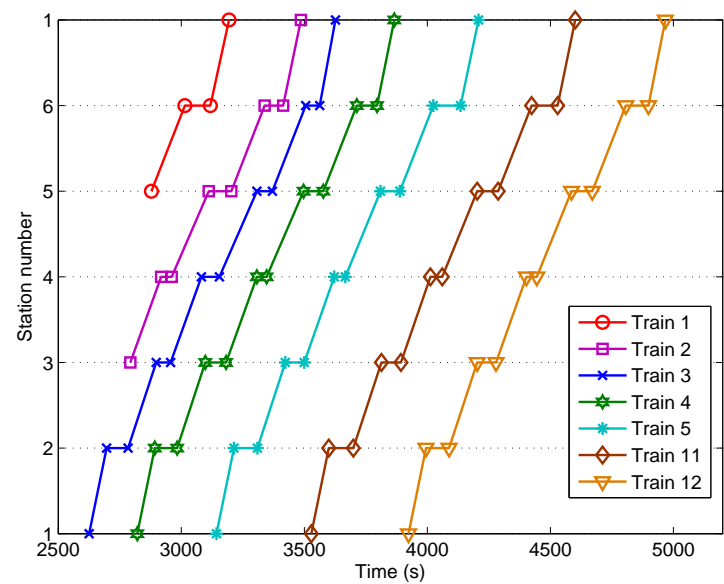


Figure 7.10: Train schedules for line 1 obtained by the SQP method

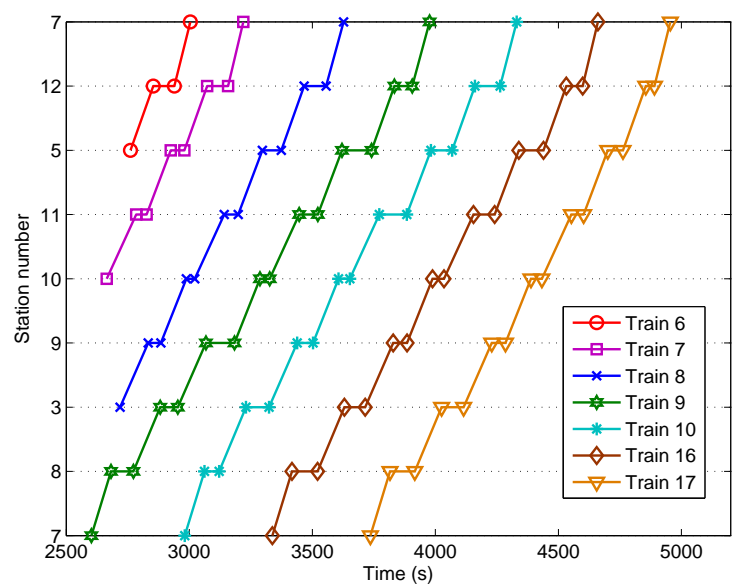


Figure 7.11: Train schedules for line 2 obtained by the SQP method

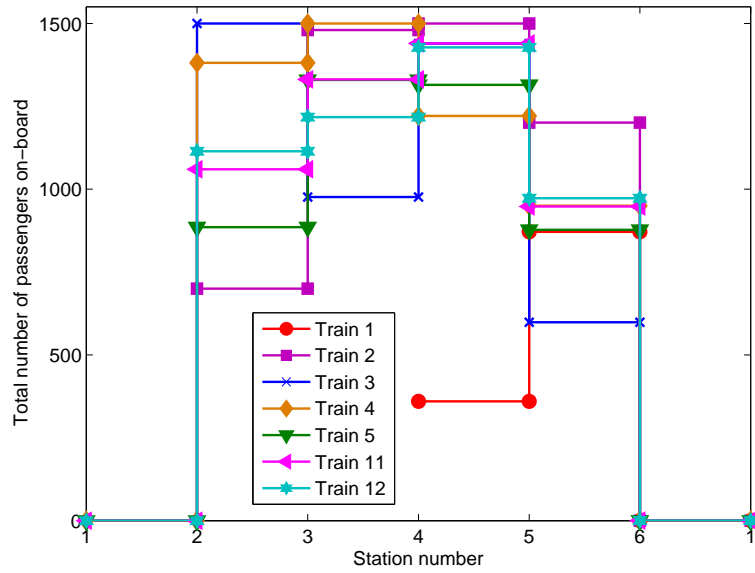


Figure 7.12: Total number of onboard passengers obtained by the SQP method for line 1

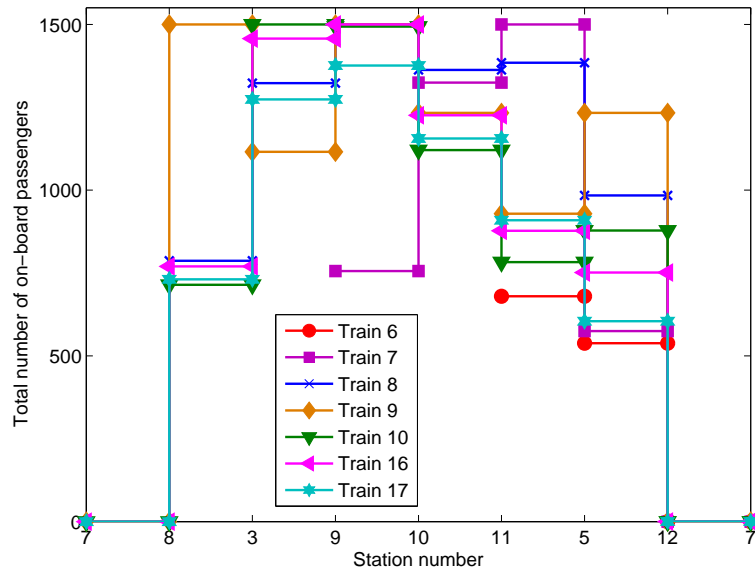


Figure 7.13: Total number of onboard passengers obtained by the SQP method for line 2

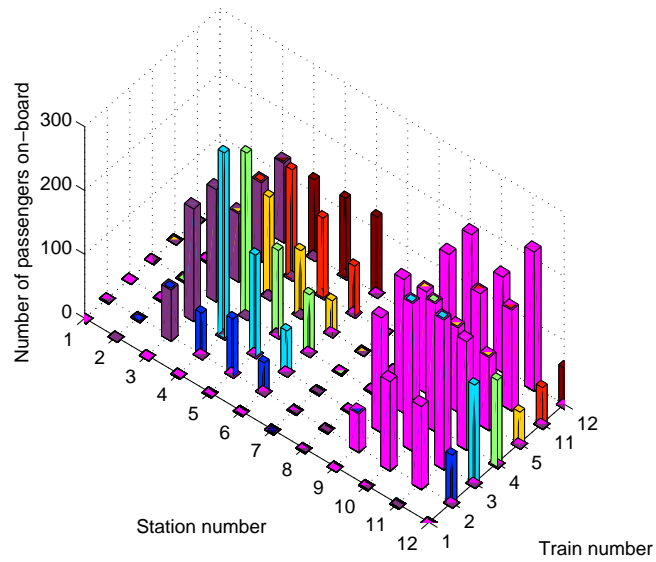


Figure 7.14: Number of onboard passengers with different destinations at transfer station 3 obtained by the SQP method for line 1

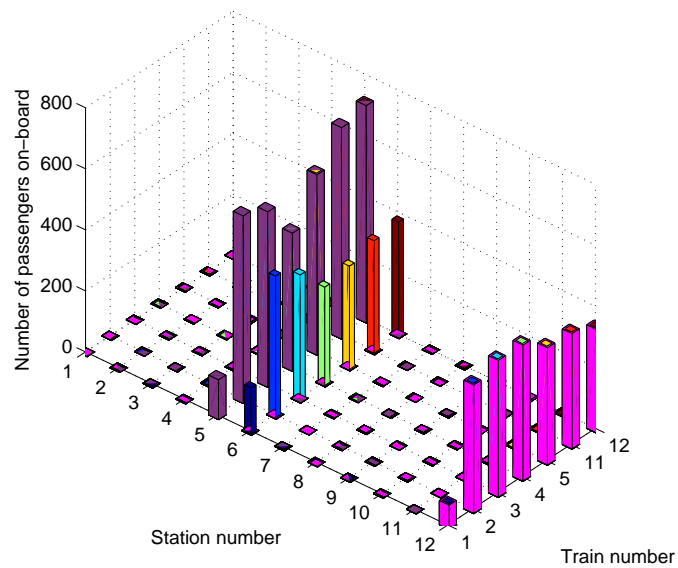


Figure 7.15: Number of onboard passengers with different destinations at transfer station 5 obtained by the SQP method for line 1

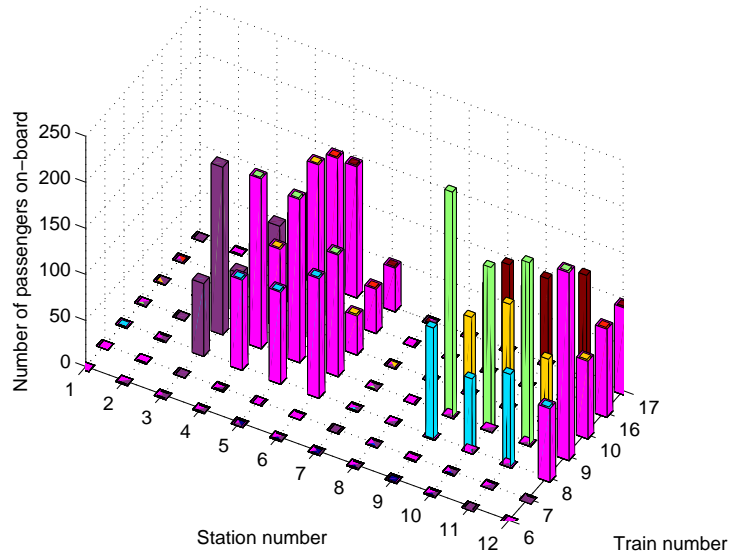


Figure 7.16: Number of onboard passengers with different destinations at transfer station 3 obtained by the SQP method for line 2

