



Integration of Disruptions in Railway Planning

Florian Fuchs

Email: flfuchs@student.ethz.ch

Student Nr.: 14-658-157

Supervisors:

Dr. Nikola Bešinović

Prof. Dr. Rob M. P. Goverde

Prof. Dr. Francesco Corman

DRT-Lab TU Delft

IVT ETH Zurich

Master Thesis

March 2022

Integration of Disruptions in Railway Planning

Florian Fuchs
Rebstockweg 10
CH-8049 Zurich
fuchs.florian@outlook.com

March 2022

Abstract

Railways play a crucial role in transport. However, unforeseen events in the form of disruptions drastically affect performance. A literature review reveals that research already addresses detecting and dealing with disruptions. However, no approach considers these tasks within the planning process of a timetable. This thesis bridges this gap by integrating increasing resilience in the process. Hence, we propose an iterative approach for creating timetables from scratch while identifying and addressing the critical disruption. We use a demand-centred performance metric based on generalised travel time, providing a unified objective. Aiming for a holistic approach, we include passenger routing, track assignment and rolling stock circulation. We introduce the concept of the [Train Slot Sequence \(TSS\)](#) with which we reduce complexity by restricting trains to time windows. We apply logical-Bender's decomposition using [TSS](#), splitting the task into a [Mixed Integer Program \(MIP\)](#) and a [Boolean Satisfiability Problem \(SAT\)](#). Furthermore, we cannot rely on predefined scenarios given the context. Therefore, we present a primal-dual algorithm identifying critical disruptions while considering all responses.

We conduct a case study with real-life data of [RhB](#), a Swiss railway company. Our experiments indicate that solving the timetabling model directly with a commercial solver is only appropriate for small instances. However, we quickly obtain high-quality solutions with a fix-and-divide heuristic. Furthermore, within the case study, our propositions increase resilience. For example, in an instance based on artificially generated demand, performance increases from 35.9% to 58.4%, improving the remaining performance by a factor of 1.63. Another valuable insight is that our procedure to identify and assess critical disruption also works for simultaneously occurring disruptions. We apply the approach to nine instances that vary in demand and trains. The results reveal that the proposition performs as intended. However, solutions differ in resilience against critical disruptions and regular performance. Thus, determining the appropriate solution can require a trade-off. Nevertheless, a practitioner can select suitable solutions since we provide all created results.

Keywords

Railway Planning, Timetabling, Resilience, Disruption

Suggested Citation

Fuchs F. (2022), Integration of Disruptions in Railway Planning, Master Thesis, DRTL & IVT, TU Delft & ETH Zurich, Zurich

Integration of Disruptions in Railway Planning

Florian Fuchs
Rebstockweg 10
CH-8049 Zurich
fuchs.florian@outlook.com

Zusammenfassung

Eisenbahnen spielen eine entscheidende Rolle im Verkehrswesen. Unvorhergesehene Ereignisse in Form von Störungen beeinträchtigen jedoch die Performance. Ein Betrachten der Literatur zeigt, dass sich die Forschung bereits mit dem Erkennen von und dem Umgang mit Störungen befasst. Es gibt jedoch keinen Ansatz, der diese Aufgaben als Teil des Planungsprozesses berücksichtigt. Die vorliegende Arbeit schliesst diese Lücke, indem sie die Steigerung der Resilienz in den Planungsprozess integriert. Um Fahrpläne von Grund auf zu erstellen und gleichzeitig kritische Störungen anzugehen, schlagen wir einen iterativen Ansatz vor. Es wird ein nachfragebasierter Leistungsindex, welcher auf der generalisierten Reisezeit aufbaut und ein einheitliches Ziel darstellt, vorgeschlagen. Vorgeschlagen wird ein ganzheitlicher Ansatz, welcher die Fahrgastlenkung, die Gleiszuweisung und den Fahrzeugumlauf einbezieht. Um die Komplexität zu reduzieren, wird das Konzept der [Train Slot Sequence \(TSS\)](#) eingeführt, welches die Züge innerhalb eines Zeitfensters beschränkt. Mit Hilfe von [TSS](#) kann die logische Bender-Dekomposition angewendet werden, so dass die Problemstellung in ein [Mixed Integer Program \(MIP\)](#) und ein [Boolean Satisfiability Problem \(SAT\)](#) aufteilbar ist. Aus dem Kontext ergibt sich weiter, dass keine vordefinierte Störungsszenarien verfügbar sind. Daher wird ein *primal-dual*-Algorithmus vorgestellt, der kritische Störungen identifiziert und verfügbare Gegenmassnahmen berücksichtigt.

Es wird eine Fallstudie mit realen Daten von der [RhB](#), einer Schweizer Eisenbahngesellschaft, durchgeführt. Experimente zeigen, dass das direkte Lösen des Fahrplanmodells mit einem kommerziellen Solver nur für kleine Instanzen geeignet ist. Mit einer Fix-and-Dive-Heuristik können jedoch schnell hochwertige Lösungen erzeugt werden. Außerdem erhöht der Ansatz innerhalb der Fallstudie die Resilienz. In einer Instanz, die auf künstlich erzeugter Nachfrage basiert, steigt der Leistungsindex beispielsweise von 35,9 % auf 58,4 %, was die verbleibende Leistung um den Faktor 1,63 verbessert. Eine weitere wertvolle Erkenntnis ist, dass das Verfahren zur Identifizierung und Bewertung kritischer Störungen auch bei gleichzeitig auftretenden Störungen funktioniert. Der Ansatz wird auf 9 Instanzen angewendet, die sich in Nachfrage und Anzahl Zügen unterscheiden. Die Ergebnisse zeigen, dass der Vorschlag wie beabsichtigt funktioniert. Allerdings unterscheiden sich die Lösungen im Verhältnis von Resilienz gegenüber kritischen Störungen und der regulären Leistung. Daher kann die Bestimmung der geeigneten Lösung einen Kompromiss erfordern. Allerdings kann ein Anwender geeignete Lösungen auswählen und vergleichen, da alle Zwischenergebnisse ausgegeben werden.

Schlagworte

Railway Planning, Timetabling, Resilience, Disruption

Zitierungsvorschlag

Fuchs F. (2022), Integration of Disruptions in Railway Planning, Master Thesis, DRTL & IVT, TU Delft & ETH Zurich, Zurich

Preface

I would like to express my most profound appreciation to my supervisory committee of Professor Dr Rob M.P. Goverde from TU Delft, Dr Nikola Bešinović from TU Delft and Professor Dr Francesco Corman from ETH Zurich. I feel deeply honoured that I am given a chance to conduct my thesis under the guidance of such an outstanding and world-class committee. I desire to emphasize how much I appreciate this opportunity to visit TU-Delft; many thanks to all three committee members for making that possible.

Besides, I am incredibly grateful to [RhB](#) and particularly Martin Moser as my contact, providing me with data for the case study. I truly appreciate the outstanding flexibility and speed I was provided with support and insights. Another vital component in this endeavour is the financial support that I was given by the IDEA League. I am very grateful for this contribution, as it was one of the essential elements that rendered this exchange possible.

I would also like to extend my gratitude to all the members of the DRT-Lab at TU Delft. I wish to thank you for treating me as one of your peers from day one on and for your wonderful fellowship throughout the thesis.

Another person I want to give special thanks to is Dr Matthias Hellwig of SMA and Partner AG. I genuinely appreciate your insightful comments on my work, benefiting from your wide range of experiences. Particularly helpful to me during this time was Dominique Luder, providing me with vital comments on how to improve the report but also with essential emotional support. Eventually, I'm particularly grateful to my Partner, Nina Flükiger, as you were definitely the key to success. Your extensive support is what made the difference.



Eigenständigkeitserklärung

Die unterzeichnete Eigenständigkeitserklärung ist Bestandteil jeder während des Studiums verfassten Semester-, Bachelor- und Master-Arbeit oder anderen Abschlussarbeit (auch der jeweils elektronischen Version).

Die Dozentinnen und Dozenten können auch für andere bei ihnen verfasste schriftliche Arbeiten eine Eigenständigkeitserklärung verlangen.

Ich bestätige, die vorliegende Arbeit selbständig und in eigenen Worten verfasst zu haben. Davon ausgenommen sind sprachliche und inhaltliche Korrekturvorschläge durch die Betreuer und Betreuerinnen der Arbeit.

Titel der Arbeit (in Druckschrift):

Integration of Disruptions in Railway Planning

Verfasst von (in Druckschrift):

Bei Gruppenarbeiten sind die Namen aller Verfasserinnen und Verfasser erforderlich.

Name(n):

Fuchs

Vorname(n):

Florian

Ich bestätige mit meiner Unterschrift:

- Ich habe keine im Merkblatt [„Zitier-Knigge“](#) beschriebene Form des Plagiats begangen.
- Ich habe alle Methoden, Daten und Arbeitsabläufe wahrheitsgetreu dokumentiert.
- Ich habe keine Daten manipuliert.
- Ich habe alle Personen erwähnt, welche die Arbeit wesentlich unterstützt haben.

Ich nehme zur Kenntnis, dass die Arbeit mit elektronischen Hilfsmitteln auf Plagiate überprüft werden kann.

Ort, Datum

Delft, 03.03.2022

Unterschrift(en)

F. Fuchs

Bei Gruppenarbeiten sind die Namen aller Verfasserinnen und Verfasser erforderlich. Durch die Unterschriften bürgen sie gemeinsam für den gesamten Inhalt dieser schriftlichen Arbeit.

Contents

List of Tables	vii
List of Figures	viii
1 Introduction	1
1.1 Motivation and Problem Statement	1
1.2 Our Contributions	2
1.3 Defining the Research Questions	3
1.4 Outline of the Thesis	3
2 Background	4
2.1 The Planning Process in Railway Systems	4
2.1.1 Line Planning	5
2.1.2 Timetabling	5
2.1.3 Vehicle Scheduling	7
2.1.4 Advantages of (partially) Integrated Planning for Railways	7
2.2 Addressing Resilience and Disruptions in Railway Systems	8
2.2.1 Characteristics of Resilience and Disruptions in Railway Systems	9
2.2.2 Disruption Management and Resilience in Railway Systems	10
2.2.3 Network-Wide Approaches to Assess and Enhance Resilience	11
2.2.4 Scenario-Based Approaches to Assess and Enhance Resilience	12
2.2.5 Network Interdiction	14
2.3 Identifying the Research Gap	15
3 Methodology	16
3.1 Problem Description	16
3.1.1 Narrowing the Scope	16
3.1.2 Considered Perspectives	17
3.1.3 Defining the Performance Metric	17
3.1.4 Proposing the TSS to encompass TOC Requirements	19
3.2 Overall Approach	21
3.2.1 Fundamental Data Structures	22
3.2.2 Defining the Train Slot Sequence (TSS)	23
3.2.3 Generating the <i>SSN</i>	25
3.2.4 Generating the <i>EAN</i>	26
3.3 Introducing the Optimisation Module (OM)	29
3.3.1 The Master Slot Sequence Selection Problem (M-SSSP)	29
3.3.2 Addressing a Critical Disruption in the M-SSSP	31
3.3.3 The Sub Slot Sequence Selection Problem (S-SSSP)	32
3.3.4 Encoding the S-SSSP with Boolean Satisfiability Problem (SAT)	34
3.3.5 Connecting the M-SSSP and S-SSSP in the Optimisation Module (OM)	35
3.3.6 A Fix-and-Dive-Heuristic for the Slot Sequence Selection Problem (SSSP)	36
3.3.7 Shrinking the size of the M-SSSP	36

3.4	Introducing the Assessment Module (AM)	38
3.4.1	Modifying the SSSP towards Primal Slot Sequence Selection Problem (Primal-SSSP)	38
3.4.2	Finding the most critical Disruptions with the SSSP	40
3.4.3	Primal-Dual Algorithm for Finding the most Critical Disruption	41
4	Case Study	43
4.1	Implementation of the iterative Approach	43
4.1.1	Heuristic Architecture in the OM	43
4.1.2	Implementation Details	44
4.2	Case Study Data	45
4.2.1	Infrastructure Network and Timetabling Parameters	45
4.2.2	Line Plans for the Case Study	46
4.2.3	Train Slot Sequence (TSS) Generation	48
4.2.4	Rolling Stock	49
4.2.5	Defining the Parameters for Generalised Travel Time	49
4.2.6	Demand Scenarios	50
4.3	An Overview on the Computational Experiments	52
4.4	Computational Experiments for Individual Assessments	52
4.4.1	Using the OM to Optimise Timetables	52
4.4.2	The effect of Shrinking the Instance Sizes in the M-SSSP	55
4.4.3	Using the AM to Detect $\mathcal{H}_{\text{critical}}$: One Disrupted Link	56
4.4.4	Using the OM to increase Resilience against one Critical Link	58
4.4.5	Using the OM to increase Resilience against two Simultaneously Disrupted Links	60
4.5	Computational Experiments to Assess the Iterative Methodology	63
4.5.1	Assessment with <code>passenger</code> Trains and <code>no-conflict</code> Detection	63
4.5.2	Assessment with <code>passenger</code> Trains	65
4.5.3	Assessment with <code>passenger</code> and <code>cargo</code> Trains	66
4.5.4	Computation Time per Iteration	68
5	Conclusion and Future Work	70
5.1	Conclusion	70
5.2	Future Work	73
5.2.1	Future Work on the OM	73
5.2.2	Future Work on the AM	74
5.2.3	Future Work on the Iterative Approach using the OM and AM	74
6	References	76
A	Supporting Material for the Case Study	84
A.1	Subset of Stations for Specific Purposes	84
A.2	Considered SAT Solvers	85
A.3	Interactive Figures as <i>html</i> -files	85

List of Tables

1	Schemes to jointly address multiple planning stages in railway planning.	8
2	Speed in m/s per train category to approximate \hat{w}_a	46
3	Bounds per activity in the <i>EAN</i>	46
4	Input parameters to calculate the TSS per train category.	49
5	Available rolling stock, based on RhB (2021a).	49
6	Results for solving the SSSP with passenger trains and no-conflict	53
7	Results for solving the SSSP with passenger trains.	53
8	Results for solving the SSSP with passenger and cargo trains.	54
9	Number of required variables in the M-SSSP in the OM.	55
10	Average computation time in seconds for one iteration.	69

List of Figures

1	Disruptions per year in absolute numbers and hours for the Dutch railway network (note the effect of Covid-19 in 2020/2021).	1
2	Planning stages and phases aligned with the time frame.	4
3	Bathtub model illustrating system performance during a disruption.	9
4	The performance metric visualised with the resilience bathtub curve.	18
5	The TSSs to couple the line plan, timetable and infrastructure.	20
6	An overview on the Approach.	21
7	An illustration of how S_κ captures the route of κ .	24
8	Example SSN where based on the example line in Fig. 5.	26
9	The first three steps of the EAN creation for one $\mathcal{Z}_{\kappa,\pi}$.	27
10	An excerpt of an EAN for two $\mathcal{Z}_{\kappa,\pi}$ with headways.	28
11	Example EAN and TFN for two $\mathcal{Z}_{\kappa,\pi} \in \mathcal{Y}_{\text{active}}$.	32
12	Primal-Dual algorithm for finding the most critical disruption.	42
13	The heuristic procedure implemented in the OM.	43
14	Macroscopic view on the infrastructure network.	45
15	Network of the Rhaetian Railway (RhB) (RhB, 2021b) with an excerpt of the infrastructure.	47
16	Bus lines available as response measures.	48
17	$OD_{\text{artificial}}$ visualised with a macroscopic view of the infrastructure.	50
18	OD_{subset} visualised with a macroscopic view of the infrastructure.	51
19	Example timetable as 3D-time-space diagram, where z is time in seconds and x and y are coordinates in space (available as <i>html</i> (Appendix A.3)).	55
20	Results of the <code>primal-dual</code> algorithm (available as <i>html</i> (Appendix A.3)).	57
21	Location of $\mathcal{H}_{\text{critical}}$ depending on <code>no-buses</code> / <code>five-buses</code> / <code>ten-buses</code> .	58
22	Bathtub curves for $\mathcal{H}_{\text{critical}}$ (available as <i>html</i> (Appendix A.3)).	59
23	Effect of enforcing redundant connections on passenger flow	60
24	Bathtub curves for $\mathcal{H}_{\text{critical}}$ with $K = \{2\}$ (available as <i>html</i> (Appendix A.3)).	61
25	Passenger flow and locations of $h \in \mathcal{H}$ for two simultaneous disruptions.	62
26	Performance P during 10 iterations with <code>no-conflict</code> and <code>five-buses</code> (available as <i>html</i> (Appendix A.3)).	64
27	Performance P during 10 iterations with <code>passenger</code> trains and <code>five-buses</code> (available as <i>html</i> (Appendix A.3)).	65
28	Performance P during 10 iterations with <code>passenger</code> , <code>cargo</code> and <code>auto-trains</code> (available as <i>html</i> (Appendix A.3)).	67
29	Box-plots to summarise the computation time per iteration (available as <i>html</i> (Appendix A.3)).	69
30	Stations suitable for transferring in the OM are marked blue.	84
31	Stations suitable for transferring in the AM are marked blue.	84
32	Stations suitable for short turning trains in the OM are marked blue.	85

Acronyms

- EAN* Event Activity Network. v, vii, viii, 22, 23, 26–29, 32, 35, 39, 46
- SSN* Slot Sequence Network. v, viii, 22, 23, 25, 26, 29
- TFN* Train Flow Network. viii, 32–35
- AM Assessment Module. vi, viii, 21, 38, 41, 47, 52, 56, 58, 63, 68, 73, 74, 84
- CGN Change & Go Network. 5
- CNF Conjunctive Normal Form. 34
- DAG Directed Acyclic Graph. 27
- Dual-SSSP Dual Slot Sequence Selection Problem. 38, 40–42, 56–58
- IM Infrastructure Manager. 17, 72
- LPP Line Planning Problem. 5–8
- M-SSSP Master Slot Sequence Selection Problem. v–vii, 29–31, 35, 36, 38–40, 44, 52, 53, 55, 59, 66, 85
- MIP Mixed Integer Program. i, ii, 5–8, 13, 14, 34, 36, 37, 44, 53–56, 68, 70
- OM Optimisation Module. v–viii, 21, 23, 29, 35, 38, 43, 47, 52, 54–56, 58, 60, 62, 63, 65, 68, 73, 74, 84, 85
- PESP Periodic Event Scheduling Problem. 6, 25, 26, 34, 46
- PTN Public Transport Network. 5
- Primal-SSSP Primal Slot Sequence Selection Problem. vi, 38–42, 47, 56–58, 61, 74
- RQ (Main) Research Question. 3, 70
- RhB Rhaetian Railway. i–iii, viii, 45–50, 52, 60, 62, 63, 70, 73
- S-SSSP Sub Slot Sequence Selection Problem. v, 29, 30, 32–36, 39, 44, 52–54, 63, 74, 85
- SAT Boolean Satisfiability Problem. i, ii, v, 6, 7, 34, 35, 44, 70, 85
- SI Service Intention. 2, 19, 70, 72
- SQ Sub-Question (of the Research Question). 3, 4, 16, 18, 43, 70–73
- SSSP Slot Sequence Selection Problem. v–vii, 29, 36–38, 40, 43, 52–54, 58, 59, 64, 74, 75
- TOC Train Operating Company. v, 17, 19, 23, 48, 72
- TPE Train Path Envelope. 2, 19, 70, 72
- TSS Train Slot Sequence. i, ii, v–viii, 2, 16, 19–23, 25, 29, 30, 39, 48, 49, 70, 72, 74
- TTP Time Tabling Problem. 6–8, 17, 21–23, 26, 28, 29, 32, 34, 36
- VSP Vehicle Scheduling Problem. 7, 8

1 Introduction

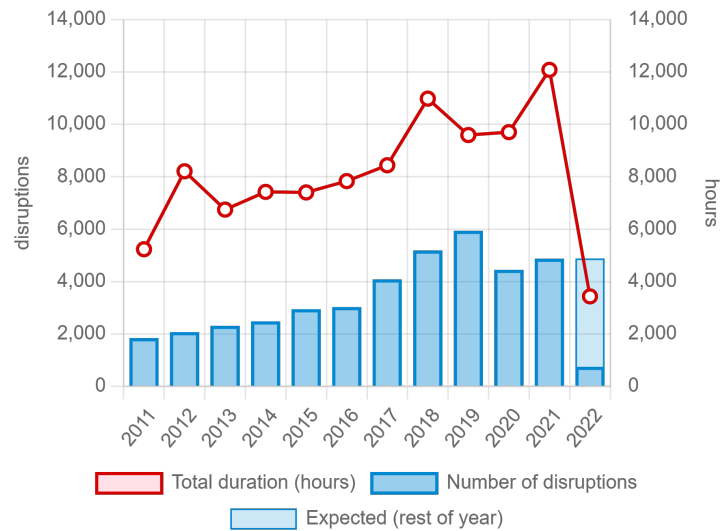
Initially, we briefly introduce the thesis. We begin with stating our motivation and describing the problem in Section 1.1. Next we summarise our contributions in Section 1.2, before defining the research questions in Section 1.3. Finally, we give a brief outline on the structure of the thesis in Section 1.4.

1.1 Motivation and Problem Statement

Railway networks are an essential part of the transport infrastructure today and in the future. Because of social and economic dependencies, the networks must provide high-quality service and operate as reliably as possible. Unexpected blockages of open tracks, station tracks, or complete stations during a specific period represent typical disruptions in railway operations.

While myriad reasons can cause disruptions, the impact is generally identical. The lost capacity due to blocked and unavailable infrastructure affects the performance drastically. Although the severity of the disruption can vary, it is frequently no longer possible to operate the original timetable. Therefore, the strategically or tactically designed timetable has to be adapted on short notice to provide an alternative form of service in such a situation (Bešinović, 2020).

Figure 1: Disruptions per year in absolute numbers and hours for the Dutch railway network (note the effect of Covid-19 in 2020/2021).



Source: [Rijden de Treinen \(2022\)](#)

A common observation with disruptions is their growing number with increasing network utilisation (Bešinović, 2020). Using the Dutch railway network as a representative illustration, Fig. 1 indicates that the number of disruptions has gradually increased until Covid-19 caused a drastic reduction in demand (Bešinović and Szymula, 2021). Given lower demand, fewer services operate, consequently reducing the total number of disruptions.

Thereby, when aiming for high performance, considering disruptions is crucial. Several approaches and strategies to cope with known disruption exist (Ghaemi *et al.*, 2017b; Zhu and Goverde, 2021, 2020a,b; Bešinović *et al.*, 2020). Besides, it is possible to assess resilience with network-wide approaches to identify critical infrastructure (Szymula and Bešinović, 2020; Bababeik *et al.*, 2017). However, less attention has been directed towards the integration of dealing with disruptions in earlier planning phases (Bešinović, 2020).

This thesis aims to address this gap with a methodology that integrates disruptions in the early planning stages. The proposed methodology aims to increase the system's resilience against the most critical disruption (i.e. the one with the highest impact on performance). Consequently, the resilience of a timetable can be increased when it is developed.

1.2 Our Contributions

Based on the previously defined research gaps, we now provide a brief overview of our approach before summarising the key contributions.

Essentially, railway planning and enhancing resilience share a common objective in optimally utilising and allocating (available) infrastructure and resources. While the former seeks to provide the best possible solution (e.g., while exploiting all infrastructure (Fuchs *et al.*, 2021)), the latter focuses on reducing the impact of disruptions (e.g., delays and cancelled trains (Zhu and Goverde, 2021)). Consequently, combining both requires a holistic perspective and a detailed incorporation of the dynamics of a railway system (Bešinović, 2020). We, therefore, include the available infrastructure on a mesoscopic level (de Fabris *et al.*, 2014; Wüst *et al.*, 2019b) and consider the task of routing trains jointly during timetabling similar to Fuchs *et al.* (2021) to exploit the available infrastructure. Furthermore, to follow and enhance the state-of-the-art approaches, our method should be capable of (re-)routing passengers (Zhu and Goverde, 2020b) and accounting for vehicle rotations and capacity (Veelenturf *et al.*, 2017). Eventually, integrating alternative modes such as buses as responses towards disruptions is also desirable (Borecka and Bešinović, 2021; Jin *et al.*, 2014).

Interestingly, many timetabling models to address all or a subset of these requirements already exist (Schiewe, 2020; Polinder *et al.*, 2021; Fuchs *et al.*, 2021). However, including these requirements jointly is leading to a challenging problem, which has not been solved satisfactorily (Schiewe, 2020). Therefore, we propose addressing the problem with a novel approach that defines a sequence of time slots for each train, the **Train Slot Sequence (TSS)**. The concept of **TSS** is similar to the **Train Path Envelope (TPE)** (Albrecht *et al.*, 2013; Wang *et al.*, 2020) and the **Service Intention (SI)** (Caimi *et al.*, 2011b,a; Wüst *et al.*, 2019a), both successfully applied to railway timetabling.

We use these **TSSs** to define departure, arrival and intermediate pass-through times as time windows instead of exact timestamps. Given **TSS**, we can reduce the complexity of the integrated timetabling problem such that we can solve real-life instances. Additionally, as we are considering

resilience during the early stages of the planning processes, we can not rely on predefined disruption scenarios (Szymula and Bešinović, 2020). Thus, we extend our model to identify and accurately assess the critical disruption (i.e., the one with the highest impact) without a given input. Subsequently, our methodology integrates resilience into the railway planning process and yields a holistic and accurate result. We can summarise our contributions as follows:

- Integrating resilience into the early stages of the planning process
- Jointly identifying the critical disruption and the best response
- Breaking down the complexity of integrated timetabling
- Applying the proposition to a real-life case study

Subsequently, we proceed to define the research questions.

1.3 Defining the Research Questions

Based on the stated research gap and the contributions, we may formulate the (Main) Research Question (RQ) and the Sub-Question (of the Research Question) (SQ) form the thesis's core:

(Main) Research Question (RQ) Which optimisation approach is suitable to assess and enhance the resilience of a railway network during the strategical and tactical planning phases?

SQ 1 What state-of-the-art metrics/measures are suitable to assess the resilience of a railway timetable during timetabling?

SQ 2 How can we characterise disruptions such that we can assess them during the timetabling process?

SQ 3 How to mathematically model the requirements and resources of the different stakeholders to include them as constraints?

SQ 4 What are the benefits when using the developed optimisation approach to enhance the resilience of an existing railway system?

The RQ and SQs guide the thesis in the following. The RQ provides the main motivation of the thesis and we will develop the required methodology in Section 3, before applying it in the case study Section 4. Furthermore, we will address the SQ 1, 2 and 3 in Section 3 based on the literature reviewed in Section 2. The computational experiments that are part of the case study in Section 4 provide us with insights to address the SQ 4. Furthermore, these questions allow us to draw a conclusion and identify future work in Section 5.

1.4 Outline of the Thesis

The remainder of this thesis is structured as follows; Section 2 provides the necessary background on the railway planning processes and resilience in railways, to then identify the research gap. Subsequently, we propose our methodology in Section 3, which we then apply and assess in the case study in Section 4. Eventually, we draw a conclusion in Section 5, where we address the research questions and also outline future research possibilities.

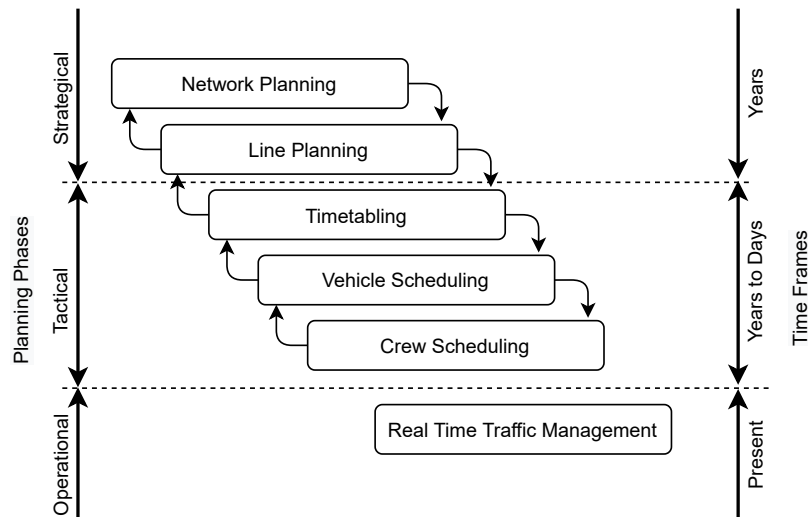
2 Background

This section provides technical background and an overview of the-state-of-the-art. We focus on investigating SQ 1, SQ 2 and SQ 3, since these SQ require literature background. In Section 2.1, we provide information on the general planning process in railway systems. Complementary, in Section 2.2, a review on resilience assessment and disruption management in railway systems follows. Eventually, we identify the research gap in Section 2.3.

2.1 The Planning Process in Railway Systems

Planning a railway system is a challenging and interconnected process, so the planning process is usually divided into a sequence of stages (Lusby *et al.*, 2018). The stages in Fig. 2 represent the general planning stages of railway systems according to Liebchen and Möhring (2004) and Lusby *et al.* (2018). Following the notion of Lusby *et al.* (2018), the planning stages span over the three phases of strategical, tactical, and operational planning.

Figure 2: Planning stages and phases aligned with the time frame.



Source: Based on Liebchen and Möhring (2004) and Lusby *et al.* (2018)

While strategical planning covers network and line planning, the tactical phase encompasses timetabling and vehicle and crew scheduling. Finally, real-time traffic management is contained by the operational planning phase and is performed at present (Lusby *et al.*, 2018). Since any modification of the available infrastructure is outside this thesis's scope, we limit the considered stages to the stage of line planning and onward. Furthermore, as we aim to enhance the resilience of a railway system during its planning phase, we do not consider the stages of crew scheduling and real-time management tasks. Consequently, we briefly introduce line planning (Section 2.1.1), timetabling (Section 2.1.2), and vehicle scheduling (Section 2.1.3) in the following.

2.1.1 Line Planning

The task of line planning covers the stage of determining a line plan by solving the [Line Planning Problem \(LPP\)](#). A line represents a group of train services that serve a defined sequence of stations. In general, a solution to the [LPP](#) specifies which lines should be operated at which frequency. Consequently, a line plan generally relies on a periodic pattern, such that the frequencies define the number of train services per line and period. Based on the frequency and assigned vehicle, it is possible to assign a passenger capacity to the chosen lines. Predominantly, passengers may only be routed via a line if sufficient capacity is available. A wide variety of approaches to tackle the [LPP](#) exist. We refer to [Schöbel \(2012\)](#) for a general overview.

Most [LPP](#) seek to minimize the passenger discomfort by finding a line plan which minimizes the total travel time. [Schöbel and Scholl \(2006\)](#) use the [Public Transport Network \(PTN\)](#) to route passengers along activated lines in a [Mixed Integer Program \(MIP\)](#). However, as the [PTN](#) only implicitly accounts for dwell/wait and transfer times, [Schmidt and Schöbel \(2015a\)](#) propose the [Change & Go Network \(CGN\)](#), which extends the [PTN](#) with dwell/wait and transfer arcs.

Based on the [CGN](#) [Goerigk and Schmidt \(2017\)](#) propose a [LPP](#)-model that minimizes travel times while demanding that all passengers travel via the shortest available path. However, the resulting [LPP](#) is a bilevel-MIP which is challenging to solve. Consequently, [Goerigk and Schmidt \(2017\)](#) propose a genetic algorithm to efficiently obtain solutions of reasonable quality.

Almost all proposed models start from a pool of candidate lines of which a subset forms the resulting line plan. Naturally, the pool of candidates affects the solution, as it only contains lines that are part of the pool. [Gattermann et al. \(2017\)](#) provide methods to generate candidate line pools with algorithmic methods. Besides, [Borndörfer et al. \(2007\)](#) formulate a column generation approach that does not provide integer frequencies but also alleviates the need for a line pool because lines are defined during the column generation process.

Note that it is also possible to consider vehicles during line planning, as demonstrated by [Goossens et al. \(2006\)](#); The formulated [LPP](#) considers the available vehicles and limits the line plan to not require more vehicles than available. Alternatively, [Friedrich et al. \(2017\)](#) propose to include the cost to operate the line plan in the [LPP](#). Consequently, the resulting line plan satisfies passengers by minimising travel times and operators by respecting the available budget.

2.1.2 Timetabling

Following the scope of this thesis, most of the reviewed literature focuses on periodic railway timetabling. Note, however, that timetabling does not necessarily require periodicity. For an overview of periodic and non-periodic timetabling, we refer to [Caimi et al. \(2017\)](#). Further resources for periodic timetabling are provided by [Liebchen and Möhring \(2004\)](#) and [Peeters \(2003\)](#), with the former providing an extensive overview while the latter yields precise insights.

We will refer to the task of periodic timetabling as the **Time Tabling Problem (TTP)** from here on. A key contribution to the **TTP** is the **Periodic Event Scheduling Problem (PESP)** introduced by [Serafini and Ukovich \(1989\)](#). Although the original formulation did not aim towards railways, many adaptations rely on the **PESP** formulation of [Serafini and Ukovich \(1989\)](#). While extensively researched, finding a timetable is still challenging, as is apparent when considering the proposed solutions. The majority of the approaches rely on **MIP**-formulations, which can be divided into the node potential formulation ([Kroon *et al.*, 2014](#); [Caimi *et al.*, 2011a](#)) and the arc tension formulations ([Peeters, 2003](#); [Liebchen and Peeters, 2009](#); [Herrigel *et al.*, 2018](#)).

[Liebchen and Peeters \(2009\)](#) examine various cycle bases for the cycle-based **MIP** and contrast the performance with the node potential formulation. Their result that the cycle-based formulation exceeds the node potential **MIP** is supported by the findings of [Herrigel *et al.* \(2018\)](#). To increase the performance of the cycle-based **MIP**, [Herrigel *et al.* \(2018\)](#) propose adding services incrementally. Since the already added services are fixed in time, the search space is limited, yielding considerable speedup potential. However, increasing the slack on the already added services is required when conflicts arise. While [Herrigel *et al.* \(2018\)](#) demonstrate the benefits, they also show that the regular cycle-based **MIP** outperforms their approach in some cases.

Contrasting with already introduced approaches, [Zhang *et al.* \(2019\)](#) suggest using a reformulated approach using time-space graphs and Lagrangian multipliers. Another approach for finding solutions to the **TTP** is to encode it as a **Boolean Satisfiability Problem (SAT)** as proposed by [Großmann \(2011\)](#). Using this encoding, [Kümmling *et al.* \(2015\)](#) show that it is possible to find solutions for real-life instances in an acceptable time. Another advantage reported by [Kümmling *et al.* \(2015\)](#) is the capability to find conflicting requirements such that the planners might relax some of them in order to find a feasible solution.

Furthermore, [Großmann \(2016\)](#) extends the **SAT**-encoding such that it is possible to consider more than one available routing option or to insert additional trains in a timetable. With a similar aim, [Fuchs *et al.* \(2021\)](#) extend the **SAT**-encoding such that it is possible to route trains during timetabling, exploiting all available infrastructure. Combining **SAT**, and **MIP**, [Borndörfer *et al.* \(2020\)](#) provide a concurrent approach to find superior solutions on benchmark instances. The concurrent solver also uses the algorithm provided by [Goerigk and Liebchen \(2017\)](#), which allows improving a given feasible solution using the modulo network simplex.

Like the **LPP**, the **TTP** frequently minimises total travel time. Commonly, passengers are assigned to the trains before solving the **TTP** (see [Peeters \(2003\)](#); [Fuchs and Corman \(2019\)](#); [Herrigel *et al.* \(2018\)](#)). However, as [Schmidt and Schöbel \(2015b\)](#) show, the travel times can depend on the found **TTP** solution. Furthermore, [Schmidt and Schöbel \(2015b\)](#) show that considering the routing of passengers during **TTP** can lead to superior solutions. [Borndörfer *et al.* \(2017\)](#) provide further insights and show that pre-assigning passenger routes can lead to infinitely worse solutions, when compared to a joint approach.

[Robenek *et al.* \(2016\)](#) show that passenger routes may also be considered when non-periodic **TTP**-instances are tackled. While all the above formulations to route passengers during timetabling

rely on **MIP** formulations, [Gattermann et al. \(2016\)](#) propose a max-**SAT** encoding for this purpose. [Polinder et al. \(2021\)](#) target the step of strategical timetabling by providing a **MIP** formulation that allows planners to assess the trade-off between total travel time and service regularity. Note that none of the above approaches considers the capacity of the vehicles.

Besides total travel time, robustness against delays is also a conceivable objective for railway timetabling, as shown by [Burggraeve et al. \(2017\)](#), who use a **MIP**-formulation to maximise the minimal buffer time to provide robust solutions to the **TTP**. Similarly, [Bešinović et al. \(2016\)](#) provide robust non-periodic timetables by iterating between micro-and macroscopic timetabling. Another approach aiming to increase robustness reduces the minimum cycle time, as suggested by [Zhang and Nie \(2016\)](#). [Sparing and Goverde \(2017\)](#) expand the approach to handle flexible train orders, running and dwell times, and overtaking locations.

[Caimi et al. \(2011a\)](#) introduce event flexibility to periodic timetabling. As a result, it is possible to generate regular timetables with flexible events. Another approach for introducing flexibility is the proposition of [Kroon et al. \(2014\)](#), leading to flexible connections. These allow selecting connections at stations such that the time for transfers and the number of vehicles can be reduced. [Fuchs and Corman \(2019\)](#) implement these flexible connections and report similar findings.

2.1.3 Vehicle Scheduling

The stage of vehicle scheduling is frequently also referred to as *vehicle rostering*. We introduce the term **Vehicle Scheduling Problem (VSP)** to encompass both. Given a timetable, [Peeters and Kroon \(2008\)](#) propose an approach to minimise the number of circulating rolling stock units. The **Vehicle Scheduling Problem (VSP)** is a subsequent step to **LPP** and **TTP**. Hence the available time horizons are shorter. [Giacco et al. \(2014\)](#) propose an approach capable of scheduling vehicles and integrating the maintenance intervals into the resulting schedule. [Luan et al. \(2017\)](#) aim for a similar result by combining vehicle scheduling with the optimal planning of preventive maintenance slots.

[Lusby et al. \(2018\)](#) highlight that a timetable's robustness also depends on the robustness of the vehicle schedule. [Tréfond et al. \(2017\)](#) provide an approach, which combines optimisation and simulation to provide robust **VSP** solutions.

2.1.4 Advantages of (partially) Integrated Planning for Railways

So far, we considered the stages of **LPP**, **TTP** and **VSP** independently. However, it is not a requirement to address the stages individually; in fact, as shown by [Schiewe \(2020\)](#), considering the stages in an integrated approach enables exploitation of additional optimisation potential, leading to superior solutions. However, since all stages are already challenging when considered independently, a joint consideration imposes an even greater challenge ([Liebchen and Möhring, 2004](#)). Characterised by the trade-off between required effort against solution quality, [Schiewe \(2020\)](#) suggests three schemes summarised by Table 1:

Table 1: Schemes to jointly address multiple planning stages in railway planning.

scheme	potential solution quality	computational effort
integrated	best (optimal)	high / intractable
iterative	average	medium
sequential	poor	low

Source: Summarised from [Schiewe \(2020\)](#)

Generally, any previously introduced **LPP**, **TTP**, and **VSP** approach could be part of a sequential approach. For instance, [Michaelis and Schöbel \(2009\)](#) provide such a sequence of models. By rearranging the order of planning stages, [Michaelis and Schöbel \(2009\)](#) can provide improved solutions for a real-life bus network. [Pätzold et al. \(2017\)](#) also rearrange the prevalent order of planning stages; thereby, the authors can look ahead and limit the set of candidates in the line pool to a set of promising ones, leading to an overall improved solution.

[Schöbel \(2017\)](#) proposes the *Eigenmodel*, which allows the formalisation of iterative and sequential approaches. [Fuchs and Corman \(2019\)](#) implement such an iterative model to iterate between the stages of **LPP**, **TTP** and **VSP**. Similarly, [Burggraefe et al. \(2017\)](#) provide an iterative approach enhancing the **LPP** with robust timetables in the **TTP**. [Fuchs et al. \(2021\)](#) also enhance the **LPP** with an iterative approach which allows detecting and addressing conflicts in the subsequent **TTP** stage. Besides [Fuchs et al. \(2021\)](#) exploit the available infrastructure already during the **LPP** as their **TTP** considers all available train routes. Finally, [Yan and Goverde \(2019\)](#) use an iterative approach to combine **LPP** with **TTP** on strongly heterogeneous railway lines with direct connections.

[Lübbecke et al. \(2018\)](#) propose an **MIP** model that fully integrates **LPP**, **TTP** and **VSP**. However, it is not possible to solve the integrated model directly due to the inherent complexity.

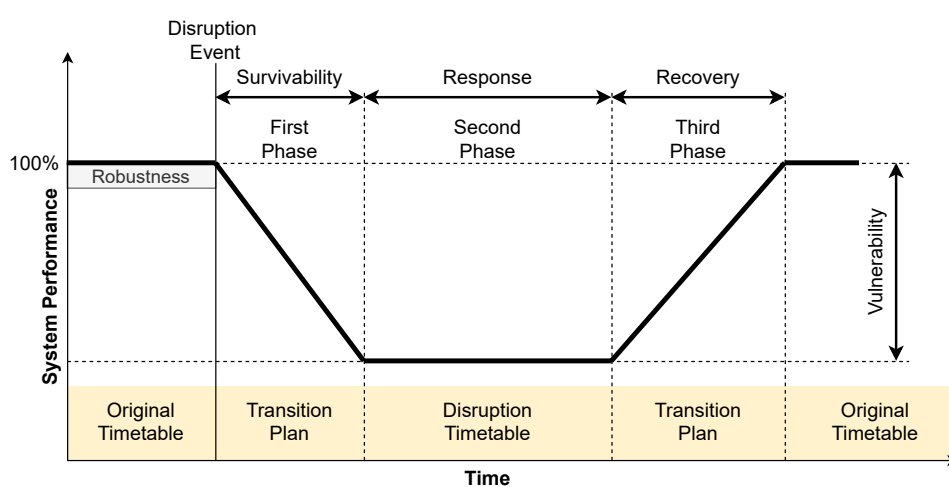
2.2 Addressing Resilience and Disruptions in Railway Systems

Railways rely on specific infrastructure. Besides, as seen in the previous chapter, planning a railway system is a sophisticated process. As a consequence of these two attributes, railways are specifically prone to unexpected events, which is why handling these unexpected events receives considerable attention in research, as we see in Section 2.2.2. However, before we discuss existing approaches to deal with unexpected events, we characterise disruptions and introduce resilience in the context of railways in Section 2.2.1. Furthermore, we investigate existing network-wide approaches in Section 2.2.3 and scenario-based approaches in Section 2.2.4 before a brief introduction of network interdiction in Section 2.2.5 follows.

2.2.1 Characteristics of Resilience and Disruptions in Railway Systems

In general, not every unexpected event represents or unfolds towards a disruption. According to [Bešinović \(2020\)](#), unexpected events during railway operations can be classified into three levels according to their severity. *Disturbances* represent minor variations in operations. *Disruptions* occur due to longer-lasting technical failures and bad weather conditions. *Disasters* symbolise the worst-case induced by extreme events, for example, floods or earthquakes. [Bešinović \(2020\)](#) defines the resilience of a railway transport system as its capability to effectively cope with disruptions and disasters while still providing a satisfactory level of service under ordinary conditions. Furthermore, resilience is a system-wide concept.

Figure 3: Bathtub model illustrating system performance during a disruption.



Source: Based on [Bešinović \(2020\)](#) and [Ghaemi et al. \(2017b\)](#)

As indicated in Fig. 3, resilience encompasses the distinct phases of survivability, response and recovery ([Bešinović, 2020](#)). Furthermore, vulnerability is a concept closely related to resilience:

Survivability describes the transition of the system from the planned system to the disrupted one. Depending on the system and the disruption, this transition results in different ways of degradation.

Response defines the set of actions taken quickly after a disruption to provide the best level of service. Thus, responses can be vital to reduce vulnerability.

Recovery covers the second transition, where the system transitions from the disrupted back to the planned state. Again, the way how the system reverts to the initial performance depends on the system and the characteristics of the disruption.

Vulnerability describes the lost performance between the planned performance and the performance during the disruption. Thus, vulnerability is closely related to resilience.

Mitigation aims to increase resilience by adapting the infrastructure. Ideally, such an adaptation relieves the current most vulnerable elements.

For a more detailed definition of these and additional terms, we refer to [Bešinović \(2020\)](#). Several metrics to evaluate the resilience of transport networks have been proposed ([Zhou et al., 2019](#)). Following [Zhou et al. \(2019\)](#), these are either topological, attributes-based or performance-based metrics. We will further elaborate them in Section 2.2.3.