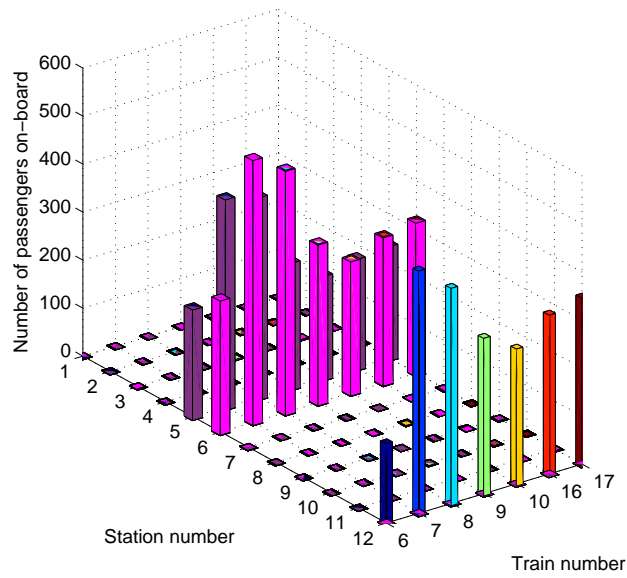


Figure 7.17: Number of onboard passengers with different destinations at transfer station 5 obtained by the SQP method for line 2

Table 7.3: Passenger arrival rates (passengers/s) in the small urban rail network

[illegible]

Table 7.4: Number of passengers on board of trains at time t_0 for the two cyclic lines

Destination station	1	2	3	4	5	6	7	8	9	10	11	12	Total number of passengers
Train 1 (Line 1)	0	0	0	0	130	150	0	0	0	0	0	80	360
Train 2 (Line 1)	0	0	80	70	90	50	0	0	60	140	130	80	700
Train 6 (Line 2)	0	0	0	0	230	280	0	0	0	0	0	170	680
Train 7 (Line 2)	0	0	0	0	160	180	0	0	0	120	134	162	756
Train 8 (Line 2)	0	0	79	98	100	130	0	0	120	80	120	80	787

Table 7.5: Number of waiting passengers at platforms of the two cyclic lines

[illegible][illegible]

Table 7.6: Performance comparison of the SQP method and the genetic algorithm

Solution approaches	SQP method	Genetic algorithm
Objective value [-]	3.522	3.858
Computation time [s]	$4.232 \cdot 10^3$	$5.941 \cdot 10^3$
Energy consumption [J]	$2.677 \cdot 10^9$	$3.220 \cdot 10^9$
Number of passengers that finished their trips [passengers]	$2.993 \cdot 10^4$	$2.550 \cdot 10^4$
Number of passengers that did not finish their trips [passengers]	$9.215 \cdot 10^3$	$1.364 \cdot 10^4$
Travel time for passengers that finished their trips [s]	$1.761 \cdot 10^7$	$1.484 \cdot 10^7$
Waiting time for passengers that did not travel [s]	$3.218 \cdot 10^6$	$7.265 \cdot 10^6$

of onboard passengers with different destinations at station 3 and 5 of line 2 is shown in Figures 7.16 and 7.17.

A comparison of the performance of the two approaches, i.e., the SQP method and the genetic algorithm, is illustrated in Table 7.6, where the values of the objective function, the computation time, the energy consumption of trains, the number and the travel time of the passengers that finished their trips, and the number and the waiting time of the passengers that did not travel are listed. It is observed that the SQP method has a better performance than the genetic algorithm for this case study. In particular, the objective value of the SQP method is about 10% smaller than that obtained by the genetic algorithm. The train schedule obtained by the SQP method has a lower energy consumption and more passengers arrive at their destination. However, the travel time for passengers that finished their trip obtained by the genetic algorithm is smaller than that obtained by the SQP method. The reason for this is that the number of passengers that finished their trips in the schedule obtained by genetic algorithm is smaller than that of the SQP method. In addition, the higher energy consumption of the schedule obtained by genetic means less running time, so the travel time is also less.

For the given case study, the performance of the SQP method is about 10% better than that of the genetic algorithm. In addition, the SQP method also has a smaller computation time. In particular, for the small network with 2 lines given in Figure 7.9, the computation time of the SQP method is about one hour using Matlab on a 64-bit linux operation system running on 1.8 GHz Intel Core2 Duo CPU. Multiple experiments with different scenarios are needed to assess the performance of the solution approaches.

7.7 Conclusions

In this chapter, the train scheduling problem for an urban rail transit network is investigated. We have built an event-driven model with three types of events, i.e., departure events, arrival events, and passenger arrival rate change events at platforms. The splitting of passenger flows and passenger transfers at transfer stations are included in the event-driven model. For the given case study, the SQP method provides a better trade-off between control performance and computational complexity than the genetic algorithm.

An extensive comparison and assessment of the SQP method, the genetic algorithm, and other solution approaches for different scenarios will be a topic for future work. For the cases with multiple lines and a large number of stations and trains, distributed optimization approaches are expected to be applied to solve the train scheduling problem. In addition, we will also investigate the simplification of the proposed model so that the train scheduling problem can be solved in real time.

Chapter 8

Conclusions and Future Research

In this dissertation we have discussed the optimal trajectory planning problem for trains and the real-time train scheduling problem for urban rail transit systems. This final chapter first presents the main contributions of the previous chapters, discusses some remaining open problems, and gives some recommendations for future research.

8.1 Conclusions

The main contributions of the work presented in this dissertation can be summarized as follows:

- We have proposed a new iterative convex programming (ICP) approach to solve the train scheduling problem with constant origin-destination-independent (OD-independent) passenger demands for an urban railway transit line. For the case study we have considered, the ICP approach outperforms other solution approaches, e.g., a sequential quadratic programming (SQP) method, pattern search method, a mixed integer linear programming (MILP) approach, and a mixed integer nonlinear programming (MINLP) approach.
- We have developed a train scheduling model considering a constant origin-destination-dependent (OD-dependent) passenger demand for urban rail transit systems and we have proposed an efficient bi-level approach to shorten the computation time with respect to the standard bi-level approach.
- We have proposed an event-driven model including the splitting of passenger flows and passenger transfer behavior at transfer stations for train scheduling in urban rail transit networks. This model involves three types of events, viz. departure events, arrival events, and passenger arrival rate change events at platforms.

We have also considered the trajectory planning for multiple trains, which included the constraints caused by the fixed and moving block signaling system.

The contents and conclusions of the conducted research in this thesis are now discussed in more detail for each topic.

Optimal trajectory planning for a single train

The optimal trajectory planning problem for a single train under various constraints (e.g. fixed arrival time, varying line resistance, variable speed restrictions, and varying maximum traction force) has been considered, where the objective function is a trade-off between the energy consumption and the riding comfort. First, we have proposed two solution approaches to solve this nonlinear non-convex problem, i.e. the pseudospectral method and an MILP approach.

Simulation results comparing the pseudospectral method, the MILP approach, and a discrete dynamic programming approach show that the pseudospectral method results in the best control performance, but that if the required computation time is also taken into consideration, the MILP approach yields the best overall performance.

Optimal trajectory planning for multiple trains

We have investigated the optimal trajectory planning for multiple trains, where the constraints caused by the leading train in a fixed or moving block signaling system have been included in the formulation. We have developed four solution approaches to solve this optimal control problem for multiple trains, viz. the greedy MILP approach, the simultaneous MILP approach, the greedy pseudospectral method, and the simultaneous pseudospectral method.

Simulation results comparing the greedy MILP approach with the simultaneous MILP approach show that the simultaneous MILP approach yields a better control performance but requires a higher computation time. Moreover, the performance of the greedy and the simultaneous MILP approach has also been compared with that of the greedy and the simultaneous pseudospectral method. The results show that the energy consumption and the end time violations of the greedy MILP approach are slightly larger than those of the greedy pseudospectral method, but the computation time is one to two orders of magnitude smaller. Similarly, the simultaneous pseudospectral method has less energy consumption and less end time violations compared with the simultaneous MILP approach but requires more computation time. For the given case studies, the simultaneous MILP approach yields the best overall performance.

Train scheduling for a single line based on OD-independent passenger demands

The train scheduling model with OD-independent passenger demand for an urban rail transit line has been proposed with the aim of minimizing the total travel time of passengers and the energy consumption of the operation of trains. The resulting train scheduling problem is nonlinear and nonconvex. We have developed a new iterative convex programming (ICP) approach to solve this train scheduling problem and have compared it with a gradient-free nonlinear programming approach (in particular pattern search method), a gradient-based nonlinear programming approach (in particular, an SQP approach), an MINLP approach, and an MILP approach.

It has been shown by simulation results that the performance of the optimal train schedules obtained by the ICP approach, the pattern search method, the SQP approach, and the MNILP approach are close to each other, but the MILP approach has a worse performance probably because of the piecewise affine approximation error. The computation time of the ICP approach is smaller than that of other alternative approaches. Hence, the ICP approach provides the best trade-off between performance and computational complexity.

Train scheduling for a single line based on OD-dependent passenger demands

In order to adapt the train schedule to the OD-dependent passenger demand in an urban rail transit line, a stop-skipping strategy is adopted to reduce the passenger travel time and the energy consumption. We have proposed an efficient bi-level approach to solve this problem, in which a threshold method is applied to obtain a good initial solution for the full problem and subsequently the search space for the variables is limited to enhance the efficiency.

It is shown by the obtained simulation results that the bi-level approach yields a better performance than the efficient bi-level approach but at the cost of a higher computation time. In addition, the overall performance of the train scheduling strategy with stop-skipping is better than that of an all-stop strategy.

Train scheduling for networks with time-varying OD-independent passenger demands

For the train scheduling of urban rail transit networks, we have presented an event-driven model, which characterizes the time varying OD-dependent passenger demand, the splitting of passenger flows, and the passenger transfer behavior at transfer stations. There are three types of events in the model, i.e., departure events, arrival events, and passenger arrival rate change events at platforms. The resulting train scheduling problem is a real-valued nonlinear nonconvex problem, which can be solved by a gradient-free nonlinear programming approach (in particular pattern search method), a gradient-based nonlinear programming approach (in particular, an SQP approach), and an MILP approach.

For the given simulation experiment, we have found that the SQP method provides a better trade-off between control performance and computational complexity with respect to genetic algorithm.

8.2 Recommendations for future research

In this section, we briefly present some of the open problems that still have to be tackled based on the contents of this dissertation. Furthermore, we give some additional directions for future research.