As mentioned, vulnerability encompasses the reduced performance of the network when the system is in a stable disrupted state, see Fig. 3. Consequently, vulnerability neglects the duration of the transition phases to and from the disrupted state for both survivability and recovery ([Szymula and Bešinović, 2020](#)). [Szymula and Bešinović \(2020\)](#) state, that vulnerability represents one of resilience's (static) elements. Hence, vulnerability is a suitable concept if the scope is primarily on performance during the response phase. However, if one also aims to incorporate the dynamic (non-static) elements of resilience, vulnerability is not a suitable indicator. For further background on different definitions of vulnerability, we refer to [Mattsson and Jenelius \(2015\)](#).

[Ghaemi et al. \(2017b\)](#) also suggest using the bathtub model when dealing with railway's disruptions and divides a disruption into the three phases given in Fig. 3. When combining the two suggested models of [Ghaemi et al. \(2017b\)](#) and [Bešinović \(2020\)](#), we see that the first phase covers survivability, while the second phase represents the response, and the third phase is the recovery. Moreover, as depicted in Fig. 3, [Ghaemi et al. \(2017b\)](#) also suggest which form of the timetable is required during which phase.

2.2.2 Disruption Management and Resilience in Railway Systems

Based on the definitions of resilience and disruptions, we can identify how disruptions and resilience are addressed in current research. In general, two research streams, *Disruption Management* ([Ghaemi et al., 2017b](#)) and *Enhancement of Resilience* ([Zhou et al., 2019](#)) can be identified. In the following, we establish the link between these two.

Following [Zhou et al. \(2019\)](#), when *Enhancement of Resilience* is the aim, the considered time scope as described in the planning process earlier is frequently strategic. On the contrary, *Disruption Management* generally describes the act of focussing on a particular unexpected event occurring at present ([Dekker et al., 2021](#)). Thus, the emphasis is primarily on operative decision-making ([Lusby et al., 2018](#)).

However, strictly separating *Enhancement of Resilience* and *Disruption Management* and labelling the former as proactive and the latter as reactive is imprecise. As demonstrated by [Szymula and Bešinović \(2020\)](#), assessing vulnerability requires an appropriate consideration of response measures, as otherwise, vulnerability is overestimated. Besides, resilience can also be enhanced when scenario specific studies part of *Disruption Management* are conducted ([Bešinović, 2020](#)). Furthermore, *Disruption Management* also contains a proactive component in the form of contingency planning. A contingency plan is a predetermined procedure guiding the response to a certain disruption ([Dekker et al., 2021](#)), thus *Disruption Management* enhances the resilience through an increased preparedness of the system.

Given that *Enhancement of Resilience* and *Disruption Management* are closely related, we no longer distinguish them by their name. Instead, we consider both fields jointly. Thus, we organise the remainder of this chapter by distinguishing between *Network-wide approaches* in 2.2.3 and *Scenario based-approaches* in 2.2.4. Finally, we briefly introduce the topic of network interdiction in Section 2.2.5.

2.2.3 Network-Wide Approaches to Assess and Enhance Resilience

In their review, Zhou *et al.* (2019) provide a detailed overview of the past research of resilience in transport networks. The review synthesizes metrics and mathematical models to measure resilience as well as strategies to enhance resilience. As a conclusion of their review, Zhou *et al.* (2019) emphasize several future research directions. Furthermore, Zhou *et al.* (2019) provide a comprehensive overview of conceivable metrics which are capable of capturing the system's performance. These metrics can be divided in *Topological Metrics*, *Attributes-Based Metrics* and *Performance-Based Metrics*. *Topological Metrics* consider the network structure and can be calculated quickly and efficiently, but their expressive power is generally limited. *Attributes-Based Metrics* yield more detailed insight but often fail to capture the dynamics of the system. *Performance-Based Metrics* reveal the most information but also require the most effort. In their review Zhou *et al.* (2019) state that performance-based metrics are perceived to be superior to attribute-based metrics in capturing the resilience of a transportation system, and both excel topological metrics because these do not account for flows in the network.

Bešinović (2020) provides another vital resource, as the review provides a precise definition of resilience in railway transport (see Fig. 3), tailored explicitly towards railways. Furthermore, future research directions are given. Note that both the recommendations of Zhou *et al.* (2019) and Bešinović (2020) support each other.

According to Bešinović (2020), four approaches apply to assess resilience in railway transport:

1. *Topological approaches* represent the network on a high level of abstraction and thus, require limited data, while their use is limited to general characteristics. Identifying a critical link/component requires the removal of elements individually and hence requires considerable effort, especially when looking for a combination of critical components.
2. *Simulations* allow for a much higher level of detail than topological approaches but require more resources and a detailed understanding of the system's dynamics. Besides, finding a combination of critical components still requires an enumeration of all possible combinations.
3. *Optimization* aims to overcome the shortcomings of the previous two approaches since it can effectively deal with combinatorial difficulties and consider the system's dynamics.
4. *Data-driven approaches* completely contrast with all former ideas, as no system model is required to gain specific insights on the system's resilience. However, such approaches are currently limited on a posterior analysis and require a considerable amount of high-quality data.

Among the studies that focus on railways, Zhu and Goverde (2017) propose guiding the application of mitigation measures by using a model that allows firstly measuring the vulnerability of the

system and secondly evaluating the effect of a mitigation measure on the network. As the model uses historical data within a Monte Carlo simulation, the authors can identify the top ten most vulnerable tracks and compare scenarios with and without mitigation measures.

Intending to increase the resilience of a metro network, [Jin et al. \(2014\)](#) focus on introducing localised integration with bus services rather than designing a bus network that provides the demanded resilience against potential disruptions. Essentially, [Jin et al. \(2014\)](#) demand more capacity on bus services parallel to the disrupted metro lines. Using a two-stage stochastic programming model, [Jin et al. \(2014\)](#) demonstrate that localised integration with bus services yields significant improvements for resilience.

With a similar intention, however, addressing pre-defined disruptions in the form of railway possessions, [Borecka and Bešinović \(2021\)](#) proposes a model that allows to schedule multi-modal alternative services to bridge the gap in the network. [Borecka and Bešinović \(2021\)](#) further show that introducing alternative services allows routing affected passengers with a minimal increase in additional passenger costs.

[Bababeik et al. \(2017\)](#) aim to analyze the vulnerability of railway networks in case of a multi-link blockage by using flow interdiction. By removing individual links and rescheduling trains in the residual network, [Bababeik et al. \(2017\)](#) can assess the vulnerability based on the impact on passenger flows. Furthermore, as the approach only requires partial enumeration, the proposed heuristic is less demanding. [Szymula and Bešinović \(2020\)](#) propose the Railway Network Vulnerability Model (RNVM). It allows identifying the weak links in a railway network. The RNVM provides a vulnerability assessment by identifying the critical links that drastically impact passengers when disrupted. Although the model itself is an optimization model, identifying critical links relies on pre-computed values that encompass the attractiveness of a passenger path. As these values are calculated based on the nominal (non-disrupted) timetable, the resulting values approximate the actual vulnerability.

All three approaches have in common that they consider vulnerability in some form. As vulnerability encompasses the remaining performance while disrupted, these assessments allow a proactive evaluation of railway networks.

2.2.4 Scenario-Based Approaches to Assess and Enhance Resilience

According to [Ghaemi et al. \(2017b\)](#), disruption management in practice is a two-fold approach, consisting of the preparation of contingency plans and the subsequent implementation of these when a disruption occurs. As stated previously, a contingency plan is a predetermined procedure detailing how to handle a disruption at a given location ([Dekker et al., 2021](#)). Thus, although disruption management generally focuses on dealing with disruptions on the operational level, the preparation of contingency plans increases preparedness against predefined disruptions ([Bešinović, 2020](#)). However, as [Ghaemi et al. \(2017b\)](#) stated, the effects and applicability of contingency plans as a scenario-based approach are limited, especially given the theoretically endless combinations of failures and the increasing utilisation of railway infrastructure.

Note that not all failures require an adaptation of the timetable. For example, in cases where the impacts are only minor, the robustness of the timetable might suffice to absorb the adverse effect (Ghaemi *et al.*, 2017b). Besides, minor adversity might be resolved satisfactorily using real-time rescheduling, see Cacchiani *et al.* (2014) for more details.

In contrast, Dekker *et al.* (2021) introduce the term of *out-of-control situations*, which occur due to disasters or drastic propagation of adverse effects caused by disruptions. Consequently, out-of-control situations are among the most critical forms of disruption scenarios, where due to the lack of overview and information, a significant section of the railway system is inoperable. A common factor is that most of the required resources (e.g., infrastructure, trains) are still available in these situations. Nevertheless, dispatchers are no longer able to take adequate measures (Dekker *et al.*, 2021).

Dekker *et al.* (2021) propose to deal with such situations by decoupling the disrupted network sections from the unaffected remainder and using two different paradigms to operate the sections. First, while the unaffected section operates following the timetable and is decoupled, the second and affected section relies on complexity science's self-organising principles.

Apart from these edge cases, scenario-based approaches focus on addressing and reducing the impact on the performance of known disruptions. Note that only a few propositions address survivability, response and recovery as most focus on the isolated response (Ghaemi *et al.*, 2017b). For a general overview, we refer to Ghaemi *et al.* (2017b), who provide insights into how Dutch practitioners manage occurring disruptions and an overview of proposed approaches in the literature.

Among the approaches considering macroscopic infrastructure and using a scenario-based approach, Louwse and Huisman (2014) propose a model that adjusts the timetable of a railway operator when facing major disruptions that may lead to either a partial or a complete blockage of a railway line. Also focusing on large-scale disruptions, Veelenturf *et al.* (2016) present a *Mixed Integer Program* (MIP) that minimises the number of cancelled and delayed train services while considering the rolling stock capacity and the possibility to reroute the trains.

Van Aken *et al.* (2017) introduce the *Train Timetable Adjustment Problem*, which provides an alternative timetable with minimal deviation from the original timetable under the consideration of the given station and open-track possessions. The problem is again formalised as MIP.

In contrast to the previously presented work, Ghaemi *et al.* (2017a) approach the case of complete blockages on a microscopic scale. Ghaemi *et al.* (2017a) propose a MIP model to compute a disruption timetable with optimal short-turning stations, routes, and platform tracks. However, given the microscopic nature of the model, it is not applicable to bigger instances.

All these previous approaches have in common that although the passenger routes are affected by the disruption and subsequent responses, the routes are only considered implicitly. In contrast, Veelenturf *et al.* (2017) propose a heuristic approach that considers that passengers adapt their

route and provides superior results from a passenger's perspective. Also intending to design more passenger-oriented disruption timetables, [Zhu and Goverde \(2019\)](#) propose a dynamic passenger assignment model that replicates the behaviour of passengers. The model thus provides an essential step towards the integration of passengers in disruption management, as it makes it possible to evaluate a disruption timetable, including the passenger's perspective.

[Besinovic et al. \(2019\)](#) propose an integrated approach to manage disruption within urban railway systems. Combining a rescheduling model and a model for adjusting passenger flows minimises the delay experienced by the passengers due to the disruption. [Zhu and Goverde \(2020b\)](#) propose a MIP model which allows the joint rescheduling of trains and reassignment of passengers. The objective function minimises the generalised travel time of all passengers and accounts for both in-vehicle time and waiting time at origin and transfer stations. [Zhu and Goverde \(2020b\)](#) demonstrate that their proposition provides solutions that yield shorter generalised travel time when compared with operator-based approaches. [Zhu and Goverde \(2020a\)](#) addresses all stages of disruption and accounts for the uncertainty of when the system will recover using a rolling horizon. Based on this approach, [Zhu and Goverde \(2020a\)](#) can provide different transition plans that do not depend on a predetermined recovery point. The presented approach is hence likely to provide better solutions with fewer delays and cancellations.

Multiple simultaneous disruptions present a unique challenge, which [Zhu and Goverde \(2021\)](#) tackle with an approach capable of accounting for multiple disruptions during overlapping time intervals. Using this model, [Zhu and Goverde \(2021\)](#) show that the joint consideration of disruptions can find solutions where a sequential approach that resolves the disruptions individually fails. Furthermore, the joint approach requires fewer adaptations of the original timetable, providing superior solutions.

In conclusion, scenario-based approaches benefit from an advanced state of research, as many solutions have been proposed. A further conclusion that can be drawn is that scenario-based approaches frequently focus on adjusting the timetable to respond to a disruption. These adjustment problems are usually solved with MIP formulations. The current state of research yields models that allow assessing disruptions fast and efficiently. Eventually, different objectives and perspectives (passenger- or operator-oriented) have been suggested and evaluated.

2.2.5 Network Interdiction

The task of interdicting a network can be seen as the antithesis of enhancing its resilience. Commonly studied in the defence context, network interdiction describes the task of limiting the operations of the antagonist with specific interdiction measures. These measures aim to alter or block the network infrastructure of the opposition such that their value diminishes maximally. For an overview of network interdiction models and algorithms, we refer to [Smith and Song \(2020\)](#).

Although not directly related to the railway planning process, we may adopt network interdiction methods, since the advanced research state provides reliable foundations for railway

resilience assessment and enhancement optimisation models, especially when identifying critical infrastructure.

2.3 Identifying the Research Gap

Building on the previously introduced literature, we highlight identified research gaps to summarise our literature review.

Integrating resilience into the early stages of the planning process is one of the future research directions suggested by [Bešinović \(2020\)](#). Some recently proposed approaches allow finding a critical link ([Szymula and Bešinović, 2020](#)), the best response to given disruption ([Zhu and Goverde, 2020b](#)), or the inclusion of bus bridging services to reduce the loss of performance ([Borecka and Bešinović, 2021](#)). However, all of these approaches use an existing timetable as input, for which the resilience is assessed, while none provides feedback on how to adapt the timetable. We aim to bridge this gap by providing an approach that integrates optimising the timetable and increasing resilience.

Besides, jointly identify the critical disruption and the best response is something that has not been addressed explicitly for railways so far. Current work either limits the scope on a network-wide perspective or scenario-based approaches. Thanks to the network-wide perspective, the former allows the identification of vulnerable components, such as critical links ([Szymula and Bešinović, 2020](#)) and yields network-wide performance indicators ([Zhou et al., 2019](#)). On the contrary, scenario-based approaches can yield accurate and detailed information on the effect of a disruption ([Ghaemi et al., 2017b](#)); however, these models all rely on predefined scenarios ([Zhu and Goverde, 2020b](#); [Van Aken et al., 2017](#)). We aim to bridge this gap by providing an approach that is similar to ([Szymula and Bešinović, 2020](#)); however, our model does not rely on pre-computed link importance. Instead, we propose identifying the critical disruption while considering that passengers adapt their travel route to react to the disruption. Besides, we aim to identify the critical disruption under the consideration of typical responses, such as short turning and cancellations ([Ghaemi et al., 2017b](#)). However, as resilience is a system-wide concept ([Bešinović, 2020](#)), we follow the proposition of [Borecka and Bešinović \(2021\)](#) to allow responding to disruption with an alternative in the form of buses. Consequently, we bridge the gap between approaches that detect the critical disruptions from a network perspective and scenario-based approaches for finding the best response. Following a recommendation for future research by [Bešinović \(2020\)](#), our approach should be capable of coping with single and simultaneous disruptions.

3 Methodology

The methodology consists of four sections. Initially, we provide a brief problem description in Section 3.1, providing the necessary background and concepts while also addressing SQ 1 and SQ 2. Next, we introduce and motivate the structure of the overall approach in Section 3.2, mainly addressing SQ 3. Then we give individual insights into the two modules of the approach in Section 3.3 and Section 3.4, thus finalising SQ 3.

3.1 Problem Description

Before developing the required methodology, we narrow down the scope and describe all considered perspectives, allowing us to define the resilience metric. We then propose the **Train Slot Sequence (TSS)** as the fundamental data structure.

3.1.1 Narrowing the Scope

In the scope of this thesis, we aim to provide an approach to optimise the performance of the railway’s system while also assessing and enhancing its capability to cope with disruptions. Similar to [Szymula and Bešinović \(2020\)](#), we focus on typical disruptions in the form of complete track blockages. Thus, a disruption represents an infrastructure section ($h \in \mathcal{H}$) between two stations, completely blocked. Let \mathcal{H} be the set of all infrastructure sections between stations, which may be blocked. Then, $\mathcal{H}_{\text{disrupted}} \subseteq \mathcal{H}$ yields all the sections that are currently disrupted, i.e., blocked. Although we limit the size of $\mathcal{H}_{\text{disrupted}}$ to a given budget, $k_{\text{disrupted}}$, we consider all disruption scenarios where $|\mathcal{H}_{\text{disrupted}}| \in [1, k_{\text{disrupted}}]$. Hence, we detect both single and combinations of simultaneous disruptions failures.

As indicated by [Szymula and Bešinović \(2020\)](#), the impact of disruptions can vary depending on $\mathcal{H}_{\text{disrupted}}$. Thus, we focus on the most critical disruption, characterised by the $\mathcal{H}_{\text{disrupted}}$ which causes the highest impact on performance. Ideally, we can adapt the timetable to absorb these critical disruptions as the impact is reduced. Consequently, we define *resilience* as the “ability to absorb the most critical disruptions with as little impact on the performance as possible”.

[Zhu and Goverde \(2020a\)](#) show that the exact beginning and end of a disruption considerably affect the total impact on performance. However, this requires comprehensive modelling of survivability and recovery, exceeding the scope of the planning phase. Hence, we limit the scope of our approach to the response state, as commonly done in this planning phase ([Szymula and Bešinović, 2020](#); [Borecka and Bešinović, 2021](#)). As we are looking at single and simultaneous disruptions, we select an optimisation approach to overcome the combinatorial challenges of identifying and addressing $\mathcal{H}_{\text{disrupted}}$ for the critical disruption ([Zhou et al., 2019](#); [Bešinović, 2020](#)).

According to [Caimi et al. \(2017\)](#), European Railway timetables are likely to follow a periodic pattern. Therefore, we also decided to pursue a periodic pattern to ensure that our approaches

are compatible and meet the strategic and tactical planning requirements. Therefore, we use \mathcal{T} to denote the interval of the planning period. Furthermore, dt represents a time step.

3.1.2 Considered Perspectives

We explicitly consider the passengers perspective by integrating the demand during \mathcal{T} . We structure the demand in the form of a bipartite, directed graph \mathcal{OD} , where O is the set of all origins while D denotes all destinations. Besides, γ^{od} contains the number of passengers p on the relation od that desire to travel from o to d . Given \mathcal{OD} , we integrate the passengers perspective by evaluating the solution quality considering all $\gamma^{od} \in \mathcal{OD}$. Secondly, we account for the [Train Operating Company \(TOC\)](#) perspectives by considering their intentions in the form of formalised commercial requirements, characterised by restricted time windows. Additionally, we also consider the number of available vehicles (given by $\max(r)$) per type of available vehicles $r \in \mathcal{R}$. Furthermore, we also consider the capacity of a vehicle (given by $\text{capacity}(r)$).

Finally, we also consider the [Infrastructure Manager \(IM\)](#) perspective, which has to find a satisfactory solution for passengers and all [TOC](#), including those that transport other commodities such as cargo. Note that we only implicitly address the requirements of [TOCs](#) that transport other commodities as passengers, as their requirements are considered in the timetable, but they are not part of the objective. Besides, the [IM](#) aims to utilise the available infrastructure optimally ([Fuchs et al., 2021](#)). Thus, we integrate the task of assigning tracks during [TTP](#), aiming for more flexibility by including all options.

3.1.3 Defining the Performance Metric

Based on the recommendation by [Zhou et al. \(2019\)](#), we opt for a *Performance-Based Metric* to capture the impact of a disruption. Furthermore, given the passenger perspective as part of the scope, travel time is a natural indicator to evaluate the performance of a timetable. Besides, travel time is frequently chosen as an objective in both the railway planning process ([Caimi et al., 2017](#)) and the disruption management ([Zhu and Goverde, 2020b](#)). Consequently, we select the generalised total travel time as defined by [Zhu and Goverde \(2020b\)](#) to evaluate the performance of a timetable.

To define our performance metric, we introduce corresponding notation. Let \mathcal{L}_{od} be the set of links which are used by γ^{od} to travel from o to d . Furthermore, we assign each $l \in \mathcal{L}$ a duration \bar{w}_l and weighting factor c_l to generalise it. As it is possible, that not all passengers travel on the same path, we use p_l^{od} to denote the fraction of γ^{od} travelling on a particular l . Consequently, we can calculate the generalised (total) travel time C as follows:

$$C = \sum_{od \in \mathcal{OD}} \sum_{l \in \mathcal{L}_{od}} c_l^{od} \cdot \bar{w}_l \cdot p_l^{od} \quad (1)$$

C as defined in Eq. (1) enables us to evaluate the performance P of a timetable when looking for optimal timetables and assessing the impact of disruptions. Typically, for the same \mathcal{OD} , a lower C indicates better P due to reduced generalised travel time. Let C_{ideal} be the generalised

travel time for a timetable, where C is as low as possible for the given problem instance. A trivial example for C_{ideal} is a timetable allowing all passengers to travel on their shortest possible path. Consequently, C_{ideal} the lower bound of C . To find an upper bound on C , we define $C_{\text{worst-case}}$. We consider that passengers will look for an alternative means of transport if C for this od exceeds a certain limit of generalised travel time given by c_{max}^{od} . This assumption is taken frequently, (Schmidt and Schöbel, 2015b; Zhu and Goverde, 2020b; Szymula and Bešinović, 2020). Assuming that we are aware of c_{max}^{od} for all $od \in OD$, we can define $C_{\text{worst-case}}$ as follows:

$$C_{\text{worst-case}} = \sum_{od \in OD} c_{\text{max}}^{od} \cdot \gamma^{od} \quad (2)$$

In essence, $C_{\text{worst-case}}$ in Eq. (2) defines C in the case where no service for any $od \in OD$ is provided, i.e., no trains run. Based on the definition of C_{ideal} and $C_{\text{worst-case}}$, we can derive a relative value P , indicating the performance and allowing us to rank any timetable that we found by its nominal generalised travel time C_{nominal} , without any disruption.

$$P_{\text{nominal}} : 100 - \left(\frac{C_{\text{nominal}} - C_{\text{ideal}}}{C_{\text{worst-case}} - C_{\text{ideal}}} \right) \cdot 100 \quad [\%] \quad (3)$$

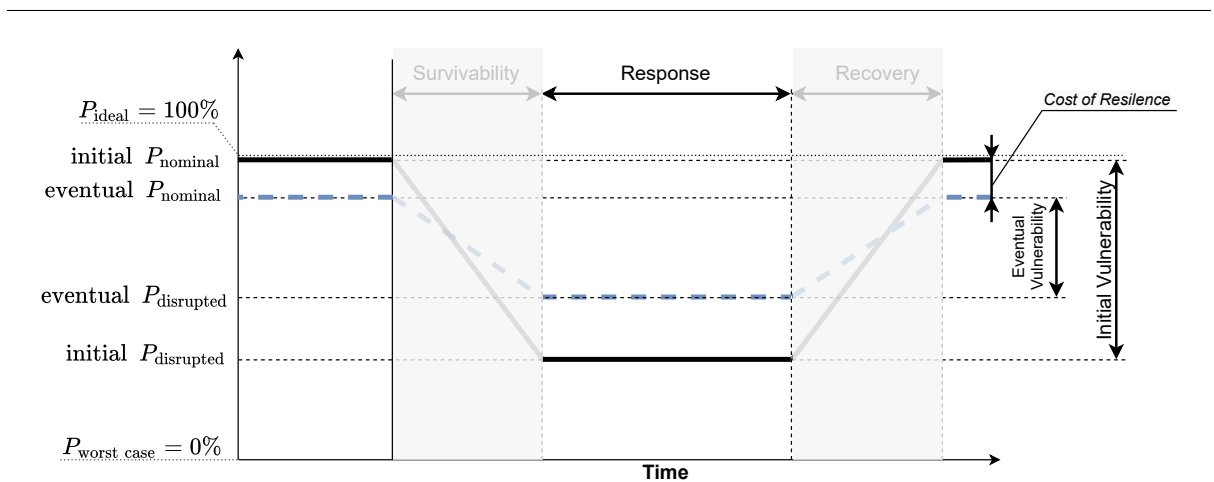
Similarly, we can also rank the generalised travel time of a timetable when it is disrupted, given by $C_{\text{disrupted}}$:

$$P_{\text{disrupted}} : 100 - \left(\frac{C_{\text{disrupted}} - C_{\text{ideal}}}{C_{\text{worst-case}} - C_{\text{ideal}}} \right) \cdot 100 \quad [\%] \quad (4)$$

Based on the definition of P_{nominal} in Eq. (3) and $P_{\text{disrupted}}$ in Eq. (4), it is apparent that for the ideal performance $P_{\text{ideal}} = 100\%$ holds, while $P_{\text{worst-case}} = 0\%$. Note that we deliberately distinguish between an *ideal* and *nominal* (optimal) C and P . This distinction allows us to compare several different performances for solutions of the same problem instance, even when the solutions differ in P_{nominal} . This might be especially relevant in cases, where due to constraints or the use of heuristics, it is not possible to find a P_{nominal} with 100% performance.

Based on the SQ 1 and SQ 2, we visualise our aim. Therefore, we adapt the bathtub model of Bešinović (2020), presented in Section 2, Fig. 3 using the proposed performance metric.

Figure 4: The performance metric visualised with the resilience bathtub curve.



The adapted bathtub in Fig. 4 yields two curves, one before applying the approach (*initial*) and one for the final result (*eventual*). As depicted, we aim to provide an approach that allows the reduction of the impact of the *critical* disruption. As stated in Section 3.1.1, the critical disruption is the one disruption among the set of possible disruptions causing the highest impact. The impact is given by the absolute difference in P_{nominal} and $P_{\text{disrupted}}$, similar to vulnerability. We consider two approaches to reduce the impact. Firstly, we include available response measures and resources to reduce the disruption’s impact optimally. Secondly, we aim to adapt the initial timetable to a potentially less vulnerable one. Ideally, both response and adaptation complement each other, such that resilience increases.

However, since we adapt the timetable, we might cause the *nominal* performance to deteriorate. Therefore, we suggest to also consider the loss in P_{nominal} , which we define as the *Cost of Resilience*, depicted in Fig. 4. Neglecting the cost of *Cost of Resilience* might lead to adverse results. A trivial example for such a case is if the *eventual* $P_{\text{nominal}} = P_{\text{worst-case}}$, i.e., we cancel all trains. Although such a step eliminates the vulnerability, it is apparent that the *Cost of Resilience* exceeds any acceptable limit. Note that thanks to P_{ideal} , we may also compare a set of solutions.

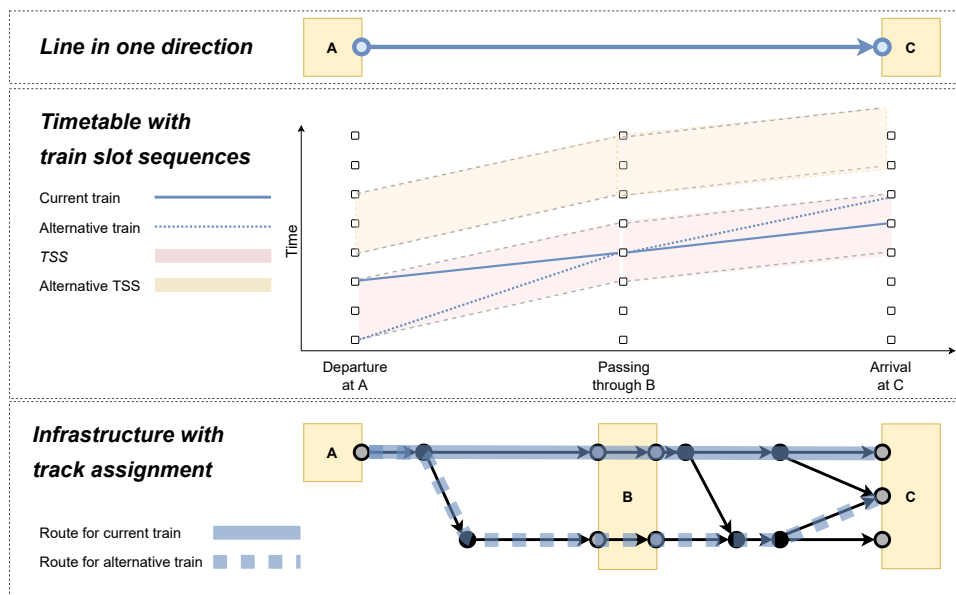
3.1.4 Proposing the TSS to encompass TOC Requirements

As stated in Section 3.1.2, our proposition should be capable of capturing a wide range of (commercial) requirements demanded by the TOC. Hence, we introduce the concept of **Train Slot Sequences (TSSs)** which restricts the trains to run within predefined time windows. This concept is similar to the **Service Intention (SI)** proposed by Caimi *et al.* (2011b) and Wust2019PeriodicInfrastructure as a means to formalise commercial requirements during railway planning in practice. Another similar concept is the **Train Path Envelope (TPE)** as used in Wang *et al.* (2020). Nevertheless, whereas a TPE yields a conflict-free timetable as long as all trains run within their TPEs, the same does not hold for a SI or a TSS. However, since both TPE and SI are also useful to specify commercial requirements (e.g., latest departure), we assume that using TSS similarly is compatible with the railway planning process.

Besides, using slots is also reasonable when considering the context. As we focus on strategic and tactical planning horizons, there may be situations where we can only rely on a line plan as input. However, a line plan generally does not yield any information on duration and time. Thus, using slots allows us also to include the requirements of TOCs. Furthermore, in a more tactical environment with shorter time horizons, the room for adjustments might be considerably smaller, which we can again incorporate by adjusting the number of available **Train Slot Sequences (TSSs)**.

Although we will subsequently provide a formal definition of the TSS, Fig. 5 provides an overview on the general concept.

Figure 5: The TSSs to couple the line plan, timetable and infrastructure.

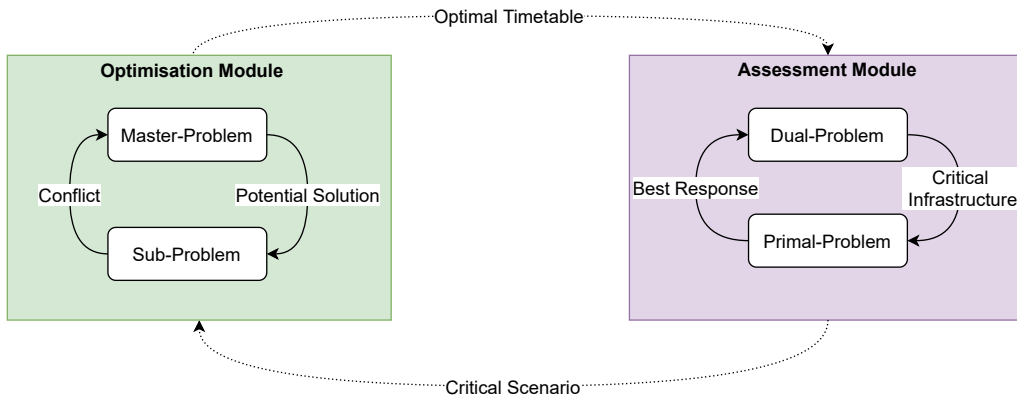


As depicted in Fig. 5, a TSS limits a train to run within the *slots* which are given by a TSS. However, the scheduled events can be adjusted as long as they remain within the sequence of slots. Furthermore, a TSS does not limit the train to use a particular track if alternative routing options are available. Thus, a TSS is more restrictive than a line since it restricts the train to certain time windows. However, it still leaves room for adjusting the train run and the track assignment.

3.2 Overall Approach

Next, we provide an overview of our approach. The integrated optimisation of railway timetables is already challenging without consideration of resilience (Schiewe, 2020). Thus, we propose an iterative approach to tackle optimisation, assessment and improvement of the timetable. The iterative model is depicted in Fig. 6, and it consists of two main components, the **Optimisation Module (OM)** and the **Assessment Module (AM)**.

Figure 6: An overview on the Approach.



Essentially, the **OM** yields P_{ideal} , P_{nominal} and $P_{\text{worst-case}}$ for a timetable, while the **AM** provides $P_{\text{disrupted}}$ and information on the most critical disruption $\mathcal{H}_{\text{critical}}$. This information is then used by the **OM** to adapt the timetable before demanding another assessment from the **AM**. In the remainder, we now briefly outline each module.

Optimisation Module (OM) As given by its name, the **OM** allows to find an optimal timetable under consideration of the perspectives given in Section 3.1.2. Given the broad scope addressed, we propose a logical-Benders decomposition of the **TTP** task as suggested by Hooker and Ottosson (2003). Using **TSS**, we can decompose the **TTP** into the tasks of finding an optimal solution and checking for conflicts. Such separation attempts have already proven to be beneficial when addressing the task of passenger routing and timetabling jointly in **TTP** (Polinder *et al.*, 2021; Szymula and Bešinović, 2020).

Assessment Module (AM) As the second module, the **AM** addresses the remaining tasks of assessing the resilience with and without response actions. As stated in Section 2.3, we do not rely on predefined disruption scenarios when identifying the most critical disruption $\mathcal{H}_{\text{critical}}$ based on $P_{\text{disrupted}}$. Note that both, the **OM** and the **AM** are utilising **TSS**. Hence, we ensure consistency when transferring a solution between the modules. With the same intention, potential responses for addressing a $\mathcal{H}_{\text{critical}}$ with disruption management are also defined with **TSS**.

Note that the outlined iterative process does not necessarily lead to a solution where $P_{\text{disrupted}}$ is improved. For instance, there might be cases where disruption of $\mathcal{H}_{\text{critical}}$ leads to a complete separation of the network, and no response is available. However, we assume that even under such circumstances, the information provided by the approach is valuable, as it reveals insights on the weaknesses of the railway network, even when considering all responses. Thereby, we can guide future mitigation measures. Furthermore, one can compare the effectiveness of different responses or mitigation measures. Consequently, the iterative procedure provides decision support even when the network's resilience can not be improved immediately.

3.2.1 Fundamental Data Structures

Given that we established the basic structure of the approach, we now introduce two key networks. To allow the logical-Benders decomposition, we utilise two networks, the **Slot Sequence Network (SSN)** and the **Event Activity Network (EAN)**. The **SSN** allows us to select the **TSSs** and to route commodities when looking for an optimal solution. Complementary, the **EAN** enables us to identify conflicts in a given solution. Hence, rather simplified, the **SSN** provides a view from *commercial* aspects, whereas the **EAN** reflects *technical* perspectives. Next, we briefly introduce each network before we successively show how we can connect both networks with the **TSS**.

The Slot Sequence Network (SSN) is a directed graph composed of nodes \mathcal{N} and links \mathcal{L} . We will use the **SSN** to select the **TSSs** and to route commodities when optimising the timetable. For each node, we can specify a \bar{t}_n for which $0 \leq \bar{t}_n < \mathcal{T}$ holds. t_n allows us to specify a fixed event time, at which this node takes place. Furthermore, each link has a constant duration \bar{w}_l . Based on the purpose, we can select subsets for both, \mathcal{N} and \mathcal{L} .

For \mathcal{N} , \mathcal{N}^{arr} and \mathcal{N}^{dep} yield all arrival or departure nodes. Another subset of \mathcal{N} is given \mathcal{N}^{com} or $\mathcal{N}^{\text{n-com}}$, consisting of all commercial or non-commercial nodes. Thus, $\mathcal{N}^{\text{com}} \cap \mathcal{N}^{\text{arr}}$ yields all commercial arrival nodes. Similar to Wang *et al.* (2020), we consider arrivals and departures also at non-commercial intermediate stations, even if the trains only pass.

Finally, $\mathcal{N}^{\text{station}}$ yields all station nodes, i.e., change $\mathcal{N}^{\text{change}}$, access $\mathcal{N}^{\text{entry}}$ and exit nodes $\mathcal{N}^{\text{exit}}$. In our formulation, $\mathcal{N}^{\text{station}}$ are crucial for both, the assignment of vehicles and passengers.

Naturally, we may also select subsets of \mathcal{L} . While $\mathcal{L}^{\text{dwell}}$ covers all dwell links, $\mathcal{L}^{\text{trip}}$ spans all trips. Train links represent the union of dwells and trips $\mathcal{L}^{\text{dwell}} \cup \mathcal{L}^{\text{trip}} = \mathcal{L}^{\text{train}}$. Similar to $\mathcal{N}^{\text{station}}$, $\mathcal{L}^{\text{station}}$ contains all station activities necessary for the assignment of vehicles and passengers, such as access $\mathcal{L}^{\text{access}}$, waiting $\mathcal{L}^{\text{wait}}$ and egress links $\mathcal{L}^{\text{egress}}$.

Event Activity Network (EAN) The **EAN** is a common approach to model **TTPs** in form of a directed graph (Caimi *et al.*, 2017). The set of events \mathcal{E} contains all events to schedule, while the activities \mathcal{A} reflect all dependencies between \mathcal{E} . Each $e \in \mathcal{E}$ is constrained to take place at time $t_e \in [t_e^{\text{min}}, t_e^{\text{max}}]$, with $t_e^{\text{min}} = 0$ and $t_e^{\text{max}} = \mathcal{T} - dt$ unless specified differently. Similarly, the

duration of an activity w_a is restricted by $w_a \in [w_a^{\min}, w_a^{\max}]$. As before, we can select subsets for both, \mathcal{E} and \mathcal{A} .

For \mathcal{E} , \mathcal{E}^{arr} and \mathcal{E}^{dep} yield all arrival or departure events, while $\mathcal{E}^{\text{pass}}$ contains all passing events at intermediate junctions. Another subset of \mathcal{E} is given \mathcal{E}^{com} or $\mathcal{E}^{\text{n-com}}$, consisting of all commercial or non-commercial events. As trains do not stop at passing nodes, $|\mathcal{E}^{\text{com}} \cap \mathcal{E}^{\text{pass}}| = 0$ holds. Again following Wang *et al.* (2020), trains also arrive and depart at non-commercial intermediate stations, although they do not necessarily stop. Lastly, $\mathcal{E}^{\text{anchor}}$ yields all anchor events, necessary to restrict $e \in \mathcal{E}^{\text{com}} \cap \mathcal{E}^{\text{arr}} \cap \mathcal{E}^{\text{dep}}$ to take place in the time windows imposed by the TSS.

$\mathcal{A}^{\text{dwell}}$ cover all dwell activities, $\mathcal{A}^{\text{trip}}$ span all trips and train activities represent the union of both $\mathcal{A}^{\text{dwell}} \cup \mathcal{A}^{\text{trip}} = \mathcal{A}^{\text{train}}$. As common in periodic TTPs, headway activities $\mathcal{A}^{\text{headway}}$ ensure that a timetable is free of conflicts. Finally, $\mathcal{A}^{\text{anchor}}$ impose the time windows given by the TSSs.

In contrast to the SSN, the EAN does not contain any commercial requirements, as we use the TSS for such purpose.

3.2.2 Defining the Train Slot Sequence (TSS)

In the following section, we define the TSS more rigorously, before we show how we convert lines part of line plan into our input for the OM. Based on this definition, we may then decompose the TTP in the OM with logical-Benders decomposition.

Our approach requires a line plan \mathcal{K} as input. In contrast to common notation, our definition of \mathcal{K} contains *directed lines* $\kappa \in \mathcal{K}$ (Schmidt and Schöbel, 2015b). As implied by the name, a *directed line* covers one directed segment from the start to the end of one line. As an example, if a plan contains three lines in standard notation, said \mathcal{K} consists of six *directed lines* in our case. One *directed line* indexed by κ is defined as the tuple Ω_κ containing the following elements:

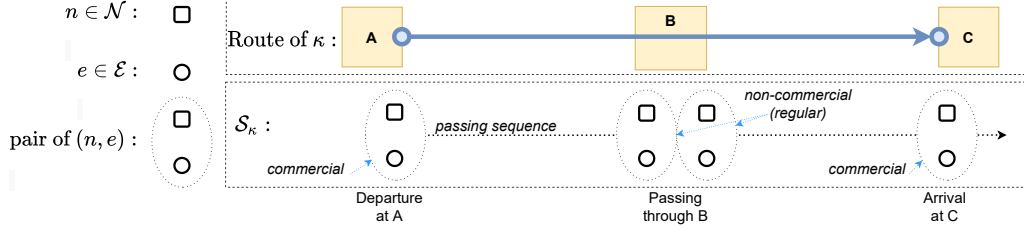
$$\Omega_\kappa := (\mu_\kappa^{\text{com}}, \mu_\kappa^{\text{gen}}, \Pi_\kappa, f_\kappa, \mathcal{S}_\kappa) \quad (5)$$

The parameters $\mu_\kappa^{\text{com}}, \mu_\kappa^{\text{gen}}$ and $\pi \in \Pi_\kappa$ represent the requirements submitted by the TOCs under which they are willing to operate κ . Both $\mu_\kappa^{\text{com}}, \mu_\kappa^{\text{gen}}$ are integer multiples of dt and specify the width of the slot sequences. μ_κ^{com} defines the slot width at a *commercial* arrival or departure at a station. μ_κ^{gen} defines additional slot width to add at any *general* arrival or departure (i.e., one that is not of commercial purpose). Each $\pi \in \Pi_\kappa$ represents an absolute offset by which the slot sequence can be shifted. Note that any $\pi \in \Pi_\kappa$ is also an integer multiple of dt and $0 \leq \pi < \mathcal{T}$ holds.

\mathcal{S}_κ defines the route of κ , as it contains a sequence of (n, e) pairs, which essentially define the route of κ . Each of these pairs consists of either an $e \in \mathcal{E}^{\text{dep}}$ and an $n \in \mathcal{N}^{\text{dep}}$ or of an $e \in \mathcal{E}^{\text{arr}}$ and an $n \in \mathcal{N}^{\text{arr}}$. The sequence of the (n, e) pairs is given by the direction of κ , thus each (n, e) pair either covers the arrival or departure at a station served or passed by κ .

We provide an illustration in Fig. 7 to visualise how the sequence of pairs $(n, e) \in S_\kappa$ defines the route of κ . As we have to add one pair for each station arrival or departure, we need four pairs in Fig. 7. While the first one is a commercial departure, the two intermediate ones are a non-commercial arrival and departure respectively. The last pair is a commercial arrival.

Figure 7: An illustration of how S_κ captures the route of κ .



These (n, e) pairs are a one to one mapping between some $n \in \mathcal{N}$ and some $e \in \mathcal{E}$. Thus, each e is at most member of one (n, e) pair, and vice versa for n . However, as there are some $n \in \mathcal{N}$ and some $e \in \mathcal{E}$ which are not member of a (n, e) pair, we may not define one as an attribute of the other.

Let $\hat{t}_{n,e}$ be the (minimal) time required to reach any $(n, e) \in S_\kappa$ from the first (n, e) on. We may then calculate all slots constraining the time window for a $(n, e) \in S_\kappa$ given by $t_{(n,e)}^{\min}, t_{(n,e)}^{\max}$ for all $(n, e) \in S_\kappa$ and a given $\pi \in \Pi_\kappa$ as follows:

$$t_{(n,e)}^{\min}, t_{(n,e)}^{\max} := \begin{cases} \hat{t}_{n,e} + \pi, \hat{t}_{n,e} + \pi + \mu_\kappa^{\text{com}} & \text{if } n \in \mathcal{N}^{\text{com}} \\ \hat{t}_{n,e} + \pi - \mu_\kappa^{\text{gen}}, \hat{t}_{n,e} + \pi + \mu_\kappa^{\text{com}} + \mu_\kappa^{\text{gen}} & \text{else} \end{cases} \quad (6)$$

As indicated by Eq. (6), a slot for a given event $(n, e) \in S_\kappa$ has at least the width μ_κ and is extended by $2 \cdot \mu^{\text{gen}}$ if $n \notin \mathcal{N}^{\text{com}}$. Furthermore, we use $\pi \in \Pi_\kappa$ to shift the sequence of slots by π such that $|\Pi_\kappa|$ slot sequences are available for a given κ .

We use $\mathcal{Z}_{\kappa,\pi}$ to denote a slot sequence given by κ and $\pi \in \Pi_\kappa$, while \mathcal{Y}_κ holds all $\mathcal{Z}_{\kappa,\pi}$ defined by κ . Summarising our definition, a slot $z \in \mathcal{Z}_{\kappa,\pi}$ is defined as follows:

$$z := (n_z, e_z, t_z^{\min}, t_z^{\max}) \quad (7)$$

Each $z \in \mathcal{Z}_{\kappa,\pi}$ associates a node n_z and an event e_z with the according time window $[t_z^{\min}, t_z^{\max}]$, that is specified by κ and π of $\mathcal{Z}_{\kappa,\pi}$. Thus, we can use the $z_{\kappa,\pi}$ to restrict $t_e \in [t_z^{\min}, t_z^{\max}]$ and define $\bar{t}_n = t^{\max} \bmod \mathcal{T}$. Let $\mathcal{N}_{\kappa,\pi}$ be the subset of all $n \in \mathcal{N}$ affected by $\mathcal{Z}_{\kappa,\pi}$ and likewise $\mathcal{E}_{\kappa,\pi}$ be the subset of $e \in \mathcal{E}$. Besides, let $\mathcal{Y}_{\text{available}}$ be the set of all $\mathcal{Z}_{\kappa,\pi}$ given by κ and $\pi \in \Pi_\kappa$ for all $\kappa \in \mathcal{K}$. Essentially, $\mathcal{Y}_{\text{available}}$ contains all available $\mathcal{Z}_{\kappa,\pi}$, of which we then have to select a subset $\mathcal{Y}_{\text{active}}$ which we operate in our timetable.