8.2.1 Optimal trajectory planning and train scheduling

The optimal trajectory planning problem for a single train and for multiple trains with constraints caused by signaling system has been investigated. Furthermore, the train scheduling based on passenger demand for urban rail transit systems has been considered in this thesis. There are still many issues to be tackled. However, the conflict between modeling accuracy and computation efficiency is one of the most important aspects that should be taken into

account during the research. If the accuracy is increased, then the computational complexity will also increase, and vice versa. Hence, it is important to achieve a balanced trade-off between accuracy and computational efficiency. Furthermore, with the increasing complexity of the problem formulation, the computational complexity will also increase and it is difficult to solve the problem in real time using the current approaches. Below, we give some topics to improve the accuracy and some methods to increase the computational efficiency.

Modeling accuracy

- Hybrid control methods for the train trajectory planning.

In the literature of the train trajectory planning, there are four optimal operation regimes for the operation of trains, viz. maximum acceleration, cruising, coasting, and maximum braking. Hybrid control methods, e.g. switched nonlinear systems, can be adopted to obtain optimal train trajectories where real train characteristics and line conditions could be taken into account.

- Microscopic modeling.

The model used in this thesis for train scheduling is a macroscopic model, where the details of the infrastructure (e.g., block sections and signals) are not considered. In addition, the layout of the terminals is not taken into account in the train scheduling model. The terminus is usually a bottleneck of an urban rail transit system, where the turnaround times of trains affect the minimum headway and there may exist route conflicts. Therefore, for conflict detection and feasibility checking of the train schedules, it is necessary to use a microscopic model. Efficient trajectory planning algorithms can then first be used to determine time-position profiles for trains and next blocking time¹ theory [57, 98] can then be applied to identify conflicts.

- Short-turning of trains.

In some urban rail transit lines, there may exist a zone with much higher passenger demands (e.g., the part in the city center) compared with other parts of the line. In order to transport passengers efficiently and to avoid too many passengers waiting at platforms (this may cause safety problems), it is important to develop methods to integrate short-turning strategies into train scheduling, especially for the zone with higher passenger demands and for the peak hours. In such a strategy, there are short-turning trips only serving the stations in the zone with higher passenger demands and full-length trips serving all stations of the line [20, 111].

- Train scheduling with incidents.

When an incident has happened in an urban rail transit network, e.g., a part of a line cannot be used due to technical problems, an updated train schedule needs to be proposed as soon as possible to provide alternative route choices to passengers to enable them to arrive at their destinations. Based on the information about the incidents, several possible measures can be considered, e.g., efficient methods can be developed

¹The blocking time is the total elapsed time that a section of track is exclusively allocated to a train and therefore blocked for other trains.

to determine the short-turning strategy for the other parts of the line without technical problems.

- Robust train trajectory planning and train scheduling.

Since there are stochastic disturbances during the operation of trains and the knowledge we obtain are incomplete and uncertain, the parameters of trains characteristics, the running times, dwell times, etc. may vary within a certain range. It is better that the trajectory planning and train scheduling can adapt to the variation of parameters, running times, and dwell times. A sensitivity analysis of the parameters, running times, and dwell times could be carried out to obtain the crucial ones. New trajectory planning and scheduling methods should be developed to take variations in these crucial factors into account.

- Extensive experiments including disturbances and randomness

An extensive comparison and assessment of the approaches proposed in this thesis should be performed via different scenarios for train and line characteristics, OD-(in)dependent passenger demands, etc. In addition, different set-ups for various case studies should also be investigated for different numbers of trains, different urban rail network layouts, different layouts of stations, etc. Furthermore, a micro-simulation model could be used as simulation models instead of using the same model for both prediction and simulation as was done in this thesis. Moreover, randomness caused by disturbances, model errors, etc. in experiments should be taken into account in case studies to make the conclusions more general.

Computational efficiency

- Hierarchical optimization techniques.

When more and more trains are involved in the optimal trajectory planning problem, the proposed greedy approach and simultaneous approach will become slow, especially for the simultaneous approach. Similarly, with the increasing size of urban rail networks, the size of the train scheduling problem will grow dramatically and the computation time will also increase. Therefore, a hierarchical optimization structure where a big network could be decomposed into smaller networks, can be developed to address trajectory planning and train scheduling at different levels [30, 116]. The train trajectory planning could be defined based on different aggregation levels, i.e., rough trajectories can be optimized at the high level and more accurate trajectories can be obtained at the lower level based on more detailed models. Similarly, a rough timetable could also be obtained for the train scheduling problem in the higher level based on a simple model for the whole network. In the lower level, a more detailed model can be employed for the train scheduling.

- Distributed optimization techniques.

A distributed optimization structure could also be presented for the trajectory planning for multiple trains and the train scheduling problem for large urban rail networks, where a large railway network can be divided into multiple smaller subnetworks to decrease the size of the problem and to reduce the computation time. The subnetworks

can then exchange information with each other, make their own decisions by taking the information provided by others into account, negotiate with each other, and finally converge to a global equilibrium. Game theory [97] and distributed model predictive control methods [18] can be applied to design the communication and coordination schemes for the subnetworks.

- Fast optimization approaches.

When solving the trajectory planning problem for multiple trains and the train scheduling for large urban rail networks, the optimization approaches proposed in this thesis are too slow for real-time applications. Fast optimization approaches, e.g., exploring the explicit structure of MILP problems and fast model predictive control [126], can be developed to reduce the computation time. In addition, rule-based control and case-based control approaches could be adopted, where the trade-off between computation time and performance should be assessed.

8.2.2 Additional directions for future research

Some more general research directions for the operation of urban rail transit systems are presented as follows:

- Full integration and interfacing between scheduling and operation control for urban rail transit networks.

A hierarchical approach can be adopted for the full integration of scheduling and operation control for urban rail transit networks. The high level focuses on the whole railway network, where e.g. a max-plus algebra approach [119] may be used to optimize train schedules based on a single model for the whole network. At the middle level, the whole network may be divided into several subnetworks and a more detailed model (may include turnaround times, blocking times, passenger transfers, etc.) could be used for determining more detailed train schedules based on the rough schedules given by the high level. The high level should coordinate different subnetworks of the middle level to make train schedules feasible. Single urban rail transit lines could be considered at the low level, where the train schedules can be further refined and the rolling stock and crews could also be scheduled. An additional lower level may be introduced for the trajectory planning and the operation of trains, where the detailed characteristics of trains are taken into account.

The interfaces between these levels should be developed since they are important for the effectiveness of the full integration. Furthermore, a monitoring system for the whole urban rail system should be built to obtain real-time information (e.g., departure and arrival times of trains, running speeds and position of trains, number of passengers entering the network) for the scheduling and train control. Furthermore, approaches for the estimation of OD-dependent passenger demands and the prediction of traffic states, etc. should also be investigated since they are important issues in the full integration.

- Passenger behavior and route choices.

Urban rail transit networks are becoming more and more complex, especially in large

cities like Beijing, Shanghai, Tokyo, and Paris. Passengers can arrive at their destinations through multiple available routes. There are many factors that influence the route choice of passengers, such as travel time, transfer time, number of transfers, crowdedness of the route. However, nowadays rail operators can only give general route advices and passengers have the freedom to choose routes themselves. The passenger behavior, e.g. how passengers adapt to personalized route advices and to updated train schedules, can be investigated. In addition, new train scheduling models could be developed, which can integrate or anticipate on the route choice behavior of passengers.

- Multi-operator networks.

In urban rail transit systems, there may exist multiple rail operators that operate different lines in one network. These rail operators may be competing with each other and each of them would want to attract more passengers to increase its market share and profits. As a consequence, the information about e.g. the lines and passenger demands could then result in being only partly shared with other operators. Cooperative scheduling based on partial information sharing could then be considered [65].

The optimization models and algorithms proposed in this thesis can also be applied to control and to optimize other transportation systems such as:

- Bus transit systems.

Bus transit systems with buses running on conventional roads to carry numerous passengers on short journeys, share many similarities with urban rail transit systems, such as variety in origin-destination passenger demands, uncertainty in dwell times and running times, and fixed routes. However, bus transit systems are operated with lower capacities, lower passenger demands, and more degrees of freedoms on the operations (e.g., overtaking, crossing, and turnaround). The model and solution approaches for train scheduling and trajectory planning provided in this thesis can be extended to bus transit systems, where the effect of other road traffic e.g. cars and of traffic signals should be taken into account.

- Multi-car elevator systems.

Elevator systems are usually controlled by a centralized upper-level controller, which determines where a car should stop to load or to unload passengers. Similar as the train scheduling considering passenger demands, elevator dispatching is also characterized by time-varying passenger arrival patterns. Therefore, the scheduling and trajectory planning methods presented in this thesis can be applied to reduce the waiting time of passengers and the energy consumption.

- Automated guided vehicles (AGVs).

AGVs are automated vehicles that can load, unload, and transport goods in port container terminals or manufacturing systems. In general, there are two types of AGVs: one type follows prescribed routes (indicated by markers or wires in the floor) or it has to select its route from a prescribed route set, while the other type uses vision or lasers for navigation and it can go to anywhere in the specified area. In addition,

energy consumption is very crucial for AGVs with battery or fuels since they could run out of energy. The models and solution algorithms presented in this thesis can also be extended for AGVs with prescribed routes, in particular the limited-energy aspect should be included in the scheduling problem.

Bibliography

- [1] T. Albrecht. Automated timetable design for demand-oriented service on suburban railways. *Public Transport*, 1(1):5–20, May 2009.
- [2] T. Albrecht, C. Gassel, A. Binder, and J. van Luipen. Dealing with operational constraints in energy efficient driving. In *IET Conference on Railway Traction Systems (RTS 2010)*, pages 1–7, Birmingham, UK, April 2010.
- [3] T. Albrecht, C. Gassel, and A. Binder. 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*, pages 1–4, Leuven, Belgium, June 2011.
- [4] W.O. Assis and B.E.A. Milani. Generation of optimal schedules for metro lines using model predictive control. *Automatica*, 40(8):1397–1404, August 2004.
- [5] A. Atamturk and M.W.P. Savelsbergh. Integer-programming software systems. *Annals of Operations Research*, 140(1):67–124, November 2005.
- [6] K.E. Atkinson. *An Introduction to Numerical Analysis*. John Wiley & Sons, New York, NJ, USA, 1978.
- [7] S.I. Azuma, J.I. Imura, and T. Sugie. Lebesgue piecewise affine approximation of nonlinear systems. *Nonlinear Analysis: Hybrid Systems*, 4(1):92–102, February 2010.
- [8] V. M. Becerra. Psopt optimal control solver user manual - release 3, August 2010. URL <http://www.psopt.org/downloads>.
- [9] V.M. Becerra. Solving complex optimal control problems at no cost with psopt. In *Proceedings of IEEE Multi-Conference on Systems and Control*, pages 1391–1396, Yokohama, Japan, September 2010.
- [10] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, March 1999.
- [11] T. Berthold, G. Gamrath, A.M. Gleixner, S. Heinz, T. Koch, and Y. Shinano. Solving mixed integer linear and nonlinear problems using the SCIP Optimization Suite. ZIB-Report 12-17, Zuse Institute Berlin, Berlin, Germany, 2012.