3.2.3 Generating the *SSN*

Given that both, the *TSS* and *SSN* are defined, we subsequently show how we create *SSN* based on a given \mathcal{K} which defines $\mathcal{Y}_{\text{available}}$. Besides, we need two additional parameters for *egress* and *access* time, μ_{access} and μ_{egress} . As indicated μ_{access} defines the minimal \bar{w}_l to transfer from an $n \in \mathcal{N}^{\text{station}}$ to an $n \in \mathcal{N}^{\text{dep}}$. Likewise, μ_{egress} applies to \bar{w}_l to transfer from an $n \in \mathcal{N}^{\text{arr}}$ to $n \in \mathcal{N}^{\text{station}}$.

The generation process covers two steps. Initially, we add all network components required for each $\mathcal{Z}_{\kappa,\pi} \in \mathcal{Y}_{\text{available}}$. Eventually, we link these components with $\mathcal{N}^{\text{station}}$, such that the network is suitable to route commodities.

Since the definition of a $\mathcal{Z}_{\kappa,\pi}$ already defines timestamps \bar{t}_n for the associated $n \in \mathcal{N}_{\kappa,\pi}$ introducing $\mathcal{N}^{\text{train}}$ is straightforward. Furthermore, as $\mathcal{Z}_{\kappa,\pi}$ also defines the order of $n \in \mathcal{N}_{\kappa,\pi}$ we can connect the successive pairs $n_i, n_j \in \mathcal{N}_{\kappa,\pi}$ with the appropriate $l \in \mathcal{L}^{\text{train}}$. Assuming that there is no $\bar{w}_l > \mathcal{T}$ at this stage, calculating the duration of \bar{w}_l for an $l \in \mathcal{L}^{\text{train}}$, is as follows;

$$\bar{w}_l = (\bar{t}_j - \bar{t}_i) \bmod \mathcal{T} \quad \forall n_i, n_j \in \mathcal{N}^{\kappa,\pi} \quad (8)$$

which is common for *PESP* and ensures that the *SSN* adheres to the requirements of periodic planning (Caimi *et al.*, 2017). At this stage, the *SSN* is incomplete and consists of weakly connected components. Each component stems from a $\mathcal{Z}_{\kappa,\pi}$.

Next, we add the $\mathcal{N}^{\text{station}}$ for each station individually. To accurately reflect changing between trains, we add \mathcal{T}/dt many n per station. Besides, we assign \bar{t}_n such that one n reflects one $t \in 0, dt, \dots, (\mathcal{T} - dt)$. Subsequently, we link these n with an $l \in \mathcal{L}^{\text{wait}}$ for each pair where $(\bar{t}_j - \bar{t}_i) \bmod \mathcal{T} = dt$. We then proceed to repeat this step for each station.

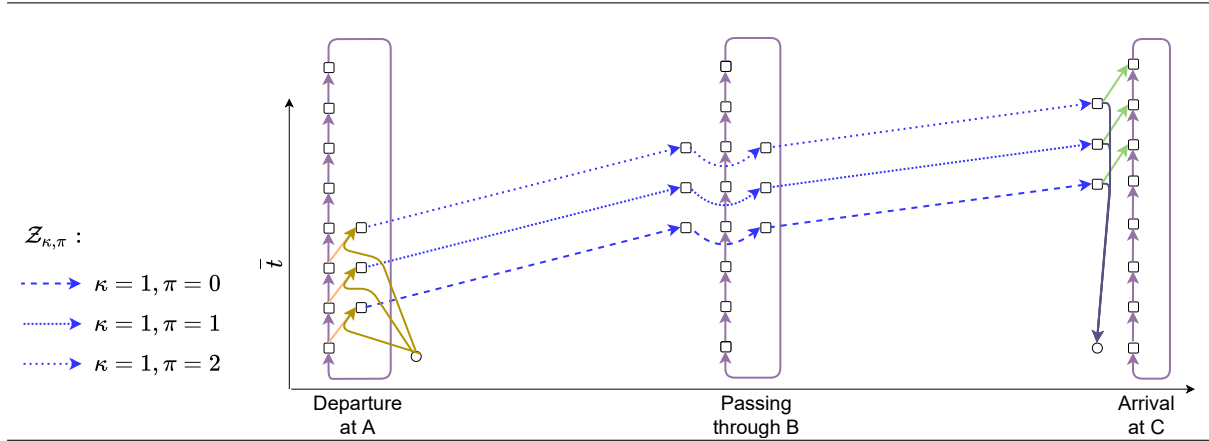
Next, we add $\mathcal{L}^{\text{egress}}$. We add one $l \in \mathcal{L}^{\text{egress}}$ for each $n \in \mathcal{N}^{\text{com}} \cap \mathcal{N}^{\text{arr}}$, such that it originates at n and connects it to an $n \in \mathcal{N}^{\text{change}}$ such that \bar{w}_l according to Eq. (8) equals μ^{egress} .

Adding $\mathcal{L}^{\text{access}}$ requires an additional consideration. Since a $z \in \mathcal{Z}_{\kappa,\pi}$ ties a \bar{t}_n to $t_{(n,e)}^{\text{max}}$, we always assume that the time window is fully utilised, even if the t_e of the corresponding e from the constrained (n, e) pair takes place at an earlier time. Hence, any \bar{w}_l for an $l \in \mathcal{L}^{\text{access}}$ connecting an $n \in \mathcal{N}^{\kappa,\pi}$ is given by $\mu^{\text{access}} + \mu_{\kappa}^{\text{comm}}$, as we otherwise violate the time windows of $\mathcal{Z}_{\kappa,\pi}$. Apart from that, adding $\mathcal{L}^{\text{access}}$ is similar to adding $\mathcal{L}^{\text{egress}}$, only difference being that an $l \in \mathcal{L}^{\text{egress}}$ originates at an $n \in \mathcal{N}^{\text{change}}$ and connects it with an $n \in \mathcal{N}^{\text{com}} \cap \mathcal{N}^{\text{dep}}$.

Lastly, we introduce $\mathcal{N}^{\text{entry}}$ and $\mathcal{N}^{\text{exit}}$ to the *SSN*. We add one $n \in \mathcal{N}^{\text{entry}}$ for each $o \in O$ and one $\mathcal{N}^{\text{exit}}$ for each $d \in D$ respectively. Note $\mathcal{N}^{\text{exit}} \cup \mathcal{N}^{\text{exit}}$ are auxiliary n acting as source and sink nodes. Hence, we omit to assign a \bar{t}_n for $\mathcal{N}^{\text{exit}} \cup \mathcal{N}^{\text{exit}}$. Eventually, we connect each $n \in \mathcal{N}^{\text{entry}}$ with an $l \in \mathcal{L}^{\text{access}}$ to all $n \in \mathcal{N}^{\text{dep}} \cap \mathcal{N}^{\text{com}}$, with $\bar{w}_l = \mu_{\text{access}}$ at this station. Similarly, we link each $n \in \mathcal{N}^{\text{exit}}$ with an $l \in \mathcal{L}^{\text{egress}}$ originating at the $n \in \mathcal{N}^{\text{arr}} \cap \mathcal{N}^{\text{com}}$ of this station. Naturally, $\bar{w}_l = \mu_{\text{egress}}$ holds for these links.

To support our explanation above, we present an illustration of an SSN in Fig. 8. In the example, $\mathcal{Y}_{\text{active}}$ contains three $\mathcal{Z}_{\kappa,\pi}$. Furthermore, there is one $n \in \mathcal{N}^{\text{entry}}$ at A, which is connected to all three departure nodes. Similarly, there is one $n \in \mathcal{N}^{\text{exit}}$ at C, connected to all commercial arrivals. At all three stations, we see the $n \in \mathcal{N}^{\text{change}}$ that are connected to a circle per station. Worth mentioning is that since the κ passes B, there is no connection from or to the $n \in \mathcal{N}^{\text{change}}$ at B. $l \in \mathcal{L}^{\text{access}}$ at A are yellow, while all $l \in \mathcal{L}^{\text{egress}}$ are coloured in dark blue at C.

Figure 8: Example SSN where based on the example line in Fig. 5.



As stated in Section 2.3 we aim to include both, the routing of passengers and vehicles in the timetable optimisation process. Likely, μ_{access} and μ_{egress} depend on the commodity. In this case, we propose to generate two SSN , one for the purpose of routing vehicles, denoted as SSN_{vehicle} and one for routing passengers SSN_{pax} . Furthermore, we do not include $\mathcal{N}^{\text{entry}}$ and $\mathcal{N}^{\text{exit}}$ in SSN_{vehicle} , as we demand all vehicles to circulate.

3.2.4 Generating the EAN

As mentioned in Section 3.2.1, the EAN is common in periodic $TTPs$. However, since we also include the task of assigning tracks in TTP , the EAN which we generate slightly deviates from usual notation. Besides, we add a special set of $\mathcal{E}^{\text{anchor}} \subseteq \mathcal{E}$ and $\mathcal{A}^{\text{anchor}} \subseteq \mathcal{A}$ to include the restrictions imposed by $\mathcal{Y}_{\text{active}}$. Similar to Fuchs *et al.* (2021), we create the EAN on a mesoscopic level of detail. This level is an aggregated representation of the microscopic level of detail, providing a more accurate representation than the common macroscopic level used for PESP (Caimi *et al.*, 2017; de Fabris *et al.*, 2014). As we are also considering the task of track assignment, we opt for the more detailed level of detail, which is still suitable for the strategic and tactical planning context (de Fabris *et al.*, 2014).

To create the EAN , we always require an $\mathcal{Y}_{\text{active}}$, hence all trains in the EAN must be operated. To construct the EAN , we extract the accessible mesoscopic infrastructure defined by each $\mathcal{Z}_{\kappa,\pi} \in \mathcal{Y}_{\text{active}}$. As $\mathcal{Z}_{\kappa,\pi}$ defines $\mathcal{E}^{\text{train}}$ in a successive sequence, the sequence of passed stations is given. Hence, we use the same procedure as Fuchs *et al.* (2021) for extracting the accessible mesoscopic infrastructure per $\mathcal{Z}_{\kappa,\pi}$. We convert the tracks to $\mathcal{A}^{\text{train}}$ by adding one $a \in \mathcal{A}^{\text{trip}}$ for

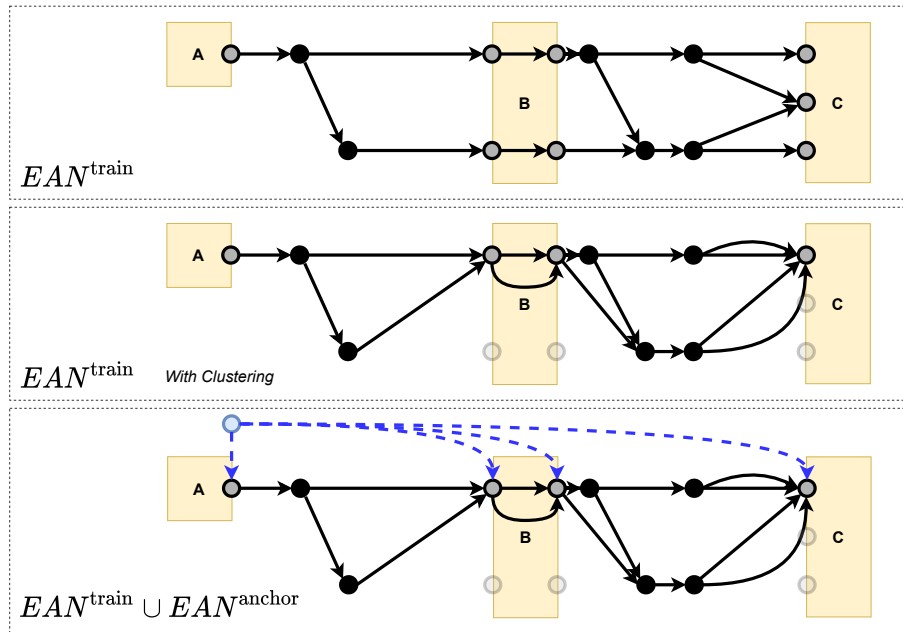
each section track and a $a \in \mathcal{A}^{\text{dwell}}$ at each station exit. Each $a \in \mathcal{A}^{\text{dwell}}$ originates at an $e \in \mathcal{E}^{\text{arr}}$ and leads to an \mathcal{E}^{dep} . On all remaining instances where the infrastructure indicates two or more connecting $a \in \mathcal{A}^{\text{train}}$, we add an $e \in \mathcal{E}^{\text{passing}}$. At the end of this step, the resulting EAN consists of $|\mathcal{Y}_{\text{active}}|$ weakly connected components. Each component is a **Directed Acyclic Graph (DAG)**. We provide an illustration for one such component in Fig. 9 at the EAN^{train} stage.

Next, as Fuchs *et al.* (2021) proposed, we cluster all $e \in \mathcal{E}^{\text{arr}}$ and $e \in \mathcal{E}^{\text{dep}}$ events at each station. This step is also illustrated in Fig. 9 at the second stage. On the one hand, this step allows us to remove redundant $e \in \mathcal{E}$. On the other hand, it enables us to introduce the restrictions imposed by $\mathcal{Z}_{\kappa,\pi} \in \mathcal{Y}_{\text{active}}$ as we now show.

To ensure that the resulting timetable does not violate the $[t_z^{\text{min}}, t_z^{\text{max}}]$ imposed by $z \in \mathcal{Z}_{\kappa,\pi}$ on the $e \in \mathcal{E}^{\text{arr}} \cup \mathcal{E}^{\text{dep}}$, we could enforce $t_e \in [t_z^{\text{min}}, t_z^{\text{max}}]$. However, in a case where $t_z^{\text{min}} \bmod \mathcal{T}$ is bigger than $t_z^{\text{max}} \bmod \mathcal{T}$, the resulting time window would be disjoint. Thus, we opt for an alternative approach, where introduce an $e \in \mathcal{E}^{\text{anchor}}$ per $\mathcal{Z}_{\kappa,\pi} \in \mathcal{Y}_{\text{active}}$ with $t_e = t_e^{\text{min}} = t_e^{\text{max}} = \pi$ for the $e \in \mathcal{E}^{\text{anchor}}$.

Next we connect each of these $e \in \mathcal{E}^{\text{anchor}}$ with all $e \in \mathcal{E}^{\text{arr}}$ and $e \in \mathcal{E}^{\text{dep}}$ that it should constrain according to the $z \in \mathcal{Z}_{\kappa,\pi}$. To add these constraining activities, we use $a \in \mathcal{A}^{\text{anchor}}$. As a consequence, each cluster of $e \in \mathcal{E}^{\text{arr}}$ and $e \in \mathcal{E}^{\text{dep}}$ is connected to the $e \in \mathcal{E}^{\text{anchor}}$.

Figure 9: The first three steps of the EAN creation for one $\mathcal{Z}_{\kappa,\pi}$.

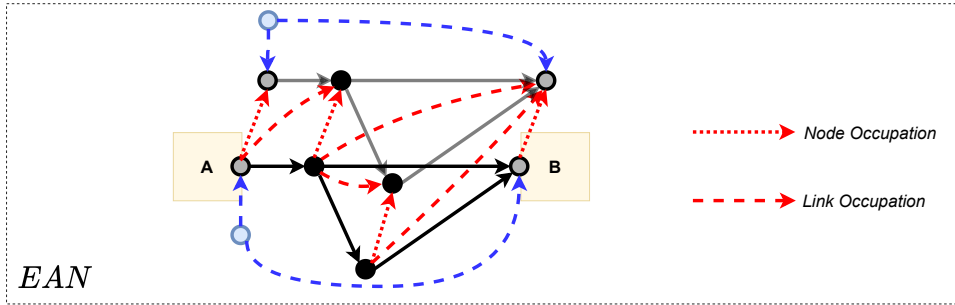


Then, we constrain the duration $w_a \in [w_a^{\text{min}}, w_a^{\text{max}}]$ for all these added $a \in \mathcal{A}^{\text{anchor}}$. As each $a \in \mathcal{A}^{\text{anchor}}$ represents a $z \in \mathcal{Z}_{\kappa,\pi}$, the time window is given by $[t_z^{\text{min}}, t_z^{\text{max}}]$ of the corresponding z . Thus, after introducing $\mathcal{E}^{\text{anchor}}$ and $\mathcal{A}^{\text{anchor}}$ to the EAN , any solution respects all $\mathcal{Z}_{\kappa,\pi} \in \mathcal{Y}_{\text{active}}$.

Fig. 9 provides visualisation of the steps so far.

Finally, we include all $\mathcal{A}^{\text{headway}}$ the timetable is free from collisions. Given our TTP is capable of assigning tracks, we introduce two types of $a \in \mathcal{A}^{\text{headway}}$, *node-occupation* and *link-occupation* $\mathcal{A}^{\text{headway}}$. While *node-occupation* $\mathcal{A}^{\text{headway}}$ ensure that two trains do not occupy the same infrastructure node without a safe separation interval, *link-occupation* $\mathcal{A}^{\text{headway}}$ ensure that trains are separated to avoid collision on a shared section. By explicitly distinguishing between *node-occupation* and *link-occupation*, we ensure that safety requirements are always met, without wasting capacity in the TTP. Fig. 10 provides an illustration for both, *node-occupation* and *link-occupation* $\mathcal{A}^{\text{headway}}$. To calculate $[w_a^{\min}, w_a^{\max}]$ for any w_a corresponding to an $a \in \mathcal{A}^{\text{headway}}$, we use the methods proposed by Peeters (2003).

Figure 10: An excerpt of an EAN for two $\mathcal{Z}_{\kappa, \pi}$ with headways.



Note that there are no \mathcal{A} that explicitly deal with commercial requirements ($|\mathcal{A}^{\text{com}}| = 0$), since any valid assignment of an $\mathcal{Y}_{\text{active}}$ already enforces any commercial requirements.

3.3 Introducing the Optimisation Module (OM)

Given the advances in research, a wide range of solution techniques for periodic timetables are available (Caimi *et al.*, 2017). However, the task itself remains challenging, especially when more aspects are integrated (Schiewe, 2020). Hence, we propose to decompose the **TTP** with logical Benders decomposition introduced by Hooker and Ottosson (2003). This decomposition scheme has proven to be appropriate for scheduling problems (Hooker, 2007).

We already introduced the concept of **TSSs**. Furthermore, all available **TSSs** are given by $\mathcal{Y}_{\text{available}}$. Thus, we now propose to find an optimal $\mathcal{Y}_{\text{active}} \subseteq \mathcal{Y}_{\text{available}}$ with the **Slot Sequence Selection Problem (SSSP)**. To cope with the complexity, we use logical Benders decomposition to decompose **Slot Sequence Selection Problem (SSSP)** into two hierarchically connected problems, the **Master Slot Sequence Selection Problem (M-SSSP)** and the **Sub Slot Sequence Selection Problem (S-SSSP)**. While the master problem, **Master Slot Sequence Selection Problem (M-SSSP)** uses the **Slot Sequence Network (SSN)** for finding the optimal combination of $\mathcal{Z}_{\kappa,\pi} \in \mathcal{Y}_{\text{active}} \subseteq \mathcal{Y}_{\text{available}}$. For a given $\mathcal{Y}_{\text{active}}$ the **S-SSSP** utilises the **EAN** to find a timetable for a given $\mathcal{Y}_{\text{active}}$, such that all trains can run within their $\mathcal{Z}_{\kappa,\pi}$ without causing a conflict. If the **S-SSSP** is infeasible for $\mathcal{Y}_{\text{active}}$, it determines the conflicting $\mathcal{Z}_{\kappa,\pi}$ given by $\mathcal{Y}_{\text{conflict}} \subseteq \mathcal{Y}_{\text{active}}$, hence the current $\mathcal{Y}_{\text{active}}$ and any other possible $\mathcal{Y}_{\text{active}}$ containing $\mathcal{Y}_{\text{conflict}}$ are removed in the **M-SSSP**.

3.3.1 The Master Slot Sequence Selection Problem (M-SSSP)

To formulate the **M-SSSP**, we define the decision variables p_l^{od} , and g_l^r . Each of these variables represents a flow that is routed on $l \in \mathcal{L}$ of the respective network. While all p_l^{od} are routed via the SSN_{pax} , the SSN_{vehicle} provides the network to route the flow of vehicles g_l^r . Eventually, we introduce the binary decision variable $\delta_{\kappa,\pi}$, which indicates if the corresponding $\mathcal{Z}_{\kappa,\pi}$ is selected. Note that we use $\text{deg}(n)^+$ to obtain all outgoing $l \in \mathcal{L}$ for a given n , while $\text{deg}(n)^-$ yields all incoming $l \in \mathcal{L}$.

Furthermore, we must ensure that the commodities are routed according to the *od*-relation, so we use a copy of the $SSN_{\text{pax},od}$ per $od \in \mathcal{OD}$. The same applies for each vehicle r ; hence we also use a copy of $SSN_{\text{vehicle},r}$ for each $r \in \mathcal{R}$.

$$\min \sum_{od \in \mathcal{OD}} \sum_{l \in \mathcal{L}_{od}} c_l^{od} \cdot \bar{w}_l \cdot p_l^{od} \quad (9a)$$

$$\text{sb. to } \sum_{l \in \text{deg}^-(n)} p_l^{od} - \sum_{l \in \text{deg}^+(n)} p_l^{od} \begin{cases} = \gamma^{od} & \text{if } n \in \mathcal{N}^{\text{entry}} \\ = -\gamma^{od} & \text{if } n \in \mathcal{N}^{\text{exit}} \\ = 0 & \text{else} \end{cases}, \quad \forall n \in \mathcal{N}_{\text{pax},od}, \forall od \in \mathcal{OD}, \quad (9b)$$

$$\sum_{l \in \text{deg}^-(n)} g_l^r - \sum_{l \in \text{deg}^+(n)} g_l^r = 0 \quad \forall n \in \mathcal{N}_r, \forall r \in \mathcal{R}, \quad (9c)$$

$$\sum_{od \in \mathcal{OD}} p_l^{od} \leq g_l^r \cdot \text{capacity}(r), \quad \forall l \in \mathcal{L}_{\kappa,\pi}, \forall \pi \in \Pi_\kappa, \forall \kappa \in \mathcal{K}, \forall r \in \mathcal{R}, \quad (9d)$$

$$\sum_{l \in \mathcal{L}_r} g_l^r \cdot \bar{w}_l = y^r \cdot T, \quad \forall r \in \mathcal{R}, \quad (9e)$$

$$g_l^r = \delta_{\kappa,\pi}, \quad \forall l \in \mathcal{L}_{\kappa,\pi}, \quad \forall \pi \in \Pi_\kappa, \quad \forall \kappa \in \mathcal{K}, \quad (9f)$$

$$\sum_{\pi \in \Pi_\kappa} \delta_{\kappa,\pi} \begin{cases} = f_\kappa & \text{if } \kappa \text{ is mandatory} \\ \leq f_\kappa & \text{else} \end{cases} \quad \forall \kappa \in \mathcal{K}, \quad (9g)$$

$$\sum_{(\kappa,\pi) \in \mathcal{Y}_{\text{conflict}}} \delta_{\kappa,\pi} \leq |\mathcal{Y}_{\text{conflict}}| - 1, \quad (9h)$$

$$\delta_{\kappa,\pi} \in \{0, 1\}, \quad \forall \pi \in \Pi_\kappa, \quad \forall \kappa \in \mathcal{K}, \quad (9i)$$

$$p_l^{od} \in \mathbb{R}^+, \quad \forall l \in \mathcal{L}_{\text{pax},od}, \quad \forall od \in \mathcal{OD}, \quad (9j)$$

$$g_l^r \in \mathbb{Z}^+, \quad \forall l \in \mathcal{L}_{r,\text{vehicle}}, \quad \forall r \in \mathcal{R}, \quad (9k)$$

$$y^r \in \{0, 1, \dots, \max(r)\}, \quad \forall r \in \mathcal{R}. \quad (9l)$$

The objective in Eq. (9a) minimizes the generalised travel time as defined by C in Section 3.1.3. Hence, the objective of an optimal solutions corresponds to C_{nominal} , based on generalised travel time in Eq. (1). Furthermore, flow conservation in each $SSN_{\text{pax},od}$ for $od \in \mathcal{OD}$ is ensured by Eq. (9b). Likewise, the flow balance for vehicles in each $SSN_{\text{vehicle},r}$ for each $r \in \mathcal{R}$ is enforced by Eq. (9c). Joining vehicles and passengers, Eq. (9d) ensures that the accumulated passenger flow over all $od \in \mathcal{OD}$ per $l \in \mathcal{L}^{\text{train}}$ does not exceed the capacity of the g_l^r operating on l .

Eq. (9e) limits the number of available vehicles and requires the total number of used vehicles to be an integral multiple of the period duration. Besides, Eq. (9f) ensures that a vehicle g_l^r may only be routed via an l of a **TSS** if the corresponding $\delta_{\kappa,\pi}$ of the **TSS** is chosen to be active.

As all variables $\delta_{\kappa,\pi}$ belong to a directed line κ , Eq. (9g) assures that the number of active $\delta_{\kappa,\pi}$ for all $\pi \in \Pi_\kappa$ does not exceed the frequency f_κ . Furthermore, if κ must be operated, we enforce that the number of active $\delta_{\kappa,\pi}$ equals f_κ . Note that similar to Polinder *et al.* (2021), we do not enforce regularity constraints if $f_\kappa > 1$ and instead leave the decision on the separation between the repetitions to the **M-SSSP**.

Eq. (9h) is the constraint that we use to couple the **M-SSSP** with the **S-SSSP** in the Logic-based Benders decomposition. It allows for banning any combination of $\delta_{\kappa,\pi}$ for which the **S-SSSP** could not find a solution due to conflicts.

3.3.2 Addressing a Critical Disruption in the M-SSSP

Following the establishment of the M-SSSP we propose an approach to increase the resilience of a given M-SSSP solution against a known critical disruption. Note that we explain how we obtain the critical disruption throughout the two subsequent sections. For the adaption, we require the critical disruption given by $\mathcal{H}_{\text{critical}}$ and the current solution given by $\mathcal{Y}_{\text{active}}$. We aim to provide redundant connections by pushing some of the $p^{od} \in \mathcal{OD}$ on redundant, less optimal connections. Then, if we resolve the M-SSSP, these connections benefit from a higher demand. As a consequence, they likely offer alternative, high-quality connections. Since these connections are redundant in normal circumstances, we refer to them as *redundant*. As demanding redundant connections may lead to a worse C_{nominal} and increase the *Cost of Resilience* (Section 3.1.3), we propose to apply this adaptation measure only for critical disruptions, i.e. the disruptions with most impact on performance P .

Let $\mathcal{OD}_{\text{critical}}$ be the set of all od affected by $\mathcal{H}_{\text{critical}}$. Furthermore, let $l \in \mathcal{L}_{\mathcal{H},od}$ be all the links $l \in \mathcal{L}^{\text{train}}$ for one od affected by $\mathcal{H}_{\text{critical}}$. Then, we may identify all p_l^{od} currently affected by $\mathcal{H}_{\text{critical}}$. However, before we can impose a limit for each od , we have to capture the share of p_l^{od} that can be rerouted, since only parts of the passenger flow might be capable of reaching their destination via a redundant connection.

To obtain the share of γ^{od} that is available, we restrict the M-SSSP to the current $\mathcal{Y}_{\text{active}}$. Subsequently, we iterate over all $od \in \mathcal{OD}_{\text{critical}}$ by setting the corresponding objective as defined by Eq. (10) in the M-SSSP.

$$p_{\mathcal{H}}^{od,\min} = \min \sum_{l \in \mathcal{L}_{\mathcal{H}}} p_l^{od} \quad \forall od \in \mathcal{OD}_{\text{critical}} \quad (10)$$

Consequently, we obtain the minimal flow required to still ensure feasibility of the M-SSSP for each $od \in \mathcal{OD}_{\text{critical}}$. Let $p_{\mathcal{H}}^{od,\min}$ be the minimal cumulated flow, while $p_{\mathcal{H}}^{od,\max}$ in Eq. (11) describes the current cumulated flow of p^{od} for an od via $\mathcal{H}_{\text{critical}}$.

$$p_{\mathcal{H}}^{od,\max} = \sum_{l \in \mathcal{L}_{\mathcal{H}}} p_l^{od} \quad \forall od \in \mathcal{OD}_{\text{critical}} \quad (11)$$

Based on a user defined parameter $\lambda \in (0, 1]$ we force a share of passengers to choose a redundant route by adding Eq. (12) to the M-SSSP.

$$\sum_{l \in \mathcal{L}_{\mathcal{H}}} p_l^{od} \leq (p_{\mathcal{H}}^{od,\max} - p_{\mathcal{H}}^{od,\min}) \cdot \lambda \quad \forall od \in \mathcal{OD}_{\text{critical}} \quad (12)$$

As mentioned before, we aim to add these constraints such that the vulnerability of the next $\mathcal{Y}_{\text{active}}$ against the current $\mathcal{H}_{\text{critical}}$ is reduced. Hence, we proceed to integrate the proposition in an iterative approach, allowing us to iteratively improve a given $\mathcal{Y}_{\text{active}}$.

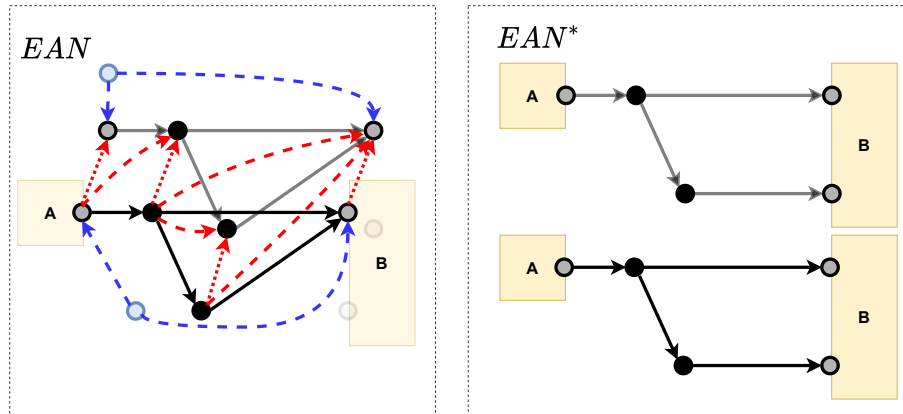
3.3.3 The Sub Slot Sequence Selection Problem (S-SSSP)

As stated in Section 3.3, it is the task **S-SSSP** to verify if for a given $\mathcal{Y}_{\text{active}}$, a feasible timetable exists. While the **S-SSSP** can exploit the routing alternatives by assigning tracks, any timetable has to respect the time windows induced by the $z \in \mathcal{Z}_{\kappa,\pi}$ for all $\mathcal{Z}_{\kappa,\pi} \in \mathcal{Y}_{\text{active}}$.

Given a $\mathcal{Y}_{\text{active}}$, we use the procedure outlined in Section 3.2.4 to generate the input *EAN*. Next, we create the **Train Flow Network (TFN)**, which is required to assign tracks. The *TFN* consists of vertices $v \in \mathcal{V}$ and segments $u \in \mathcal{U}$. To create the *TFN*, we can use the non-clustered *EAN* as input. Initially, we prune this copy of the *EAN* such that it solely consists of $e \in \mathcal{E}^{\text{train}}$ and $a \in \mathcal{A}^{\text{train}}$. Let *EAN** be one such pruned copy. We then convert *EAN** to the *TFN* by mapping each $e^* \in \mathcal{E}^*$ to one $v \in \mathcal{V}$. Similarly, we can map $a^* \in \mathcal{A}^*$ to $u \in \mathcal{U}$.

Next, we define $v \in \{0,1\}$ and $u \in \{0,1\}$, such that we can use the *TFN* as a directed binary flow network. In the *TFN*, a flow corresponds to a train route. Furthermore, *TFN* consists of $|\mathcal{Y}_{\text{active}}|$ many weakly connected components. If we induce one binary flow from the sources $v \in \mathcal{V}_{\text{source}}$ to sinks $v \in \mathcal{V}_{\text{sink}}$ in each of these components, we obtain a route for each train in the corresponding *EAN*. Subsequently, the *TFN* allows us to assign tracks, while we can use the *EAN* to assign $t_e \forall e \in \mathcal{E}$. Figure 11 provides an exemplary *EAN* and *TFN* for a case with two trains to route.

Figure 11: Example *EAN* and *TFN* for two $\mathcal{Z}_{\kappa,\pi} \in \mathcal{Y}_{\text{active}}$.



To allow joint routing and timetabling in **TTP**, Fuchs *et al.* (2021) introduce the **TTP** with *selectable* and *non-selectable* activities. However, in our formulation, all $a \in \mathcal{A}$ are *selectable*. Similar to Fuchs *et al.* (2021), we define *selectable* as that $w_a \in [w_a^{\min}, w_a^{\max}]$ must hold, when a is *selected*. Otherwise $w_a \in \mathbb{R}^+$ implies that the a is inactive. Whether an a is *selected* depends on the routes chosen in *TFN*. For example, an $a \in \mathcal{A}^{\text{headway}}$ is superfluous, if the two separated trains use different infrastructure. Since a collision is impossible, a might actually over-constrain the **S-SSSP** and lead to a wrong result. Hence, we define that a *selectable* a is active when all required vertices $v \in \mathcal{V}_a$ and segments $u \in \mathcal{U}_a$ are used by trains routed through *TFN*.

To define the \mathcal{V}_a and \mathcal{U}_a for each $a \in \mathcal{A}$, we have to consider its purpose. If $a \in \mathcal{A}^{\text{train}}$, the activity has to be selected if the corresponding train uses it, thus \mathcal{U}_a holds the corresponding u while $\mathcal{V}_a = \{\}$. For an $a \in \mathcal{A}^{\text{headway}}$ that is of type *node-occupation*, \mathcal{V}_a contains the two corresponding vertices in the *TFN*, that have to be separated safely if both are used, while $\mathcal{U}_a = \{\}$. Contrarily, if an $a \in \mathcal{A}^{\text{headway}}$ is of type *link-occupation*, \mathcal{U}_a contains both u that have to be separated safely, while $\mathcal{V}_a = \{\}$. For any other $a \in \mathcal{A}$, $\mathcal{V}_a = \mathcal{U}_a = \{\}$

Recall that w_a denotes the duration of a *selectable* $a \in \mathcal{A}$ and $w_a \in [w_a^{\min}, w_a^{\max}]$ must hold if the activity is *selected*. Eq. (13) and Eq. (14) show how we model this requirement.

$$w_a \geq w_a^{\min} - (|\mathcal{V}_a| - \sum_{v \in \mathcal{V}_a} v + |\mathcal{U}_a| - \sum_{u \in \mathcal{U}_a} u) \cdot w_a^{\min} \quad (13)$$

$$w_a \leq w_a^{\max} + (|\mathcal{V}_a| - \sum_{v \in \mathcal{V}_a} v + |\mathcal{U}_a| - \sum_{u \in \mathcal{U}_a} u) \cdot (\mathcal{T} - w_a^{\max} - dt) \quad (14)$$

Eq. (13) restricts the activity duration w_a to the lower bound w_a^{\min} , when all related vertices \mathcal{V}_a and segments \mathcal{U}_a are used by routed trains. The lower bound is relaxed if at least one v or u is unused ($= 0$). The structure of Eq. (14) is similar while it constrains the upper bound. Note, that for an activity which has to be valid at all time (e.g., an $a \in \mathcal{A}^{\text{anchor}}$), $|\mathcal{V}_a| = 0$ and $|\mathcal{U}_a| = 0$ holds, implying that w_a^{\min} and w_a^{\max} must be respected in any case. Hence, there is no need to explicitly distinguish between *selectable* and *non-selectable* activities, as Fuchs *et al.* (2021) do.

By using the decision variables t'_e to model t_e , k_a to account for periodicity, and v, u to assign tracks, we may formulate the **S-SSSP** with routing now in (15):

$$\min \sum_{a \in \mathcal{A}^{\text{anchor}}} w_a \quad (15a)$$

$$\text{s. t. } w_a \geq w_a^{\min} - (|\mathcal{V}_a| - \sum_{v \in \mathcal{V}_a} v + |\mathcal{U}_a| - \sum_{u \in \mathcal{U}_a} u) \cdot w_a^{\min}, \quad \forall a \in \mathcal{A}, \quad (15b)$$

$$w_a \leq w_a^{\max} + (|\mathcal{V}_a| - \sum_{v \in \mathcal{V}_a} v + |\mathcal{U}_a| - \sum_{u \in \mathcal{U}_a} u) \cdot (\mathcal{T} - w_a^{\max} - dt), \quad \forall a \in \mathcal{A}, \quad (15c)$$

$$w_a = (t'_j - t'_i) \cdot dt + k_{i,j} \cdot \mathcal{T}, \quad \forall (i, j) \in \mathcal{A}, \quad (15d)$$

$$\sum_{u_i \in \text{deg}^+(v)} u_i = \sum_{u_j \in \text{deg}^-(v)} u_j = v, \quad \forall v \in \mathcal{V}, \quad (15e)$$

$$\sum_{v \in \mathcal{V}_{\text{source}} \cap \mathcal{V}_{\kappa, \pi}} v = 1, \quad \forall (\kappa, \pi) \in \mathcal{Y}_{\text{active}}, \quad (15f)$$

$$t'_e \in \{0, 1, \dots, (\mathcal{T} - dt)/dt\}, \quad \forall e \in \mathcal{E}, \quad (15g)$$

$$v \in \{0, 1\}, \quad \forall v \in \mathcal{V}, \quad (15h)$$

$$a \in \{0, 1\}, \quad \forall u \in \mathcal{U}, \quad (15i)$$

$$k_a \in \mathbf{Z}, \quad \forall a \in \mathcal{A} \quad (15j)$$

Although the **S-SSSP** is only required to determine feasibility, we define the objective in Eq. (15a) to minimise the total duration of all activities that connect the anchor events with the clustered arrival and departure events. Thus, we aim to guide the solver when searching for a solution. Eq. (15b) and Eq. (15c) ensure that if an activity is selected, the minimal and maximal duration is not exceeded ($w_a \in [w_a^{\min}, w_a^{\max}]$). Eq. (15d) is a typical **PESP**-constraint, which links the duration of activity with the two associated events and corrects the effect of periodicity. Even though, w_a is shown here as here as a variable, we can substitute any occurrence of w_a with Eq. (15d).

Eq. (15e) ensures flow conservation in the **TFN** components and connects the flow passing through v with the incoming and outgoing flow u . Eq. (15f) demands that one route per $\mathcal{Z}_{\kappa,\pi} \in \mathcal{Y}_{\text{active}}$ is active since the flow for each set of vertices belonging to the same $\mathcal{V}_{\text{source}}$ has to equal one. Finally, a scaled timestamp for an $e \in \mathcal{E}$ given by t'_e is limited to the scaled period duration (Eq. (15g)). Note that we have to scale t'_e such that $t_e = dt \cdot t'_e$, in order for $t_e \in \{0, dt, \dots, (\mathcal{T} - dt)\}$.

Since **TFN** is a binary flow network, all flow variables are required to be binary (Eq. (15i) and Eq. (15h)). Finally, k_a is restricted to be integer Eq. (15j). Note that one can always adjust w_a such that $k_a \in \{0, 1\}$, as shown by Peeters (2003). However, since we do not use the **MIP** formulation of the **S-SSSP**, we omit strengthening the bounds on k_a .

3.3.4 Encoding the S-SSSP with Boolean Satisfiability Problem (SAT)

We do not solve the **MIP** formulation of the **S-SSSP** but instead encode it in **SAT**. We opt for **SAT** encoding because of three reasons. Firstly, in periodic **TTP**, **SAT** is known to outperform the **MIP** formulation when determining feasibility (Kümmling *et al.*, 2015), which is the main task of the **S-SSSP**. Secondly, another task of the **S-SSSP** is to provide the master with the cause of infeasibility. As some **SAT**-solvers provide unsatisfiable cores, we may use this information to extract information on the conflicting $\mathcal{Z}_{\kappa,\pi} \in \mathcal{Y}_{\text{conflict}} \subseteq \mathcal{Y}_{\text{active}}$. Finally, as we also want include the task of train routing when solving the **S-SSSP**, we follow Fuchs *et al.* (2021), which demonstrate that encoding as **SAT** generally outperforms equivalent **MIP**-formulations.

As a result of the encoding, we obtain an equivalent formula in a **Conjunctive Normal Form (CNF)**, where clauses represent a disjunction of literals, with q denoting a literal. A **CNF** is satisfactory if at least one valid assignment for all $q \in \mathcal{Q}$ exists. We refer to Biere *et al.* (2009) for further information on **SAT**. In the following, we proceed to encode the **S-SSSP**. Note that the presented encoding is based on the encoding by Fuchs *et al.* (2021).

To begin with, we encode the **TFN** as a binary flow problem. Given that each component in the **TFN** represents the routing options for one train path, we can map the vertices \mathcal{V} and edges \mathcal{U} of the **TFN** directly to **SAT** literals $q \in \mathcal{Q}$ by a one-to-one mapping. Using Eq. (16) as a helper,

we encode the *TFN* by applying 17 to each $v \in \mathcal{V}$.

$$\text{encode-at-most-one}(\mathcal{Q}_{ss}) := \bigwedge_{\forall q_i, q_j \in \mathcal{Q}_{ss}, i < j} \neg q_i \vee \neg q_j \quad (16)$$

$$\begin{aligned} \text{encode-}TFN(v) := & (\neg q_v \wedge \bigvee_{u \in \text{deg}^+(v)} q_u) \wedge (\neg q_v \wedge \bigvee_{u \in \text{deg}^-(v)} q_u) \\ & \wedge \text{encode-at-most-one}(\text{deg}^+(v)) \\ & \wedge \text{encode-at-most-one}(\text{deg}^-(v)) \end{aligned} \quad (17)$$

Following the encoding of the *TFN*, we proceed to encode the *EAN*. At first, we map one literal to each $t_e \in \{0, dt, \dots, \mathcal{T}\}$ for all $e \in \mathcal{E}$. Subsequently, we order-encode all events \mathcal{E} as proposed by Großmann (2011) with (18).

$$\text{encode-event}(e) := (\neg q_{e, -dt} \wedge q_{e, \mathcal{T} - dt}) \wedge \bigwedge_{i \in \{0, dt, \dots, \mathcal{T} - dt\}} (\neg q_{e, i - dt} \vee q_{e, i}) \quad (18)$$

In a successive step, we encode $a \in \mathcal{A}$. As we are using *selectable* \mathcal{A} , the limitations on the duration w_a only have to be respected when the according vertices \mathcal{V}_a and segments \mathcal{U}_a in the *TFN* are used by the routed trains. Thus, we adapt the encode-rectangle function of Großmann (2011) towards the encode-conditional-rectangle (19).

$$\begin{aligned} \text{encode-conditional-rectangle}(\{i_1, i_2\} \times \{j_1, j_2\}) := & \neg q_{i, i_2} \vee q_{i, i_1} \vee \neg q_{j, j_2} \vee q_{j, j_1} \\ & \vee \bigvee_{v \in \mathcal{V}_a} \neg q_v \vee \bigvee_{u \in \mathcal{U}_a} \neg q_u \end{aligned} \quad (19)$$

We demand one path for each *TFN* component as the last encoding step. Hence, we add one assumption literal per $\mathcal{Z}_{\kappa, \pi} \in \mathcal{Y}_{\text{active}}$, denoted by $q_{\kappa, \pi}$. We may now link $q_{\kappa, \pi}$ with the corresponding source vertices $v \in \mathcal{V}_{\text{source}} \cap \mathcal{V}_{\kappa, \pi}$ with Eq. (20). Hereby, we encode the requirement to select a paths for trains in the *TFN*.

$$\text{encode-path-activation}(\kappa, \pi), \mathcal{Q}_{\text{source}} := \neg q_{\kappa, \pi} \vee \bigvee_{v \in (\mathcal{U}_{\text{source}} \cap \mathcal{U}_{\kappa, \pi})} q_v \quad (20)$$

Thereby, we can demand the $q_{\kappa, \pi}$ to be true as an assumption literal to enforce the selection of path in the *TFN* component. Furthermore, in the case of infeasibility, all $q_{\kappa, \pi}$ that are part of the unsatisfiable core are the conflicting components of the *TFN*. Extracting $\mathcal{Y}_{\text{conflict}}$ from the unsatisfiable core is thus straightforward. Consequently, we can use the *SAT* approach to find a solution for the *S-SSSP* or to extract $\mathcal{Y}_{\text{conflict}} \subseteq \mathcal{Y}_{\text{active}}$.