- [12] J.T. Betts. Survey of numerical methods for trajectory optimization. *Journal of Guidance, Control, and Dynamics*, 21(2):193–207, March 1998.
- [13] J.T. Betts. A direct approach to solving optimal control problems. *Computing in Science & Engineering*, 1(3):73–75, August 2002.
- [14] Y.V. Bocharnikov, A.M. Tobias, C. Roberts, S. Hillmans, and C.J. Goodman. Optimal driving strategy for traction energy saving on DC suburban railways. *IEE Proceedings - Electric Power Applications*, 1(5):675–682, September 2007.
- [15] P.T. Boggs and J.W. Tolle. Sequential quadratic programming. *Acta Numerica*, 4(1):1–51, January 1995.
- [16] R.H. Byrd, M.E. Hribar, and J. Nocedal. An interior point algorithm for large-scale nonlinear programming. *SIAM Journal on Optimization*, 9(4):877–900, August 1999.
- [17] V. Cacchiani and P. Toth. Nominal and robust train timetabling problems. *European Journal of Operational Research*, 219(3):727–737, June 2009.
- [18] E. Camponogara, D. Jia, B.H. Krogh, and S. Talukdar. Distributed model predictive control. *IEEE Control System Magazine*, 22(1):44–52, February 2002.
- [19] C. Canuto, M.Y. Hussaini, A. Quarteroni, and T.A. Zang. *Spectral Methods in Fluid Dynamics*. Springer-Verlag, New York, NJ, USA, 1988.
- [20] A. Ceder. Optimal design of transit short-turn trips. *Transportation Research Record*, 1221:8–22, January 1989.
- [21] C.S. Chang and S.S. Sim. Optimizing train movements through coast control using genetic algorithms. *IEE Proceedings - Electric Power Applications*, 144(1):65–73, January 1997.
- [22] C.S. Chang and D.Y. Xu. Differential evolution based tuning of fuzzy automatic train operation for mass rapid transit system. *IEE Proceedings - Electric Power Applications*, 147(3):206–212, May 2000.
- [23] J.F. Cordeau, P. Toth, and D. Vigo. A survey of optimization models for train routing and scheduling. *Transportation Science*, 32(4):380–420, November 1998.
- [24] F. Corman, D. Pacciarelli, A. D’Ariano, and M. Pranzo. Evaluation of green wave policy in real-time railway traffic management. *Transportation Research Part C*, 17(6):607–616, December 2009.
- [25] F. Corman, A. D’Ariano, D. Pacciarelli, and M. Pranzo. Bi-objective conflict detection and resolution in railway traffic management. *Transportation Research Part C: Emerging Technologies*, 20(1):79–94, February 2012.
- [26] J.E. Cury, F.A.C. Gomide, and M. J. Mendes. A methodology for generation of optimal schedules for an underground railway systems. *IEEE Transactions on Automatic Control*, 25(2):217–222, April 1980.

- [27] A. D'Ariano and T. Albrecht. Running time re-optimization during real-time timetable perturbations. In *Computers in Railways X*, WIT Press, pages 531–540, Prague, Czech Republic, July 2006.
- [28] A. D'Ariano, M. Pranzoand, and I.A. Hansen. Conflict resolution and train speed coordination for solving real-time timetable perturbations. *IEEE Transactions on Intelligent Transportation Systems*, 8(2):208–222, June 2007.
- [29] A. D'Ariano, D. Pacciarelli, and M. Pranzo. Assessment of flexible timetables in real-time traffic management of a railway bottleneck. *Transportation Research Part C: Emerging Technologies*, 16(2):232–245, April 2008.
- [30] B. De Schutter, J. Ploeg, L.D. Baskar, G. Naus, and H. Nijmeijer. Hierarchical, intelligent and automatic controls. In A. Eskandarian, editor, *Handbook of Intelligent Vehicles*, chapter 5, pages 81–116. Springer, London, UK, 2012.
- [31] Y. Ding, Y. Bai, F. Liu, and B. Mao. Simulation algorithm for energy-efficient train control under moving block system. In *Proceedings of the 2009 World Congress on Computer Science and Information Engineering*, pages 498–502, Los Angeles, CA, USA, March 2009.
- [32] T.A.B. Dollevoet, F. Corman, A. D'Ariano, and D. Huisman. An iterative optimization framework for delay management and train scheduling. Technical Report EI 2012-10, Erasmus School of Economics (ESE), Rotterdam, The Netherlands, 2012.
- [33] H. Dong, B. Ning, B. Cai, and Z. Hou. Automatic train control system development and simulation for high-speed railways. *IEEE Circuits and Systems Magazine*, 10(2): 6–18, June 2010.
- [34] X.J. Elberlein. *Real-Time Control Strategies in Transit Operations: Models and Analysis*. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 1995.
- [35] X.J. Elberlein, N.H.M. Wilson, and C. Barnhart. The real-time deadheading problem in transit operation control. *Transportation Research Part B*, 32(2):77–100, February 1998.
- [36] X.J. Elberlein, N.H.M. Wilson, and D. Bernstein. The holding problem with real-time information available. *Transportation Science*, 35(1):1–18, February 2001.
- [37] G. Elnagar, M.A. Kazemi, and M. Razzaghi. The pseudospectral Legendre method for discretizing optimal control problems. *IEEE Transactions on Automatic Control*, 40(10):1793–1796, October 1995.
- [38] B. Fornberg and D.M. Sloan. A review of pseudospectral methods for solving partial differential equations. *Acta Numerica*, 3(1):203–267, January 1994.
- [39] R. Franke, M. Meyer, and P. Terwiesch. Optimal control of the driving of trains. *Automatisierungstechnik*, 50(12):606–614, December 2002.

- [40] R. Franke, P. Terwiesch, and M. Meyer. An algorithm for the optimal control of the driving of trains. In *Proceedings of the 39th IEEE Conference on Decision and Control*, pages 2123–2128, Sydney, Australia, December 2003.
- [41] M. Freyss, R. Giesen, and J. Carlos Muñoz. Continuous approximation for skip-stop operation in rail transit. *Transportation Research Part C*, 36:419–433, November 2013.
- [42] L. Fu, Q. Liu, and P. Calamai. Real-time optimization model for dynamic scheduling of transit operations. *Transportation Research Record*, 1857:48–55, January 2003.
- [43] R. Gao. *Railway Signal Operation Basis*. China Railway Publishing House, Beijing, China, 2008.
- [44] C. Gassel and T. Albrecht. The impact of request stops on railway operations. In *Proceedings of the 3rd International Seminar on Railway Operations Modeling and Analysis*, pages 1–15, Zürich, Switzerland, April 2009.
- [45] F. Gembicki and Y.Y. Haimes. Approach to performance and sensitivity multiobjective optimization the goal attainment method. *IEEE Transactions on Automatic Control*, 20(6):769–771, December 1975.
- [46] P. Gerber. Class Re465 locomotives for heavy-haul freight service. *Proceedings of the Institution of Mechanical Engineers Part F: Journal of Rail and Rapid Transit*, 215(F1):25–35, January 2001.
- [47] N.S.A. Ghoneim and S.C. Wirasinghe. Optimum zone structure during peak periods for existing urban rail lines. *Transportation Research Part B*, 20(1):7–18, February 1986.
- [48] K. Ghoseiri, F. Szidarovszky, and M.J. Asgharpour. A multi-objective train scheduling model and solution. *Transportation Research Part B*, 38(10):927–952, December 2004.
- [49] P.E. Gill, W. Murray, and M.A. Saunders. Snopt: An sqp algorithm for large-scale constrained optimization. *Society for Industrial and Applied Mathematics Journal on Optimization*, 12(4):979–1006, August 2002.
- [50] Q. Gong, W. Kang, N. Bedrossian, F. Fahroo, P. Sekhavat, and K. Bollino. Pseudospectral optimal control for military and industrial applications. In *Proceedings of the 46th IEEE Conference on Decision and Control*, pages 4128–4142, New Orleans, LA, USA, December 2007.
- [51] Q. Gong, I.M. Ross, W. Kang, and F. Fahroo. Connections between the covector mapping theorem and convergence of pseudospectral methods for optimal control. *Computational optimization and applications*, 41(3):307–335, December 2008.
- [52] C.J. Goodman and S. Murata. Metro traffic regulation from the passenger perspective. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, 215(2):137–147, March 2001.

- [53] R.M.P Goverde. *Punctuality of railway operations and timetable stability analysis*. Ph.D. thesis, Delft University of Technology, Delft, Netherlands, 2005.
- [54] Q. Gu, X. Lu, and T. Tang. Energy saving for automatic train control in moving block signaling system. In *Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems*, pages 1305–1310, Washington, DC, USA, October 2011.
- [55] Q. Gu, T. Tang, F. Cao, and Y. Song. Energy-efficient train operation in urban rail transit using real-time traffic information. *IEEE Transactions on Intelligent Transportation Systems*, 15(3):1216–1233, February 2014.
- [56] S.H. Han, Y.S. Byen, J.H. Baek, T.K. An, S.G. Lee, and H.J. Park. An optimal automatic train operation (ATO) control using genetic algorithms (GA). In *Proceedings of the IEEE Region 10 Conference (TENCON 99)*, volume 1, pages 360–362, Cheju Island, South Korea, August 1999.
- [57] I.A. Hansen and J. Pachl. *Railway, Timetable & Traffic: Analysis, Modelling, Simulation*. Eurailpress, Hamburg, Germany, 2008.
- [58] E. Hellström, J. Åslund, and L. Nielsen. Design of an efficient algorithm for fuel-optimal look-ahead control. *Control Engineering Practice*, 18(11):1318–1327, November 2010.
- [59] A. Higgins, E. Kozan, and L. Ferreira. Optimal scheduling of trains on a single line track. *Transportation Research Part B*, 30(2):147–161, April 1996.
- [60] R.J. Hill and L.J. Bond. Modeling moving block railway signaling system using discrete-event simulations. In *Proceedings of the 1995 IEEE/ASME Joint Railroad Conference*, pages 105–111, Baltimore, MD, USA, April 1995.
- [61] R. Hooke and T.A. Jeeves. “Direct search” solution of numerical and statistical problems. *Journal of the Association for Computing Machinery*, 8(2):212–229, April 1961.
- [62] P. Howlett. The optimal control of a train. *Annals of Operations Research*, 98(1-4): 65–87, December 2000.
- [63] P.G. Howlett and P.J. Pudney. *Energy-Efficient Train Control*. Advances in Industrial Control. Springer-Verlag, London, UK, 1995.
- [64] D. Huerlimann and A.B. Nash. *OPENTRACK - Simulation of Railway Networks, User Manual Version 1.3*. Institute for Transportation Planning and Systems, ETH Zürich, Switzerland, 2003.
- [65] D. Huisman, L.G. Kroon, R.M. Lentink, and M.J.C.M. Vromans. Operations research in passenger railway transportation. *Statistica Neerlandica*, 59(4):467–497, November 2005.
- [66] K. Ichikawa. Application of optimization theory for bounded state variable problems to the operation of a train. *Bulletin of the Japanese Society of Mechanical Engineering*, 11(47):857–865, November 1968.