3.3.5 Connecting the M-SSSP and S-SSSP in the Optimisation Module (OM)

We expect the *M-SSSP* to yield many solution candidates $\mathcal{Y}_{\text{active}}^*$ that have to be assessed by the *S-SSSP*. On the other hand, it is possible to use the *S-SSSP* to find a feasible solution, which

we can transform towards \mathcal{Y}_{active}^- as a potential solution to the **M-SSSP**. Thus, we use a callback, such that the **M-SSSP** can interact with the **S-SSSP** when being solved.

Using the callback yields several advantages. At first, the **MIP**-solver can assess solutions during the branch and bound tree exploration. Consequently, if the **S-SSSP** finds the current solution infeasible, we add a constraint as given by Eq. (9h) to the **M-SSSP** to remove this conflicting \mathcal{Y}_{active}^* . Contrarily, if we solved the **M-SSSP** to optimality, we would have to re-start the solver when the solution is not feasible in the **S-SSSP**.

Our callback assesses complete integral \mathcal{Y}_{active}^* . If for any κ which is mandatory, the count of active $\mathcal{Z}_{\kappa,\pi}$ is less than f_κ , the callback forces to **S-SSSP** to find at least one \mathcal{Y}_{active} , where sufficient $\mathcal{Z}_{\kappa,\pi}$ are active for all $\kappa \in \mathcal{K}$. This has two advantages. Firstly, we are sure that even for a partially relaxed solution of the **M-SSSP**, at least one valid \mathcal{Y}_{active} for the **S-SSSP** exists. Secondly, we can use the **S-SSSP** to generate a potential solution candidate \mathcal{Y}_{active}^- for the **M-SSSP**. However, randomly obtaining and injecting \mathcal{Y}_{active}^- is likely not beneficial. Hence, we limit the possibility of generating \mathcal{Y}_{active}^- to warm-start the **M-SSSP**. Note that such a \mathcal{Y}_{active}^- does not necessarily lead to an improved incumbent solution in the **M-SSSP**.

3.3.6 A Fix-and-Dive-Heuristic for the Slot Sequence Selection Problem (SSSP)

Heuristics that optimise a partition of an instance while relaxing or fixing the remainder have been applied successfully in strategic **TTP** (Herrigel *et al.*, 2018; Polinder, 2020) and disruption management (Zhu and Goverde, 2020b). Consequently, we propose a similar approach to create high-quality solutions from scratch. In essence we relax a selection of variables of the **M-SSSP**. Then we iteratively fix some of the variables and dive down the branch-and-bound tree. Our approach relaxes variables from $\delta_{\kappa,\pi} \in \{0, 1\}$ to $\delta_{\kappa,\pi} \in [0, 1]$ and groups them according to the corresponding κ . Let Δ_κ be one such a group, containing all $\delta_{\kappa,\pi}$ for one κ . As a feature, the heuristic allows sorting the sequence of Δ_κ by user preference.

As stated previously, we relax all $\delta_{\kappa,\pi}$ that are part of any κ that should be relaxed. We then proceed to demand all $\delta_{\kappa,\pi} \in \{0, 1\} \forall \delta_{\kappa,\pi} \in \Delta_\kappa$ for the first κ and solve the partially relaxed **M-SSSP** to optimality or until a time limit is reached. Similar to Zhu and Goverde (2020b) there is one exception. If no feasible solution can be found within the time limit, we allow to violate the time limit until a feasible solution has been found. Given a solution, we then fix all Δ_κ that are binary to their current value and proceed to the next κ . After iterating through all the groups of $\delta_{\kappa,\pi}$, we obtain an $\mathcal{Y}_{initial}$ for the **M-SSSP**. Note that due to the callback procedure in Section 3.3.5, we are assured that at least one feasible solution for the **S-SSSP** exists, even when all $\delta_{\kappa,\pi} \forall \mathcal{Z} \in \mathcal{Y}_{available}$ in the **M-SSSP** are relaxed to non-binary values.

3.3.7 Shrinking the size of the M-SSSP

The routing of passengers in the **M-SSSP** is arc-based. Since this formulation requires $|OD|$ many copies of the SSN_{pax} , the problem size is likely to increase quickly. However, opting for a

path based formulation similar to [Szymula and Bešinović \(2020\)](#) is not viable since we solve the **SSSP** *from scratch*, rather than optimising an existing timetable. Thus, we keep the arc-based formulation but propose three measures to shrink the instance size. All steps aim to reduce the number SSN_{pax} elements in the formulation but none affects the optimality of the **SSSP**.

In the first step, we remove all $n \in \mathcal{N}^{\text{train}} \cap \mathcal{N}^{\text{n-com}}$, since these nodes are irrelevant to route the passengers. Naturally, we have to add one $l \in \mathcal{L}^{\text{train}}$ to bridge the gap that results from removing the n . Let l_i be the incoming link and l_j be the outgoing of a removed n . We then have to add one l with $\bar{w}_l = \bar{w}_i + \bar{w}_j$, which connects the predecessor n with the succeeding n .

In the next step, we utilise a frequent assumption in (strategic) passenger-oriented planning, which states that the maximal detour for an od can be limited ([Schmidt and Schöbel, 2015a](#)). This assumption is based on the observation that all cut-off routing alternatives would lead to unacceptable detours for this od -relation. In our case, we may apply this assumption without loss of optimality. This is due to our limit on the maximal detour in the **SSSP**, c_{max}^{od} per od given in Eq. (2). Thus, cutting-off any worse detours does not affect the **SSSP**. Consequently, we prune each $SSN_{\text{pax},od}$. We remove all nodes $n \in SSN_{\text{pax},od}$ where the generalised travel time of the shortest path from o to d via n exceeds c_{max}^{od} .

In the second step, we use the observation that although two od -relations do not share o and d , the two copies of SSN_{pax} might be highly similar, especially if the respective o and d are located closely. Hence we introduce a second reduction step based on the suggestion of [Bull et al. \(2019\)](#) to bundle the flow commodities of a line plan by origin, rather than by relation.

To expand on [Bull et al. \(2019\)](#), we implement multiple bundling procedures, such that we can use the one which produces the smallest **SSSP** instance: **Bundling by Origin** follows the suggestion of [Bull et al. \(2019\)](#), such that all outgoing p_{od} for one o use the same $SSN_{\text{pax},o}$. **Bundling by Destination** is the inverse strategy of *Bundling by Origin* by [Bull et al. \(2019\)](#), such that all incoming p_{od} for one d use the same $SSN_{\text{pax},d}$. **Heuristic Bundling** uses a **MIP** heuristic to find the best bundling. It uses the number of required SSN_{pax} links as objective and selects a subset of possible bundling such that all $od \in OD$ are served. However, as per origin or destination bundling $\mathcal{OD}_{\text{bundle}}$, $\binom{|\mathcal{OD}_{\text{bundle}}|}{k}$ possible bundling configurations exist, $O(|\mathcal{OD}_{\text{bundle}}|!)$ of variables would be required. However, we limit each bundle to a maximum 1000 combinations. Thus, the subsequent **MIP** yields only heuristic results.

Eventually we want to emphasise that the outlined procedures to reduce the size of the **SSSP** do not lead to any loss of optimality. Firstly, removing $n \in \mathcal{N}^{\text{train}} \cup \mathcal{N}^{\text{n-com}}$ from the $SSN_{\text{pax},od}$ does not remove any routing options and thus does not affect optimality. Secondly, pruning the $SSN_{\text{pax},od}$ with c_{max}^{od} does only remove paths, which would lead to a $C > C_{\text{worst-case}}$ as the length of any removed path exceeds c_{max}^{od} . Furthermore, bundling od by o or d does not affect optimality as well ([Bull et al., 2019](#)).

3.4 Introducing the Assessment Module (AM)

In contrary to the OM, the AM focuses on assessing an existing $\mathcal{Y}_{\text{active}}$, rather than finding one. Furthermore, the AM aims to find $\mathcal{H}_{\text{critical}}$ from all possible disruptions \mathcal{H} , even when all responses are optimally applied.

Currently, models like the RNVM proposed by Szymula and Bešinović (2020) already allow finding $\mathcal{H}_{\text{critical}}$. Furthermore, approaches to manage known disruptions exist as well (Borecka and Bešinović, 2021; Zhu and Goverde, 2017). However, none of the approaches jointly considers the task of identifying $\mathcal{H}_{\text{critical}}$ and applying responses. Hence, we propose two adaptations of the SSSP, the Primal Slot Sequence Selection Problem (Primal-SSSP) and the Dual Slot Sequence Selection Problem (Dual-SSSP). While the former model is capable of assessing and responding to the impact of a given disruption, the latter identifies $\mathcal{H}_{\text{critical}}$.

Subsequently, we combine the two models in an algorithm, capable of identifying the critical disruption even when the best of the available responses actions are taken. Furthermore, by relying on modifications of the SSSP consistency between the AM and OM is ensured, which is key for an accurate assessment. On the contrary, using a different model could lead to loss of information.

3.4.1 Modifying the SSSP towards Primal Slot Sequence Selection Problem (Primal-SSSP)

Initially, we adapt the SSSP to find the best responses as reaction to a given disruption containing one or more blocked $h \in \mathcal{H}_{\text{disrupted}}$. Recall, that we start from an existing $\mathcal{Y}_{\text{active}}$, as a starting solution.

To adapt the SSSP towards Primal-SSSP for such an analysis, it is crucial to add a bypass $l \in \mathcal{L}^{\text{bypass}}$ from each o to d given by \mathcal{OD} , as some $od \in \mathcal{OD}$ may have no more routes available. Note that we add $l \in \mathcal{L}^{\text{bypass}}$ in each $SSN_{\text{pax},od}$. For the bypasses, we set the $\bar{w}_l = 1$ while $c_l = c_{\text{max}}^{od}$ reflects the penalty. As stated in Section 3.1.3 the penalty c_{max}^{od} contains the generalised travel time of the longest path permissible for this od . Thus, completely cutting off an od -relation increases C and cutting off all $od \in \mathcal{OD}$ increases the objective to $C_{\text{worst-case}}$. Hence, C is limited to the range $[C_{\text{ideal}}, C_{\text{worst-case}}]$, while the Primal-SSSP is always feasible.

Next, we enhance Primal-SSSP, allowing to (partially) block $\mathcal{Z}_{\kappa,\pi}$ affected by $\mathcal{H}_{\text{disrupted}}$. Let $\mathcal{L}_{h,r}$ be the set of all $l \in \mathcal{L}_r^{\text{train}}$ passing h . Then, introducing a blocked infrastructure section $h \in \mathcal{H}_{\text{disrupted}}$, requires one modification and one additional constraint in the M-SSSP:

$$\sum_{l \in \mathcal{L}_{h,r}} g_l^r \leq |\mathcal{L}_{h,r}| \cdot x_h, \quad \forall h \in \mathcal{H}, \forall r \in \mathcal{R} \quad (21a)$$

$$g_l^r \leq \delta_{\kappa,\pi}, \quad \forall l \in \mathcal{L}_{\kappa,\pi}, \forall \pi \in \Pi_{\kappa}, \forall \kappa \in \mathcal{K}, \quad (21b)$$

$$x_h \in \{0, 1\} \quad \forall h \in \mathcal{H}. \quad (21c)$$

Eq. (21a) is an additional set of constraints which allows to block infrastructure sections $h \in \mathcal{H}$. If h is blocked, we set the decision variable $x_h = 0$ and consequently, no more vehicles g_i^r may use the affected $l \in \mathcal{L}_h$. Contrarily, $x_h = 1$ does not add any restriction. Secondly, Eq. (21b) replaces Eq. (9f) from the original **M-SSSP** formulation. Given this modification, it is possible to operate only parts or to ignore an active $\delta_{\kappa,\pi}$, which is crucial for implementing disruption management measures. Finally, we can introduce several response possibilities which influence the margin of adjustments that the **Primal-SSSP** has access to. The following options can be included/excluded accordingly:

Short turning means to partially cancel a **TSS** by no longer passing over the disrupted section but still serving the **TSS**'s remaining part(s). We can include the option of short-turning by adding $l \in \mathcal{L}^{\text{access}}$ and $l \in \mathcal{L}^{\text{egress}}$ links on the $SSN_{r,vehicle}$ at suitable intermediate stations, such that short turning a train there is possible.

Cancellation indicates that a κ is no longer operated. This measure is essentially similar to *short turning* at the first and last $n \in \mathcal{N}^{\text{station}}$. Subsequently, any p_i^{od} travelling on any a cancelled g_i^r has to take an alternative route.

Alternative modes such as buses, provide alternative connections, bridging the gap inflicted by the disruption. However, the limit of available vehicles $\max(r)$ limits the additional service provided. We can introduce such services by the SSN from an expanded line plan $\mathcal{K} \cup \mathcal{K}_{\text{bus}}$, where \mathcal{K}_{bus} holds all alternative κ that can be deployed during a disruption. Consequently, \mathcal{Y}_{bus} holds all $\mathcal{Z}_{\kappa,\pi}$ that we can derive from \mathcal{K}_{bus} . Given that we focus on railways, we do not check for conflicts on road traffic. Hence, any $\mathcal{Y}_{\text{active}} \subseteq \mathcal{Y}_{\text{bus}}$ is feasible in the **M-SSSP**.

All these interventions are common means to cope with disruption, thus including them allows us to obtain an accurate assessment. Furthermore, when we limit the set of $\mathcal{Y}_{\text{available}} = \mathcal{Y}_{\text{active}} \cup \mathcal{Y}_{\text{bus}}$, we skip the step of assessing the resulting $\mathcal{Y}_{\text{response}} \subseteq \mathcal{Y}_{\text{available}}$, in the **S-SSSP**. This decision is based on the assumption, that removing trains and adding buses does not lead to a conflict.

Subsequently, we introduce further responses in form of disruption management measures. Since these lead to more drastic adaptations, it is required to assess any solution with the **S-SSSP** as well. Thus, when we prepare the **EAN** for the **S-SSSP** we also have to remove all $\mathcal{A}^{\text{train}}$ blocked by the $h \in \mathcal{H}_{\text{disrupted}}$. Note that we only outline how these additional responses could be implemented. However, we will limit our later evaluation to the already proposed responses.

Reordering trains , i.e., adjusting the sequence of trains can be included by not enforcing $\mathcal{Y}_{\text{available}} = \mathcal{Y}_{\text{active}}$ such that for the active $\kappa \in \mathcal{K}'$ alternative $\mathcal{Z}_{\kappa,\pi}$ are available. Naturally, we can influence the room for adjustments with the size of $\mathcal{Y}_{\text{available}}$.

Rerouting a $\kappa \in \mathcal{K}'$ requires the introduction of an alternative $\kappa \in \mathcal{K}_{\text{alternative}}$, which can be used to improve $C_{\text{disrupted}}$. Thus, we do not explicitly distinguish between rerouting or adding relieve/backup trains. Instead, we allow the **Primal-SSSP** to take the actions which lead to the best improvement of $C_{\text{disrupted}}$. However, room for improvement is limited by the vehicles available $\max(r)$.

3.4.2 Finding the most critical Disruptions with the SSSP

Next, we modify the **Primal-SSSP** to identify the $\mathcal{H}_{critical} \subseteq \mathcal{H}$ for a given \mathcal{Y}_{active} . Thus, the adapted model **Dual-SSSP** aims to increase C by blocking some $h \in \mathcal{H}$. This task is prevalent in network interdiction (for more background, see [Smith and Song \(2020\)](#)). Therefore, we adjust the **SSSP** towards an interdiction model. As common in network interdiction, we include a budget ($k_{scenario}$), limiting $|\mathcal{H}_{critical}|$ disruptions. Hence, by varying $k_{scenario}$, we are able to adapt to different requirements.

Since we assess an \mathcal{Y}_{active} solution, all decision variables $\delta_{\kappa,\pi}$ are fixed to the given solution \mathcal{Y}_{active} . Consequently, if we prune all inactive \mathcal{N}^{train} from the SSN for all $r \in \mathcal{R}$ and $od \in \mathcal{OD}$, we can drop all $\delta_{\kappa,\pi}$, no matter if active or inactive.

Furthermore, we can utilise the binary variable x_h as indicator if the infrastructure section h is blocked ($x_h = 0$) or not ($x_h = 1$). To find $\mathcal{H}_{critical}$, **Dual-SSSP** maximises C defined in (Eq. (1)). However, since passengers still travel via the shortest paths available, we encounter the following objective:

$$\max (\min C) = \max (\min \sum_{od \in \mathcal{OD}} \sum_{l \in \mathcal{L}} c_l^{od} \cdot \bar{w}_l \cdot p_l^{od}) \quad (22)$$

This objective in Eq. (22) represents a classical leader follower situation ([Smith and Song, 2020](#)). The leader, the **Dual-SSSP** tries to maximise $C_{disrupted}$ with deactivating some x_h , while the passengers take the best following action, i.e., the paths that minimise $C_{disrupted}$. As suggested by [Smith et al. \(2013\)](#), we dualise the passenger routing part of the **M-SSSP**, transforming the max –(min) objective to max.

Therefore, we dualise constraints **9b** and **9d**, without modifying any other constraint of the **Primal-SSSP**. Let α be the dual for **9b** while $-\beta$ is the dual for **9d**. Subsequently, we can replace the objective in **9a** and constraints **9b** and **9d**, with the formulation in **23**. Note that we are adding all the remaining constraints of **9** to **23**, but we omit them here for brevity.

$$\max \sum_{od \in \mathcal{OD}} \sum_{n \in \mathcal{N}_{od}} \alpha_n^{od} \cdot \gamma_n^{od} - \sum_{r \in \mathcal{R}} \sum_{l \in \mathcal{L}} capacity(r) \cdot g_l^r \cdot \beta_l \quad (23a)$$

$$\text{s.t.} \quad \alpha_i - \alpha_j - \beta_{i,j} \leq c_{i,j} \cdot w_{i,j}, \quad \forall (i, j) \in \mathcal{L}_{od}, \forall od \in \mathcal{OD}, \quad (23b)$$

$$\sum_{h \in \mathcal{H}} x_h \geq |\mathcal{H}| - k_{scenario}, \quad (23c)$$

$$\sum_{l \in \mathcal{L}_h} g_l^r \geq |\mathcal{L}_h| \cdot x_h, \quad \forall r \in \mathcal{R}, \forall h \in \mathcal{H}, \quad (23d)$$

$$\sum_{h \in \mathcal{H}'_{critical}} x_h \leq |\mathcal{H}'_{critical}| - 1 + \sum_{h \in \mathcal{H} \setminus \mathcal{H}'_{critical}} x_h, \quad (23e)$$

$$\beta_l^{od} \in \mathbb{R}^+, \quad \forall l \in \mathcal{L}_{od}, \forall od \in \mathcal{OD}. \quad (23f)$$

Note that the dualisation leads to a quadratic objective in Eq. (23a), since both g_l^r and β_l are decision variables. However, according to [Smith et al. \(2013\)](#), we can linearise it with sufficiently large bounds on β . Eq. (23b) stems from the dualisation.

Besides, we add Eq. (23d), to deal with a side effect. As stated previously, the maximal value for C is $C_{\text{worst-case}}$; indicating that all services are cancelled, even if $\mathcal{H}_{\text{disrupted}} = \{\}$. Thus, Eq. (23d) requires all g_i^r to operate, unless the section i is blocked ($x_i = 0$). Furthermore, we restrict the total number of disrupted sections with Eq. (23c).

In some cases, we might want to exclude a known disruption given by $\mathcal{H}'_{\text{disrupted}}$. One such instance, is when we use the **Dual-SSSP** to iteratively identify $\mathcal{H}_{\text{disrupted}}$ as candidates for $\mathcal{H}_{\text{critical}}$. Thus we include Eq. (23e) to allow banning a given $\mathcal{H}'_{\text{disrupted}}$. Note that the constraint is designed specifically to still allow the case where $\mathcal{H}'_{\text{disrupted}} \subseteq \mathcal{H}_{\text{disrupted}}$. This is a strict requirement, as otherwise we would remove more than $\mathcal{H}'_{\text{disrupted}}$ from the possible solutions to the **Dual-SSSP**.

Due to the quadratic objective in Eq. (23a), we limit the **Dual-SSSP** to *short turning* and *cancellation*, which represent a subset of responses to reduce the impact of disruptions in Section 3.4.1. Thus, the reformulated **Dual-SSSP** provides an upper bound on $C_{\text{disrupted}}$. On the contrary the **Primal-SSSP** might have access to more responses to cope with the $\mathcal{H}_{\text{disrupted}}$.

3.4.3 Primal-Dual Algorithm for Finding the most Critical Disruption

Inside the **AM**, we combine the two introduced models to a **primal-dual** algorithm to identify $\mathcal{H}_{\text{critical}}$ for a given $\mathcal{Y}_{\text{active}}$. Initially, we describe the fundamental properties of the algorithm. Subsequently, we elaborate, how we scale the impact of a disruption by the number of disrupted sections, allowing the algorithm to assess disruptions with varying number of simultaneous blocked sections.

The algorithm aims to combine the benefits of the **Primal-SSSP** and **Dual-SSSP**. First, we use the **Dual-SSSP** to identify a candidate $\mathcal{H}_{\text{critical}}$ based on P and apply the **Primal-SSSP** to then obtain an accurate $P_{\text{disrupted}}$. Since the **Dual-SSSP** yields an upper bound on $P_{\text{disrupted}}$, the algorithm iterates until the current bound $P_{\text{disrupted}}$ is lower than the P_{critical} for current candidate $\mathcal{H}_{\text{critical}}$. For more background on Primal-Dual algorithms, we refer to [Goemans and Williamson \(1997\)](#).

We aim to identify the disruptions with the most adverse consequences, while also considering different numbers of simultaneous disruptions. Thus, we propose to scale $P_{\text{disrupted}}$ within the **primal-dual** algorithm. Without considering the number of disrupted sections k , any disruption where $|\mathcal{H}_{\text{critical}}| = \max(K)$ is will have an impact which is at least as big as if the disruption only concerned a subset of the same $\mathcal{H}_{\text{critical}}$. This case renders the consideration of any disruption where $k < \max(K)$ obsolete. Thus, we propose to normalise the impact of $\mathcal{H}_{\text{critical}}$ by the number of disrupted sections k as in Eq. (24). With this scaling we can compare disruptions with different sizes $k \in K$. We denote the critical disruption with $\mathcal{H}_{\text{disrupted}}$, where $x_h = 0$ indicates that section h is disrupted while $x_h = 1$ indicates that h is unaffected.

$$P_{\text{scaled}}[\%] = \frac{P_{\text{nominal}} - P_{\text{disrupted}}}{|\mathcal{H}_{\text{disrupted}}|} \quad (24)$$

Consequently, if we scale the vulnerability of $\mathcal{H}_{\text{disrupted}}$ with Eq. (24) for a given $\mathcal{X}_{\text{disrupted}}$ for the **Primal-SSSP** and **Dual-SSSP**, we are capable of comparing disruptions with a different number

of simultaneously disrupted sections. Besides, as the normalisation is a linear transformation, for any $\mathcal{H}_{\text{disrupted}}$, $P_{\text{disrupted}}^{\text{dual}} \leq P_{\text{disrupted}}^{\text{primal}}$, $P_{\text{scaled}}^{\text{dual}} \geq P_{\text{scaled}}^{\text{primal}}$ holds. This relationship is the basis of the Primal-Dual algorithm in Fig. 12.