- [67] IEEE 1474.1. IEEE Standard for Communications-Based Train Control (CBTC) Performance and Functional Requirements, 2004.
- [68] A. Ingimundarson, C. Ocampo-Martinez, and A. Bemporad. Suboptimal model predictive control of hybrid systems based on mode-switching constraints. In *Proceedings of the 46th IEEE Conference on Decision and Control*, pages 5264–5269, New Orleans, LA, USA, December 2007.
- [69] R.P. Kanwal. *Generalized Functions: Theory and Technique*. Academic Press, New York, NJ, USA, 1983.
- [70] B. Kersbergen, T. van den Boom, and B. De Schutter. Reducing the time needed to solve the global rescheduling problem for railway networks. In *Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 791–796, The Hague, The Netherlands, October 2013.
- [71] M.B. Khan and X. Zhou. Stochastic optimization model and solution algorithm for robust double-track train-timetabling problem. *IEEE Transactions on Intelligent Transportation Systems*, 11(2):81–89, March 2010.
- [72] E. Khmelnitsky. On an optimal control problem of train operation. *IEEE Transactions on Automatic Control*, 45(7):1257–1266, July 2000.
- [73] T.J. Kiresuk and R.E. Sherman. Goal attainment scaling A general method for evaluating comprehensive community mental health programs. *IEEE Transactions on Automatic Control*, 4(6):443–453, December 1968.
- [74] H. Ko, T. Koseki, and M. Miyatake. Application of dynamic programming to optimization of running profile of a train. In *Computers in Railways IX*, WIT Press, volume 15, pages 103–112, Southampton, Boston, MA, USA, September 2004.
- [75] D. Kraay, P.T. Harker, and B. Chen. Optimal pacing of trains in freight railroads: model formulation and solution. *Operations Research*, 39(1):82–99, January 1991.
- [76] J. Törnquist Krasemann. Design of an effective algorithm for fast response to the re-scheduling of railway traffic during disturbances. *Transportation Research Part C: Emerging Technologies*, 20(1):62–78, October 2012.
- [77] M. Kvasnica, P. Grieder, and M. Baotić. Automatic derivation of optimal piecewise affine approximations of nonlinear systems. In *Proceedings of the 18th IFAC World Congress*, pages 8675–8680, Milano, Italy, August–September 2011.
- [78] C.M. Kwan and C.S. Chang. Application of evolutionary algorithm on a transportation scheduling problem - the mass rapid transit. In *Proceedings of the IEEE Congress on Evolutionary Computation*, volume 2, pages 987–994, Edinburgh, UK, September 2005.
- [79] Y.J. Lee. Mathematical modeling for optimizing skip-stop rail transit operation strategy using genetic algorithm. Technical report, Morgan State University, Department of Transportation and Urban Infrastructure Studies, Baltimore, MD, USA, 2012.

- [80] L. Leemis. *Reliability: Probabilistic Models and Statistical Methods*. Prentice Hall, Englewood Cliffs, NJ, USA, 1995.
- [81] X. Li, D. Wang, K. Li, and Z. Gao. A green train scheduling model and fuzzy multi-objective optimization algorithm. *Applied Mathematical Modeling*, 37(4):617–629, February 2013.
- [82] Y. Li and M.J. Cassidy. A generalized and efficient algorithm for estimating transit route ODs from passenger counts. *Transportation Research Part B*, 41(1):114–125, January 2007.
- [83] C. Liebchen. *Periodic Timetable Optimization in Public Transport*. Ph.D. thesis, Technique University of Berlin, Berlin, Germany, 2006.
- [84] C. Liebchen. The first optimized railway timetable in practice. *Transportation Science*, 42(4):420–435, November 2008.
- [85] T. Lin and N.H.M. Wilson. Dwell time relationships for light rail systems. *Transportation Research Record*, 1361:287–295, February 1992.
- [86] J. Linderoth and T. Ralphs. Noncommercial software for mixed-integer linear programming. In J. Karlof, editor, *Integer Programming: Theory and Practice*, Operations Research Series, pages 253–303. CRC Press, Boca Raton, FL, USA, 2005.
- [87] R. Liu and I.M. Golovicher. Energy-efficient operation of rail vehicles. *Transportation Research Part A: Policy and Practice*, 37(10):917–931, October 2003.
- [88] Q. Lu and X. Feng. Optimal control strategy for energy saving in trains under the four-aspected fixed auto block system. *Journal of Modern Transportation*, 19(2): 82–87, June 2011.
- [89] M. Lüthi. *Improving the Efficiency of Heavily Used Railway Networks through Integrated Real-Time Rescheduling*. Ph.D. thesis, ETH Zürich, Switzerland, 2009.
- [90] B. Mao. *The Calculation and Design of Train Operations*. People Transport press, Beijing, China, 2008.
- [91] R. Martí. Advaced multi-start methods. In *International Series in Operations Research & Management Science: Handbook of metaheuristics*, volume 146, pages 265–281. Springer-Verlag, New York, NY, USA, March 2010.
- [92] L. Meng and X. Zhou. Robust single-track train dispatching model under a dynamic and stochastic environment: A scenario-based rolling horizon solution approach. *Transportation Research Part B: Methodological*, 45(7):1080–1102, August 2011.
- [93] S. Midya and R. Thottappillil. An overview of electromagnetic compatibility challenges in European Rail Traffic Management system. *Transportation Research Part C: Emerging Technologies*, 16(5):515–534, October 2008.

- [94] T. Norio, T. Yoshiaki, T. Noriyuki, H. Chikara, and M. Kunimitsu. Train rescheduling algorithm which minimizes passengers' dissatisfaction. In *Lecture Notes in Computer Science: Innovations in Applied Artificial Intelligence*, volume 3533, pages 829–838. Springer-Verlag, Berlin, Germany, March 2005.
- [95] C. Ocampo-Martinez, A. Ingimundarson, A. Bemporad, and V. Puig. Suboptimal hybrid model predictive control : Application to sewer networks. In *Proceedings of the 17th IFAC world Congress*, pages 10021–10026, Seoul, Korea, July 2008.
- [96] M.M. Okrent. *Effects of Transit Service Characteristics on Passenger Waiting Times*. M.S. thesis, Northwestern University, Evanston, USA, 1974.
- [97] M.J. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, Cambridge, MA, USA, 1994.
- [98] J. Pachl. *Railway Operation and Control - Second Edition*. Gorham Printing, Centralia, USA, 2009.
- [99] L.V. Pearson. *Moving Block Signalling*. Ph.D. thesis, Loughborough University of Technology, Loughborough, UK, 1973.
- [100] H. Peng. *Urban Rail Transit System*. China Communication Press, Beijing, China, 2008.
- [101] E.R. Petersen, A.J. Taylor, and C.D. Martland. An introduction to computer-assisted train dispatch. *Journal of Advanced Transportation*, 20(1):63–72, March 1986.
- [102] A.V. Rao, D.A. Benson, C. Darby, M.A. Patterson, C. Francolin, and I. Sanders. Algorithm 902: Gpops, a matlab software for solving multiple- phase optimal control problems using the Gauss pseudospectral method. *ACM Transactions on Mathematical Software*, 37(2):22:1–22:39, April 2010.
- [103] B.P. Rochard and F. Schmid. A review of methods to measure and calculate train resistances. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, 214(4):185–199, April 2000.
- [104] I.M. Ross. *User's Manual for DIDO: A MATLAB Application Package for Solving Optimal Control Problems*. Tomlab Optimization Inc, Pullman, WA, USA, 2004.
- [105] I.M. Ross and F. Fahroo. Legendre pseudospectral approximations of optimal control problems. In W. Kang, C. Borges, and M. Xiao, editors, *New Trends in Nonlinear Dynamics and Control and their Applications*, volume 295 of *Lecture Notes in Control and Information Science*, pages 327–342. Springer-Verlag, Berlin, Germany, 2003.
- [106] I.M. Ross and F. Fahroo. Pseudospectral knotting methods for solving optimal control problems. *Journal of Guidance, Control, and Dynamics*, 27(3):397–405, May 2004.
- [107] P. Rutquist and M. Edvall. *PROPT: MATLAB Optimal Control Software*. Tomlab Optimization Inc, Pullman, US, 2008.

- [108] T. Schank. A fast algorithm for computing the running-time of trains by infinitesimal calculus. In *Proceedings of RAILROME 2011*, volume 1, pages 1–16, Rome, Italy, February 2011.
- [109] A. Schrijver. *Theory of Linear and Integer Programming*. John Wiley & sons,, Chichester, UK, 1998.
- [110] J. Simon and P.G. Furth. Generating a bus route O-D matrix from on-off data. *Journal of Transportation Engineering*, 111(6):583–593, November 1985.
- [111] P.D. Site and F. Filippi. Service optimization for bus corridors with short-turn strategies and variable vehicle size. *Transportation Research Part A*, 32(1):19–38, January 1998.
- [112] O.V. Stryk. User’s guide for dircol (version 2.1): A direct collocation method for the numerical solution of optimal control problems. Technical report, Technical University of Munich, Munich, Germany, 2000.
- [113] O. Sundström and L. Guzzella. A generic dynamic programming matlab function. In *Proceedings of the 18th IEEE International Conference on Control Applications, Part of 2009 IEEE Multi-Conference on Systems and Control*, pages 1625–1630, Saint Petersburg, Russia, July 2009.
- [114] B. Szpigel. Optimal train scheduling on a single line railway. In *Proceedings of the International Conference on Operational Research*, volume 6, pages 344–351, Amsterdam, The Netherlands, August 1972.
- [115] H. Takeuchi, C.J. Goodman, and S. Sone. Moving block signaling dynamics: performance measures and re-starting queued electric trains. *IEE Proceedings - Electric Power Applications*, 150(4):483–492, July 2003.
- [116] A.N. Tarău, B. De Schutter, and J. Hellendoorn. Hierarchical route control in DCV-based baggage handling systems. *International Journal of Services Operations and Informatics*, 6(1/2):5–29, January 2011.
- [117] The Mathworks Inc. *Optimization Toolbox for use with Matlab: Users Guide*. The Mathworks Inc., Natick, MA, USA, 1999.
- [118] The Mathworks Inc. *Genetic Algorithm and Direct Search Toolbox for Use with MATLAB: Users Guide*. The Mathworks Inc., Natick, MA, USA, 2004.
- [119] T.J.J. van den Boom and B. De Schutter. Dynamic railway network management using switching max-plus-linear models. In *Proceedings of the 11th IFAC Symposium on Control in Transportation Systems*, pages 343–348, Delft, The Netherlands, August 2006.
- [120] T.J.J. van den Boom, B. Kersbergen, and B. De Schutter. Structured modeling, analysis, and control of complex railway operations. In *Proceedings of the 51st IEEE Conference on Decision and Control*, pages 7366–7371, Maui, Hawaii, December 2012.

- [121] P. Vansteenwegen and D. Van Oudheusden. Developing railway timetables which guarantee a better service. *European Journal of Operational Research*, 173(1):337–350, August 2006.
- [122] P. Vansteenwegen and D. Van Oudheusden. Decreasing the passenger waiting time for an intercity rail network. *Transportation Research Part B*, 41(4):478–492, May 2007.
- [123] M. Vařak, M. Baotić, N. Perić, and M. Bago. Optimal rail route energy management under constraints and fixed arrival time. In *Proceedings of the European Control Conference*, pages 2972–2977, Budapest, Hungary, August 2009.
- [124] F.J. Vázquez-Abad and L. Zubieta. Ghost simulation model for the optimization of an urban subway system. *Discrete Event Dynamic Systems*, 15(3):207–235, September 2005.
- [125] H. Wang, F. Schmid, L. Chen, C. Roberts, and T. Xu. A topology-based model for railway train control systems. *IEEE Transactions on Intelligent Transportation Systems*, 14(2):819–827, June 2013.
- [126] Y. Wang and S. Boyd. Fast model predictive control using online optimization. *IEEE Transactions on Intelligent Transportation Systems*, 18(2):267–278, March 2010.
- [127] Y. Wang, B. De Schutter, B. Ning, N. Groot, and T.J.J. van den Boom. Optimal trajectory planning for trains using mixed integer linear programming. In *Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems (ITSC 2011)*, pages 1598–1604, Washington, DC, USA, October 2011.
- [128] Y. Wang, B. De Schutter, T.J.J. van den Boom, and B. Ning. 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 (CTS'2012)*, pages 158–163, Sofia, Bulgaria, September 2012.
- [129] Y. Wang, B. De Schutter, T. van den Boom, and B. Ning. Optimal trajectory planning for trains under a moving block signaling system. In *Proceedings of the 12th European Control Conference*, pages 4556–4561, Zürich, Switzerland, July 2013.
- [130] Y. Wang, B. De Schutter, T.J.J. van den Boom, and B. Ning. Optimal trajectory planning for trains – A pseudospectral method and a mixed integer linear programming approach. *Transportation Research Part C*, 29:97–114, April 2013.
- [131] Y. Wang, B. De Schutter, T.J.J. van den Boom, and B. Ning. A suboptimal control scheme of multiple trains based on mode vector constraints. In *Proceedings of the 5th International Seminar on Railway Operations Modelling and Analysis (RailCopenhagen)*, Copenhagen, Denmark, May 2013.
- [132] Y. Wang, B. De Schutter, T.J.J. van den Boom, B. Ning, and T. Tang. Real-time scheduling in urban rail transit operations control. In *Proceedings of the 1st IEEE International Conference on Intelligent Rail Transportation (2013 IEEE ICIRT)*, Beijing, China, August–September 2013. Paper 108.

- [133] Y. Wang, B. De Schutter, T.J.J. van den Boom, B. Ning, and T. Tang. Real-time scheduling for trains in urban rail transit systems using nonlinear optimization. In *Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 1334–1339, The Hague, The Netherlands, October 2013.
- [134] Y. Wang, B. De Schutter, T.J.J. van den Boom, and B. Ning. Optimal trajectory planning for trains under fixed and moving signaling systems using mixed integer linear programming. *Control Engineering Practice*, 22(1):44–56, January 2014.
- [135] Y. Wang, B. De Schutter, T.J.J. van den Boom, B. Ning, and T. Tang. Efficient real-time train scheduling for urban rail transit systems using iterative convex programming. Technical report, Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands, March 2014. Submitted to a journal.
- [136] Y. Wang, B. De Schutter, T.J.J. van den Boom, B. Ning, and T. Tang. Efficient bi-level approach for urban rail transit operation with stop-skipping. Technical report, Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands, March 2014. Submitted to a journal.
- [137] Y. Wang, B. De Schutter, T.J.J. van den Boom, B. Ning, and T. Tang. Origin-destination dependent train scheduling problem with stop-skipping for urban rail transit systems. In *Proceedings of the 93rd Annual Meeting of the Transportation Research Board*, Washington, DC, January 2014.
- [138] Y.M. Wang. *Quantification Analysis on the Energy Factors of Urban Rail Transit System*. M.S. thesis, Beijing Jiaotong University, Beijing, China, 2011.
- [139] H.P. Williams. *Model Building in Mathematical Programming*. Wiley, New York, NY, USA, 1993.
- [140] H.P. Williams. *Model Building in Mathematical Programming (4th ed.)*. John Wiley & Sons, Chichester, UK, 1999.
- [141] K.K. Wong and T.K. Ho. Dwell-time and run-time control for dc mass rapid transit railways. *IET Electric Power Applications*, 1(6):956–966, November 2007.
- [142] R.C.W. Wong, T.W.Y. Yuen, K.W. Fung, and J.M.Y. Leung. Optimizing timetable synchronization for rail mass transit. *Transportation Science*, 42(1):57–69, February 2008.
- [143] S.C. Wong and C.O. Tong. Estimation of time-dependent origindestination matrices for transit networks. *Transportation Research Part B*, 32(1):35–48, January 1998.
- [144] X. Yang, X. Li, Z. Gao, H. Wang, and T. Tang. A cooperative scheduling model for timetable optimization in subway systems. *IEEE Transactions on Intelligent Transportation Systems*, 14(1):438–447, March 2013.
- [145] B. Zhang and Y. Lu. Research on dwelling time modeling of urban rail transit. *Traffic & Transportation*, 27(12):48–52, December 2011.

- [146] J. Zhao, A. Rahbee, and N. H. M. Wilson. Estimating a rail passenger trip origin-destination matrix using automatic data collection systems. *Computer-Aided Civil and Infrastructure Engineering*, 22(5):376–387, July 2007.

Symbols and Abbreviations

List of symbols

Below follows a list of the most frequently used symbols in this thesis.

Chapter 3

m	mass [kg] of a train
g	gravitational acceleration [m/s ²]
ρ	rotating mass factor [-]
v	speed [m/s] of a train
$V_{\max}(s)$	speed limit [m/s] along the track
s	position [m] of a train
u	traction or braking force [N] working on a train
u_{\max}, u_{\min}	maximum traction and braking force [N]
$R_b(v)$	roll resistance and air resistance [N] of a train
$R_l(s, v)$	line resistance [N] caused by track grade, curves, and tunnels
$\alpha(s)$	slope along the track [rad]
$r(s)$	radius [m] of the curve along the track
$l_t(s)$	length [m] of tunnels along the track
$f_c(r(s))$	curve resistance [N]
f_t	tunnel resistance [N]
T	given running time [s] for a train
s_k	boundary [m] of discrete space interval k
Δs_k	length [m] of discrete space interval k
$E(k)$	kinetic energy [J] of a train at position s_k
E_{\min}	minimum kinetic energy [J]
$t(k)$	passing time [s] of a train at position s_k

Chapter 4

v_{\max}	speed code [m/s] for a green signal aspect in a three-aspect FBS system
v_{yellow}	speed code [m/s] for a yellow signal aspect in a three-aspect FBS system

v_{\min}	speed code [m/s] for a red signal aspect in a three-aspect FBS system
$H_{\min, \text{FBS}}$	minimum headway [s] between trains in a FBS system
$H_{\min, \text{MBS}}$	minimum headway [s] between trains in a MBS system
$s_{\text{FB}, m}$	boundary [m] of fixed block section m
L_a	length [m] of a block in a FBS system
L_t^L	length [m] of the leading train
L_s	length [m] of the secure section
S_{SM}	safety margin [m] for braking of the following train
v_{\max}^F	maximum speed [m/s] of the following train
t_r^F	reaction time [s] of the driver and equipment of the following train
L_t^F	distance [m] that the following train may travel during the reaction time
a_b^F	deceleration [m/s ²] of the following train
t_d^L	station dwell time [s] of the leading train
$v^F(t)$	speed [m/s] of the following train at time instant t
a_{acc}^L	acceleration [m/s ²] of the leading train
t_{safe}^F	safety time margin [s] caused by safety distance margin and train length

Chapter 5

i	train number
j	station number
s_j	track section length [m] between station j and station $j + 1$
$d_{i,j}$	departure time [s] of train i at station j
$a_{i,j}$	arrival time [s] of train i at station j
$\tau_{i,j}$	dwell time [s] of train i at station j
$r_{i,j}$	running time [s] of train i at station j
$v_{i,j}$	train speed [m/s] of the holding phase for train i on segment j
$a_{i,j}^{\text{acc}}$	acceleration [m/s ²] for train i on segment j
$a_{i,j}^{\text{dec}}$	deceleration [m/s ²] for train i on segment j
h_0	minimum headway between two successive trains
k_{1i}, k_{2i}, k_{3i}	resistance coefficients for train i
λ_j	passenger arrival rate [passenger/s] at station j
ρ_j	passenger alighting proportion when a train arrives at station j
$C_{i,\max}$	maximum capacity [passengers] of train i
$w_{i,j}^{\text{wait}}$	number of passengers waiting for train i at station j
$w_{i,j}$	number of passengers left immediately after the departure of train i at station j
$n_{i,j}^{\text{remain}}$	remaining capacity [passengers] of train i at station j after the passenger alighting process
$n_{i,j}^{\text{board}}$	number of passengers boarding train i at station j
$n_{i,j}^{\text{alight}}$	number of passengers alighting from train i at station j
$n_{i,j}$	number of passengers on train i when it departs from station j
$\tau_{i,j,\max}, \tau_{i,j,\min}$	maximum and minimum dwell time [s]
$\alpha_{1,d}, \dots, \alpha_{4,d}$	coefficients of the minimum dwell time

n_{door}	number of doors of the train
$t_{\text{wait},i,j}$	passenger waiting time [s] at station j for train i
$t_{\text{in-vehicle},i,j}$	passenger in-vehicle time [s] at station j for train i
$E_{i,j}^{\text{acc}}$	energy consumption [J] of the acceleration phase at station j for train i
$E_{i,j}^{\text{hold}}$	energy consumption [J] of the holding phase at station j for train i
$E_{i,j}^{\text{dec}}$	energy consumption [J] of the deceleration phase at station j for train i
$t_{\text{total,nom}}$	nominal value [s] of the travel time of passengers
$E_{\text{total,nom}}$	nominal value [J] of the total energy consumption

Chapter 6

S_{skip}	skipping set
$h_{0,\text{dep}}$	minimum departure headway [s] at terminal station
$h_{0,\text{arr}}$	minimum arrival headway [s] at terminal station
$\tau_{0,\text{min}}$	minimum dwell time [s] at terminal station
C_0^{her}	capacity [passengers] of the terminal station
$y_{i,j}$	binary variable to indicate whether train i stops at station j or not
$\lambda_{j,m}$	passenger arrival rate [passengers/s] at station j for passenger with destination m
$w_{i,j,m}^{\text{wait}}$	number of passengers with destination m waiting for train i at station j
$w_{i,j,m}$	number of passengers with destination m remaining at station j immediately after the departure of train i
$w_{i,j}^{\text{want-to-board}}$	number of passengers who want to board train i at station j
$n_{i,j,m}^{\text{board}}$	number of passengers with destination m boarding train i at station j

Chapter 7

S_{ln}	set of urban rail transit lines
S_{sta}	set of stations
S_{pal}	set of platforms
p	platform index
$\text{p}^{\text{pla}}(p)$	predecessor of platform p
$\text{s}^{\text{pla}}(p)$	successor of platform p
$\text{p}^{\text{tra}}(i)$	predecessor of train i
$\text{s}^{\text{tra}}(i)$	successor of train i
e_k	event k
t_k	time instant [s] at which event e_k occurs
$Y_{\text{type},k}$	event type (departure, arrival, λ -change) of event k
i_k	index of the train corresponding to event e_k
p_k	index of the platform corresponding to event e_k
$\lambda_{j,m}^{\text{station}}$	passenger arrival rates [passengers/s] at station j with destination m
$\beta_{p,m}^{\text{station}}$	splitting rate of passenger flows that arrive at station j with destination m

$a_{0,p}^{\text{walk}}, b_{0,p}^{\text{walk}}$	coefficients for average walking time from the entrance to platform p
$\theta_p^{\text{walk-in}}$	average walking time [s] for passengers from entrances to platform p
$t_p^{\text{walk-in}}$	total walking time [s] for passengers from entrances to platform p
$\beta_{i,p,m}^{\text{train}}$	splitting rate to platform p for passengers that are on board of train i and have destination m
$t_p^{\text{walk-out}}$	total walking time [s] for passengers from platform p to exit station
$a_{p,p'}^{\text{walk}}, b_{p,p'}^{\text{walk}}$	coefficients for average walking time from platform p to platform p'
$\theta_{i,p,p'}^{\text{walk}}$	average walking time [s] for transfer passengers alighting from train i from platform p to platform p'
$a_{p,p'}^{\text{duration}}, b_{p,p'}^{\text{duration}}$	coefficients for the duration time of the transfer process from platform p to platform p'
$\theta_{i,p,p'}^{\text{duration}}$	duration time [s] for the transfer process from platform p to platform p' for train i
$w_{p_k}^{\text{wait,before}}$	number of passengers waiting at platform p_k before event e_k occurs
$w_{p_k}^{\text{wait,after}}$	number of passengers waiting at platform p_k immediately after event e_k

List of abbreviations

The following abbreviations are used in this thesis:

ATP	automatic train protection
ATO	automatic train operation
ATS	automatic train supervision
DDP	discrete dynamic programming
FBS	fixed blocking signaling
ICP	iterative convex programming
MBS	moving blocking signaling
MILP	mixed integer linear programming
MINLP	mixed integer nonlinear programming
PWA	piecewise-affine
SQP	sequential quadratic programming
OD-dependent	origin-destination-dependent
OD-independent	origin-destination-independent

TRAIL Thesis Series

The following list contains the most recent dissertations in the TRAIL Thesis Series. For a complete overview of more than 100 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.

Wang, Y., *Optimal Trajectory Planning and Train Scheduling for Railway Systems*, T2014/7, November 2014, TRAIL Thesis Series, the Netherlands

Wang, M., *Generic Model Predictive Control Framework for Advanced Driver Assistance Systems*, T2014/6, October 2014, TRAIL Thesis Series, the Netherlands

Kecman, P., *Models for Predictive Railway Traffic Management*, T2014/5, October 2014, TRAIL Thesis Series, the Netherlands

Davarynejad, M., *Deploying Evolutionary Metaheuristics for Global Optimization*, T2014/4, June 2014, TRAIL Thesis Series, the Netherlands

Li, J., *Characteristics of Chinese Driver Behavior*, T2014/3, June 2014, TRAIL Thesis Series, the Netherlands

Mouter, N., *Cost-Benefit Analysis in Practice: A study of the way Cost-Benefit Analysis is perceived by key actors in the Dutch appraisal practice for spatial-infrastructure projects*, T2014/2, June 2014, TRAIL Thesis Series, the Netherlands

Ohazulike, A., *Road Pricing mechanism: A game theoretic and multi-level approach*, T2014/1, January 2014, TRAIL Thesis Series, the Netherlands

Cranenburgh, S. van, *Vacation Travel Behaviour in a Very Different Future*, T2013/12, November 2013, TRAIL Thesis Series, the Netherlands

Samsura, D.A.A., *Games and the City: Applying game-theoretical approaches to land and property development analysis*, T2013/11, November 2013, TRAIL Thesis Series, the Netherlands

Huijts, N., *Sustainable Energy Technology Acceptance: A psychological perspective*, T2013/10, September 2013, TRAIL Thesis Series, the Netherlands

Zhang, Mo, *A Freight Transport Model for Integrated Network, Service, and Policy Design*, T2013/9, August 2013, TRAIL Thesis Series, the Netherlands

Wijnen, R., *Decision Support for Collaborative Airport Planning*, T2013/8, April 2013,

TRAIL Thesis Series, the Netherlands

Wageningen-Kessels, F.L.M. van, *Multi-Class Continuum Traffic Flow Models: Analysis and simulation methods*, T2013/7, March 2013, TRAIL Thesis Series, the Netherlands

Taneja, P., *The Flexible Port*, T2013/6, March 2013, TRAIL Thesis Series, the Netherlands

Yuan, Y., *Lagrangian Multi-Class Traffic State Estimation*, T2013/5, March 2013, TRAIL Thesis Series, the Netherlands

Schreiter, Th., *Vehicle-Class Specific Control of Freeway Traffic*, T2013/4, March 2013, TRAIL Thesis Series, the Netherlands

Zaerpour, N., *Efficient Management of Compact Storage Systems*, T2013/3, February 2013, TRAIL Thesis Series, the Netherlands

Huibregtse, O.L., *Robust Model-Based Optimization of Evacuation Guidance*, T2013/2, February 2013, TRAIL Thesis Series, the Netherlands

Fortuijn, L.G.H., *Turborotonde en turboplein: ontwerp, capaciteit en veiligheid*, T2013/1, January 2013, TRAIL Thesis Series, the Netherlands

Gharehgozli, A.H., *Developing New Methods for Efficient Container Stacking Operations*, T2012/7, November 2012, TRAIL Thesis Series, the Netherlands

Duin, R. van, *Logistics Concept Development in Multi-Actor Environments: Aligning stakeholders for successful development of public/private logistics systems by increased awareness of multi-actor objectives and perceptions*, T2012/6, October 2012, TRAIL Thesis Series, the Netherlands

Dicke-Ogenia, M., *Psychological Aspects of Travel Information Presentation: A psychological and ergonomic view on travellers response to travel information*, T2012/5, October 2012, TRAIL Thesis Series, the Netherlands

Wismans, L.J.J., *Towards Sustainable Dynamic Traffic Management*, T2012/4, September 2012, TRAIL Thesis Series, the Netherlands

Hoogendoorn, R.G., *Swiftly before the World Collapses: Empirics and Modeling of Longitudinal Driving Behavior under Adverse Conditions*, T2012/3, July 2012, TRAIL Thesis Series, the Netherlands

Carmona Benitez, R., *The Design of a Large Scale Airline Network*, T2012/2, June 2012, TRAIL Thesis Series, the Netherlands

Schaap, T.W., *Driving Behaviour in Unexpected Situations: A study into the effects of drivers compensation behaviour to safety-critical situations and the effects of mental workload, event urgency and task prioritization*, T2012/1, February 2012, TRAIL Thesis Series, the Netherlands

Muizelaar, T.J., *Non-recurrent Traffic Situations and Traffic Information: Determining preferences and effects on route choice*, T2011/16, December 2011, TRAIL Thesis Series, the Netherlands

Samenvatting

Optimale Trajectplanning en Treinroostering voor Spoorwegsystemen

Veilige, snelle, punctuele, energie-efficiënte en comfortabele spoorwegsystemen zijn belangrijk voor spooroperatoren, passagiers en het milieu. Door de toenemende energieprijzen en vanwege milieuoverwegingen is het reduceren van energieverbruik één van de voornaamste doelstellingen voor spoorwegsystemen geworden. Tegelijkertijd is het belangrijk om passagiers veilig en efficiënt te vervoeren, mede gezien de drastische toename van het aantal passagiers in stedelijke spoornetwerken. Het onderzoek dat in dit proefschrift wordt gepresenteerd, is in de hoofdzaak gericht op het bepalen en ontwikkelen van wiskundige modellen en oplossingsmethoden om de reistijd van passagiers te verkorten en om het energieverbruik in spoorwegsystemen te reduceren. In het bijzonder wordt gekeken naar de reistijd van passagiers bij de treinroostering, terwijl er tegelijkertijd ook rekening wordt gehouden met de hoeveelheid passagiers in stedelijke spoornetwerken. Met de energie-efficiëntie wordt tijdens de treinroostering én tijdens het besturen van de treinen rekening gehouden.

De hoofdonderwerpen die in dit proefschrift worden behandeld, kunnen als volgt worden samengevat:

- **Optimale trajectplanning voor een enkele trein.** We hebben het optimale trajectplanningsprobleem bestudeerd voor een enkele trein onder verschillende operationele omstandigheden, waaronder wisselende lijnweerstand, variabele snelheidslimieten en een wisselende maximale aandrijvingskracht. De doelfunctie van het optimalisatieprobleem is een afweging tussen energieconsumptie en rijcomfort. We hebben twee methoden voorgesteld om dit optimale regelprobleem op te lossen, namelijk een gemengd-integer-lineaire-programmering (MILP) aanpak en een pseudospectrale aanpak. Simulatieresultaten vergelijken de MILP-aanpak, de pseudospectrale aanpak en een discrete dynamische-programmeringsmethode en laten zien dat de pseudospectrale methode resulteert in de beste regelprestatie, maar dat de MILP-aanpak de beste algehele prestatie biedt als de benodigde rekentijd ook in beschouwing wordt genomen.
- **Optimale trajectplanning voor verscheidene treinen.** We hebben het optimale trajectplanningsprobleem voor verscheidene treinen met een vast-blok seinsysteem en met een bewegend-blok seinsysteem onderzocht. We hebben vier oplossingsme-

thoden voorgesteld: de gulzige (in het Engels: *greedy*) MILP-aanpak, de simultane MILP-aanpak, de gulzige pseudospectrale aanpak en de simultane pseudospectrale aanpak. Simulatieresultaten laten zien dat in vergelijking met de gulzige aanpak, de simultane aanpak een betere regelprestatie biedt, maar een langere rekentijd vergt. Daarnaast zijn de afwijkingen van de gewenste eindtijd bij de MILP-aanpak enigszins groter dan die van de pseudospectrale aanpak, terwijl de rekentijd van de MILP-aanpak één tot twee ordes van grootte kleiner is dan die van de pseudospectrale methode.

- **Treinroostering voor een enkele trein gebaseerd op een OD-onafhankelijk passagiersaantallen.** Het treinroosteringsprobleem voor een stedelijke spoorlijn werd beschouwd met als doel het minimaliseren van de totale reistijd van passagiers en van het energieverbruik van de trein. De vertrektijden, rijtijden en verblijftijden van de treinen zijn geoptimaliseerd gebaseerd op oorsprong-bestemming-onafhankelijke (OD-onafhankelijke) passagiersaantallen. We hebben een nieuwe iteratieve convexe programmering- (ICP) aanpak ontwikkeld om dit treinroosteringsprobleem op te lossen. De prestatie van de ICP-aanpak is vergeleken met alternatieve methoden, zoals niet-lineaire programmeringsmethoden, een gemengd-integer-niet-lineaire-programmering (MINLP) aanpak en een MILP-aanpak. In een casus bleek de ICP-aanpak voor het treinroosteringsprobleem de beste afweging te bieden tussen regelprestatie en rekencomplexiteit.
- **Treinroostering voor een enkele lijn gebaseerd op OD-afhankelijke passagiersaantallen.** Voor een enkele stedelijke spoorlijn hebben we een halte-overslaan-strategie gebruikt om de reistijden van de passagiers en het energieverbruik verder te reduceren, gebaseerd op OD-afhankelijke passagiersaantallen. Het treinroosteringsprobleem met de halte-overslaan-strategie resulteert in een MINLP-probleem; we hebben een efficiënte twee-laagsoptimalisatiemethode voorgesteld om dit probleem op te lossen. De halte-overslaan-strategie is presteert beter dan de altijd-stoppen-strategie. Simulatieresultaten laten zien dat de twee-laagsaanpak een betere regelprestatie biedt dan de efficiënte twee-laagsaanpak, maar dit gaat ten koste van een langere rekentijd.
- **Treinroostering voor netwerken met tijdsafhankelijke en OD-afhankelijke passagiersaantallen.** Voor treinroostering van stedelijke spoornetwerken hebben we een gebeurtenis-gestuurd model ontwikkeld, waarin de tijdsafhankelijke en OD-afhankelijke passagiersaantallen, het splitsen van de passagiersstromen en het gedrag van passagiers bij overstappen op stations zijn meegenomen. Het treinroosteringprobleem is een niet-lineair, niet-convex probleem met reële variabelen, dat kan worden opgelost met gradiëntvrije niet-lineaire programmeringsmethodes (zoals *pattern search* methodes), gradiënt-gebaseerde niet-lineaire programmeringsmethodes (zoals sequentieel kwadratisch programmeren (SQP)), genetische algoritmes of een MILP-aanpak. Wij hebben een SQP-methode en een genetisch algoritme toegepast op een casus van het treinroosteringprobleem. De resultaten laten zien dat de SQP-methode zorgt voor een beter compromis tussen regelprestatie en rekencomplexiteit dan het genetische algoritme.

Summary

Optimal Trajectory Planning and Train Scheduling for Railway Systems

Safe, fast, punctual, energy-efficient, and comfortable rail traffic systems are important for rail operators, passengers, and the environment. Due to the increasing energy prices and environmental concerns, the reduction of energy consumption has become one of the key objectives for railway systems. On the other hand, with the increase of passenger demands in urban rail transit systems of large cities, it is important to transport passengers safely and efficiently. The main focus of the research presented in this thesis is to determine and develop mathematical models and solution approaches to shorten the travel time of passengers and to reduce energy consumption in railway systems. More specifically, the travel time of passengers has been considered in train scheduling, where passenger demands of urban rail transit systems are included. The energy efficiency has been taken into account both in the train scheduling and in the operation of trains.

The main topics investigated in the thesis can be summarized as:

- **Optimal trajectory planning for a single train.** We have considered the optimal trajectory planning problem for a single train under various operational constraints, which include the varying line resistance, variable speed restrictions, and the varying maximum traction force. The objective function of the optimization problem is a trade-off between the energy consumption and the riding comfort. We have proposed two approaches to solve this optimal control problem, namely a mixed-integer linear programming (MILP) approach and the pseudospectral method. Simulation results comparing the MILP approach, the pseudospectral method, and a discrete dynamic programming approach have shown that the pseudospectral method results in the best control performance, but that if the required computation time is also taken into consideration, the MILP approach yields the best overall performance.
- **Optimal trajectory planning for multiple trains.** The optimal trajectory planning problem for multiple trains under fixed block signaling systems and moving block signaling systems has been investigated. Four solution approaches have been proposed: the greedy MILP approach, the simultaneous MILP approach, the greedy pseudospectral approach, the simultaneous pseudospectral method. Simulation results have shown that compared to the greedy approach, the simultaneous approach yields a better control performance but requires a higher computation time. In addi-

tion, the end time violations of the MILP approach are slightly larger than those of the pseudospectral method, but the computation time of the MILP approach is one to two orders of magnitude smaller than that of the pseudospectral method.

- **Train scheduling for a single line based on OD-independent passenger demands.** The train scheduling problem for an urban rail transit line has been considered with the aim of minimizing the total travel time of passengers and the energy consumption of the operation of trains. The departure times, running times, and dwell times of the trains have been optimized based on origin-destination-independent (OD-independent) passenger demands. We have proposed a new iterative convex programming (ICP) approach to solve this train scheduling problem. The performance of the ICP approach has been compared with other alternative approaches, such as nonlinear programming approaches, a mixed integer nonlinear programming (MINLP) approach, and an MILP approach. The ICP approach has been shown, via a case study, to provide the best trade-off between performance and computational complexity for the train scheduling problem.
- **Train scheduling for a single line based on OD-dependent passenger demands.** We have adopted a stop-skipping strategy to reduce the passenger travel time and the energy consumption further based on origin-destination dependent (OD-dependent) passenger demands in an urban rail transit line. The train scheduling problem with stop-skipping results in a mixed integer nonlinear programming problem and we have proposed a bi-level optimization approach and an efficient bi-level optimization approach to solve this problem. Simulation results show that the stop-skipping strategy outperforms the all-stop strategy. Moreover, the bi-level approach yields a better control performance than the efficient bi-level approach but at a cost of a higher computation time.
- **Train scheduling for networks with time-varying OD-dependent passenger demands.** For the train scheduling for urban rail transit networks, we have developed an event-driven model, where the time varying OD-dependent passenger demands, the splitting of passenger flows, and the passenger transfer behavior at transfer stations is included. The resulting train scheduling problem is a real-valued nonlinear nonconvex problem, which can be solved by gradient-free nonlinear programming approaches (e.g., pattern search), gradient-based nonlinear programming approaches (e.g., sequential quadratic programming (SQP)), genetic algorithms, or an MILP approach. We have applied an SQP method and a genetic algorithm to solve the train scheduling problem for a case study, the results of which have shown that the SQP method provides a better trade-off between control performance and computational complexity with respect to the genetic algorithm.

Curriculum vitae

Yihui Wang was born on November 22, 1987 in Huize, Yunnan, China. In 2007, she obtained the B.Sc. degree on Automation from the School of Electronic and Information Engineering, Beijing Jiaotong University.

In September 2007, Yihui became a Ph.D. student supervised by Prof. Bin Ning in the State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University. Her research had been focus on the safety, reliability, and maintainability of communication protocols for communication-based train control systems and the Chinese train control system level 3.

In October 2010, Yihui was sponsored by Chinese Scholarship Council (CSC) to visit the Delft Center for Systems and Control, Delft University of Technology, the Netherlands. Later, she was employed as a Ph.D. candidate at the Delft Center for Systems and Control under the supervision of Prof. dr. ir. Bart De Schutter and Dr. ir. Ton van den Boom. In her Ph.D. project, Yihui worked mostly on optimization algorithms for the trajectory planning and train scheduling of urban rail transit systems. In the mean time she also obtained the certificate of the Dutch Institute of Systems and Control (DISC) graduate school. Her main interests are model predictive control, optimization, discrete event systems, and railway operation.