Figure 12: Primal-Dual algorithm for finding the most critical disruption.

```

Result: Result
1 ( $\mathcal{H}_{\text{critical}}$ ,  $P_{\text{critical}}$ );
2 set  $\sum_{i \in \mathcal{H}} x_i = |\mathcal{H}|$ ;
3 set  $P_{\text{nominal}} = \text{solve Primal-SSSP}$ ;
4 set  $P_{\text{scaled}}^{\text{primal}} = 0$ ;
5 set  $K_{\text{scenario}} = \{1, 2, \dots, k_{\text{scenario}}\}$ ;
6 while  $|K_{\text{scenario}}| > 0$  do
7   for  $k \in K_{\text{scenario}}$  do
8     set  $\sum_{h \in \mathcal{H}} x_h \geq |\mathcal{H}| - k$  in Dual-SSSP;
9     set  $P_{\text{disrupted}}^{\text{dual}} = \text{solve Dual-SSSP}$ ;
10    set  $P_{\text{scaled}}^{\text{dual}} = \frac{P_{\text{nominal}} - P_{\text{disrupted}}^{\text{dual}}}{k}$ ;
11    if  $P_{\text{scaled}}^{\text{max, primal}} \leq P_{\text{scaled}}^{\text{dual}}$  then
12      set  $\mathcal{H}_{k,\text{critical}} = \mathcal{H}_{\text{Dual-SSSP}}$ ;
13      set  $\mathcal{H}_{\text{Dual-SSSP}} = \mathcal{H}_{\text{Dual-SSSP}} \setminus \mathcal{H}_{k,\text{critical}}$ ;
14      set  $\mathcal{H}_{\text{Primal-SSSP}} = \mathcal{H}_{k,\text{critical}}$ ;
15      set  $P_{\text{primal}} = \text{solve Primal-SSSP}$ ;
16      set  $P_{\text{scaled}}^{k,\text{primal}} = \frac{P_{\text{nominal}} - P_{\text{disrupted}}^{\text{dual}}}{k}$ ;
17      if  $P_{\text{scaled}}^{\text{max, primal}} \leq P_{\text{scaled}}^{k,\text{primal}}$  then
18        set  $P_{\text{scaled}}^{\text{max, primal}} = P_{\text{scaled}}^{k,\text{primal}}$ ;
19        set  $\mathcal{H}_{\text{critical}} = \mathcal{H}_{k,\text{critical}}$ ;
20        set  $P_{\text{critical}} = P_{\text{primal}}$ ;
21      end
22    else
23      set  $K = K \setminus \{k\}$ ;
24    end
25  end
26 end
27 return ( $\mathcal{H}_{\text{critical}}$ ,  $P_{\text{critical}}$ );

```

Initially, the algorithm in Fig. 12 obtains the regular objective score (where no disruption takes place). Subsequently, it iteratively increases the number of concurrent disruptions up to the maximally allowed k_{scenario} . During each iteration, we obtain the approximately worst disruption according to the **Dual-SSSP**.

If the relative increase is lower than the one of the currently known worst disruption $P_{\text{scaled}}^{\text{max, primal}}$, we can remove this number of concurrent disruptions k from the set of possible numbers K . Otherwise, we ban the found disruption in the **Dual-SSSP**, preventing of accounting of the same disruption twice. Furthermore, we assess the found disruption with the **Primal-SSSP** for the accurate relative increase $P_{\text{scaled}}^{k,\text{primal}}$. If the increase is still higher than $P_{\text{scaled}}^{\text{max, primal}}$, we define the current disruption as the most critical disruption.

4 Case Study

This chapter consists of three sections and eventually aims to evaluate the proposed methodology. Initially, we provide some implementation details in Section 4.1, before we outline the case study data in Section 4.2. Finally, we provide a a brief overview on the experiments in Section 4.3, before conducting two sets of computational experiments in Section 4.4 and Section 4.5, enabling an evaluation of the methodology proposed in Section 3. Naturally, all the following content is used to tackle SQ 4.

4.1 Implementation of the iterative Approach

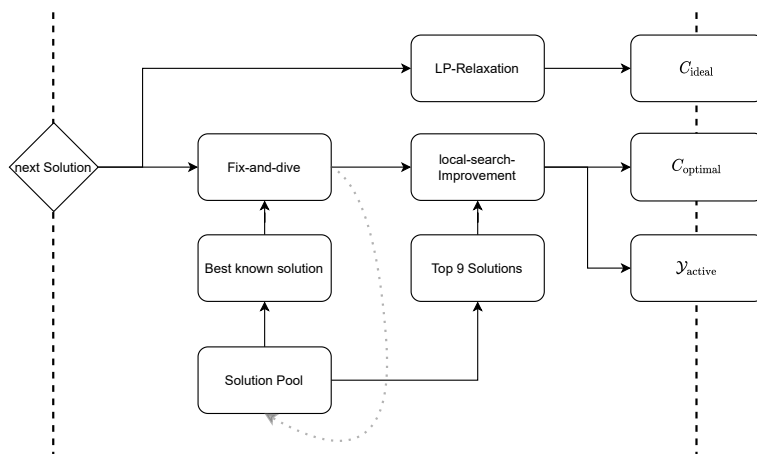
In this section, we briefly describe how we implemented the heuristic approach that we use to tackle the case study with the OM, before we provide background on the used software and tools.

4.1.1 Heuristic Architecture in the OM

The primary motivation is that the Slot Sequence Selection Problem (SSSP) in the OM is likely very challenging, as it has been reported for similar approaches that integrate passenger routing and timetabling (Polinder *et al.*, 2021; Schiewe, 2020).

Fig. 13 provides an overview of the implementation of our heuristic: The heuristic has three key outputs. C_{nominal} and $\mathcal{Y}_{\text{active}}$ are self explanatory. C_{ideal} on the other hand is an additional indicator, which we can use to derive a lower bound for P_{ideal} in cases where we tackle the SSSP with heuristic measures only. Since C_{ideal} is obtained from the linear programming relaxation of the SSSP, we are certain that $C_{\text{ideal}} \leq C_{\text{nominal}}$ holds in any case.

Figure 13: The heuristic procedure implemented in the OM.



The heuristic architecture utilises the iterative nature of the overall approach to its advantage, as it has a solution pool of known $\mathcal{Y}_{\text{active}}$ that is feasible in the **S-SSSP**. Thus, we can use the best known previous solution as a starting point for the fix and dive heuristic outlined in Section 3.3.6. In the heuristic, we sort the $\kappa \in \mathcal{K}$ by the given priority, such that initially all *express*, then all *regional* and finally all *local* trains are scheduled. Note that we do not relax any non-passenger carrying trains.

After obtaining a solution from the fix-and-dive heuristic, we apply a local search procedure that aims to improve the current C_{nominal} . To adapt the **M-SSSP** for the local search, we add one additional constraint during the search. The constraint combines the current top nine known configurations of $\mathcal{Y}_{\text{active}}$ with the $\mathcal{Y}_{\text{active}}$ given by the fix-and-dive search to a union given by $\mathcal{Y}_{\text{top-ten-active}}$. The additional constraint can then be given by:

$$\sum_{(\kappa, \pi) \in \mathcal{Y}_{\text{top-ten-active}}} \delta_{\kappa, \pi} \geq |\mathcal{Y}_{\text{active}}| - 1, \quad (25)$$

This constraint in Eq. (25) reduces the search space for alternative $\mathcal{Y}_{\text{active}}$, and we aim to improve a given solution by this procedure.

4.1.2 Implementation Details

We use a laptop computer with 32 GB RAM and an Intel Xeon E-2176M CPU with six cores to run the case study. We implement the outlined process in python 3.9 using data structures provided by the OpenBus Toolbox of Fuchs and Corman (2019), which in turn relies on the python-igraph implementation of Csardi and Nepusz (2006). Note that we run all python code with the `-O0` flag.

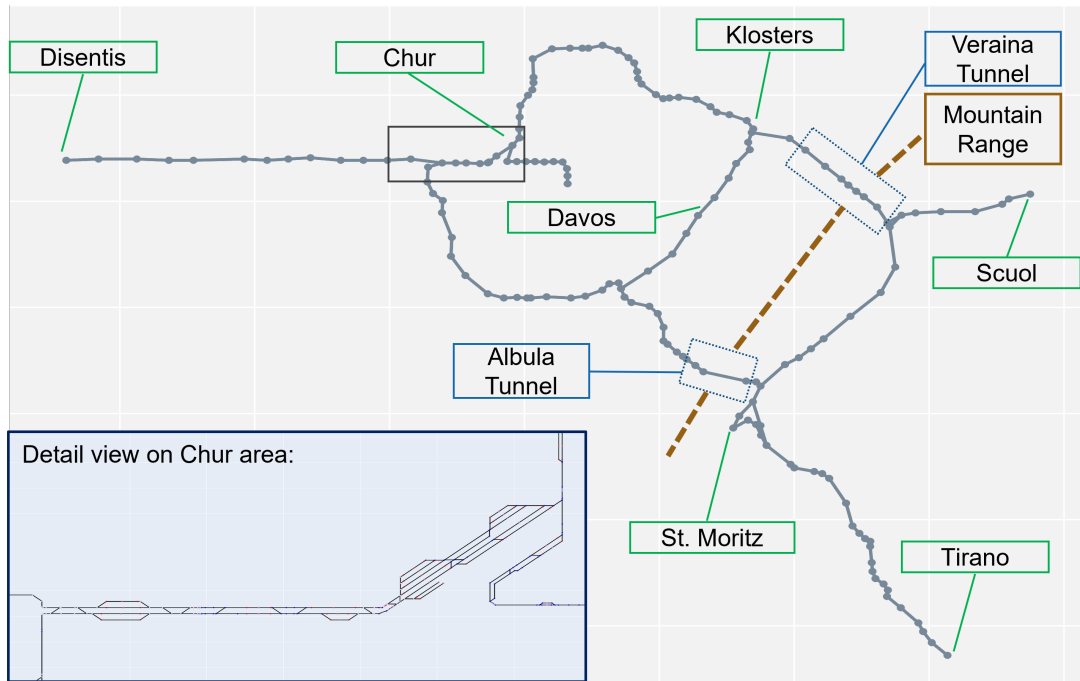
We solve all **MIP** formulations with the commercial **MIP**-solver GUROBI 9.5.0 (Gurobi Optimization, LLC, 2022), through the python API. The **MIP**-solver may use up to twelve threads. As there are many open source **SAT**-solvers available, we ran preliminary tests to identify a suitable one, see Appendix A.2 for more information. Based on the outcome of the preliminary tests, we decided on using MiniCard 1.2 provided by Liffiton and Maglalang (2012). The **SAT** solver is single-threaded. We interact with the solver through the python-**SAT** library of Ignatiev *et al.* (2018).

4.2 Case Study Data

As we apply the proposed approach on a real-life case study, we briefly introduce all used data in this section. All data is kindly provided to us by [Rhaetian Railway \(RhB\)](#), a Swiss railroad company. The case study data contains the infrastructure network, line plan, rolling stock and information on expected demand.

4.2.1 Infrastructure Network and Timetabling Parameters

Figure 14: Macroscopic view on the infrastructure network.



We use the infrastructure provided by [RhB](#) which is similar to the one used by [Fuchs et al. \(2021\)](#). Figure 14 provides an overview on the infrastructure. The network distinguishes itself because many sections consist of only a single track. Furthermore, it has a length of 380 km. Following the example of [Fuchs et al. \(2021\)](#), we enforce a headway of 120 seconds at every location. This headway value is achieved in practice and selected following technical advice of [RhB \(RhB, 2005, 2022\)](#).

[RhB](#) provides the infrastructure on a microscopic level of detail, and similar to [Fuchs et al. \(2021\)](#), we aggregate it to a mesoscopic level of detail. As the data only contains information on the length of a track but no traversal times are available, we use the assumptions of [Fuchs et al. \(2021\)](#) to approximate the traversal time for a given track. Therefore, we approximate the traversal time \hat{w}_a by a constant speed per speed category, as seen in Table 2, which are identical to the values of [Fuchs et al. \(2021\)](#). [Fuchs et al. \(2021\)](#) state that the trip times seem reasonable ([SBB, 2021](#)).

Table 2: Speed in m/s per train category to approximate \hat{w}_a .

	<i>cargo</i>	<i>local</i>	<i>regional</i>	<i>express</i>
	19	21	20	21

Additionally, the data yields a minimal stop time of 42 seconds and a connection time for passengers of three minutes. Thus, we define $\mu_{\text{access}} = 180\text{sec}$ and $\mu_{\text{egress}} = 0\text{sec}$ for passengers. Besides calculating minimal trip times, we also have to define the maximally allowed duration of the trip and dwell activities in the timetable. In their assessment, Fuchs *et al.* (2021) show that some flexibility on the trip duration is highly beneficial in finding feasible timetables for this network. Therefore, we include additional slack for all trip times where the track length exceeds 300 m, while shorter tracks must be passed with constant trip times. Thereby, we aim to include as much flexibility as possible while considering that short tracks might require excessive acceleration if trip time varies. We allow the dwelling at any station track for further flexibility, even if no commercial stop is scheduled. Table 3 summarises our approach on defining the bounds.

Table 3: Bounds per activity in the *EAN*.

activity type	$\forall a \in A^{\text{dwell}}$		$\forall a \in A^{\text{trip}}$	
	<i>non-commercial</i>	<i>commercial</i>	<i>length < 300m</i>	<i>length \geq 300m</i>
w^{\min}	0 sec	40 sec	\hat{w}_a	\hat{w}_a
w^{\max}	100 sec	140 sec	\hat{w}_a	$\hat{w}_a + 100\text{ sec}$

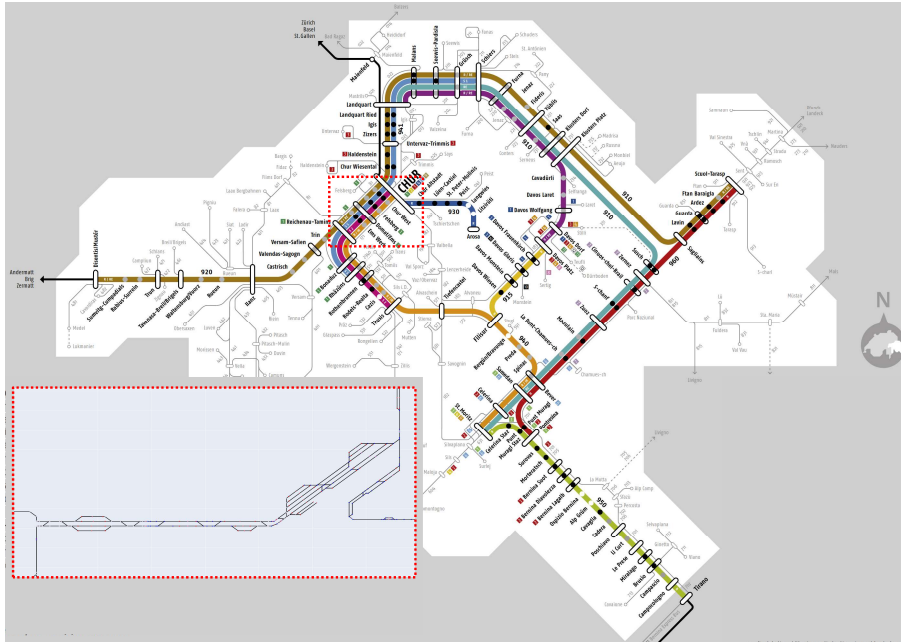
Following the current practise of RhB and Swiss standards, \mathcal{T} is one hour (RhB, 2021b). Furthermore, Fuchs *et al.* (2021) showed, that a time discretisation dt of 20 seconds is appropriate, hence we opt for the same value. Note that this is a finer granularity than the commonly used 60 seconds in PESP (Caimi *et al.*, 2017). Besides, as Dekker *et al.* (2021), we apply constraint propagation to tighten all w_a^{\min}, w_a^{\max} for all $a \in \mathcal{A}^{\text{anchor}}$.

4.2.2 Line Plans for the Case Study

Following the scope of this thesis, we use the current line plan of the RhB depicted in Fig. 15 as a given input. It consists of 10 lines which all operate at a frequency of once per hour. Since we use *directed lines* ($\kappa \in \mathcal{K}$), these ten lines amount to twenty *directed lines*. Furthermore, in accordance with RhB, we split the express line from Disentis to Schuls in Chur, such that this one line yields four *directed lines*. Thus, we end up with 22 *directed lines* for passenger service.

Besides, there is a shuttle service for cars through the Veraina tunnel that operates twice per hour (see Fig. 14 for the location of the Veraina tunnel). Furthermore, the **RhB** also plays a vital role as a cargo carrier (**RhB**, 2005). Hence, we also include the cargo trains used by **Fuchs et al.** (2021).

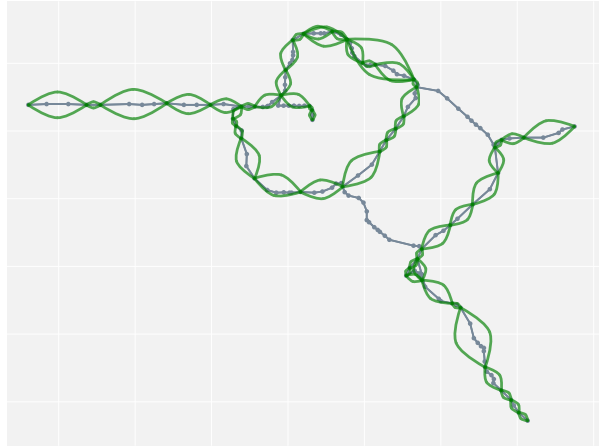
Figure 15: Network of the **Rhaetian Railway (RhB)** (**RhB**, 2021b) with an excerpt of the infrastructure.



To prevent the **OM** from providing solutions, where passengers would need to change between trains at unsuitable stations, we limit the set of stations where transfers between trains are allowed. Appendix A.1 provides a visualisation of the remaining subset. Similarly, we restrict the change stations in the **AM** to stations where infrastructure for transferring between trains is available. A visualisation is given in Appendix A.1. Lastly, we limit the stations where trains can short-turn. A station must have at least two station tracks to allow short turning. We refer to Appendix A.1 for a visualisation. All these assumptions have been given to us by **RhB**, (**RhB**, 2022).

Eventually, we add some non-mandatory bus lines, depicted in Fig. 16 that can be used as alternative services to respond to disruptions in the **Primal-SSSP**. These *replacement services* only serve stations, where **RhB** provides transfers in the regular line plan, which is similar to current practices of **RhB** (**RhB**, 2022). Furthermore, the *replacement services* only serve two stops per line. Nevertheless, if two adjacent lines are activated, it is possible to provide a service that spans more than two adjacent stations.

Figure 16: Bus lines available as response measures.



Since we already use an approximation to estimate the trip times for trains, we also opt for an approximation of the trip times for buses. Under the advice of [RhB](#), we use a constant 18 m/sec to estimate the trip times based on the length of the railway infrastructure [RhB \(2022\)](#). This procedure is a rough approximation; however, we concur with this approximation given the purpose of this case study.

Note that the speed of *replacement services* is lower when compared with any train. This difference accounts for additional time to cover detours and hindrances in road traffic.

4.2.3 Train Slot Sequence (TSS) Generation

Since the current line plan and the work of [Fuchs et al. \(2021\)](#) define the routes and stop sequences for all trains, we can use them as input. Thus, following the definition of the **TSS** in Section 3.2.2, generating all available **TSSs** in $\mathcal{Z}_{\kappa,\pi} \in \mathcal{Y}_{\text{available}}$ is straightforward.

As described in Section 3.2.2, we use $\hat{t}_{n,e}$ to calculate the time needed to reach any $(n, e) \in \mathcal{S}_{\kappa}$. We obtain $\hat{t}_{n,e}$ by calculating the minimal required time to reach the (n, e) pair from the root pair of \mathcal{S}_{κ} . To allow the **TOC** to include some slack, we scale $\hat{t}_{n,e}$ with the parameter μ^{scale} before rounding it to the nearest value, such that the granularity μ^{dt} that is demanded by the **TOC** for this \mathcal{S}_{κ} is ensured. As we always round $\hat{t}_{n,e}$ based on the time from the root pair of \mathcal{S}_{κ} , rounding errors do not accumulate.

Following Section 3.2.2, we may assign the time slots given by $t_z^{\text{min}}, t_z^{\text{max}}$ based on these rounded timestamps. As also given by Section 3.2.2, we distinguish between commercial and non-commercial slots and use the parameters given in Table 4. To generate all **TSSs** given by $\mathcal{Z}_{\kappa,\pi} \in \mathcal{Y}_{\kappa}$ for one κ , we discretize \mathcal{T} by the corresponding μ^{spread} parameter. This step yields $\pi \in \{0, \mu^{\text{spread}}, \dots, T - \mu^{\text{spread}}\}$ to create all $\mathcal{Z}_{\kappa,\pi}$ for the corresponding κ . Thus, the **TOC** defines the number of available $\mathcal{Z}_{\kappa,\pi}$ with μ^{spread}

Table 4 provides all parameter configurations we use to generate the TSSs given by $\mathcal{Z}_{\kappa,\pi} \in \mathcal{Y}_{\text{available}}$. We selected these parameters based on recommendations from RhB (RhB, 2022).

Table 4: Input parameters to calculate the TSS per train category.

	μ^{scale}	μ^{com}	μ^{gem}	μ^{spread}	μ^{dt}
<i>express</i>	1.075	180 sec	300 sec	120 sec	60 sec
<i>regional</i>	1.075	180 sec	300 sec	120 sec	60 sec
<i>local</i>	1.075	180 sec	300 sec	120 sec	60 sec
<i>bus</i>	1.075	180 sec	180 sec	60 sec	60 sec
<i>auto train</i>	1.100	600 sec	300 sec	300 sec	60 sec
<i>freight train</i>	1.100	900 sec	300 sec	300 sec	60 sec

4.2.4 Rolling Stock

In accordance with RhB, we base our vehicle pool on the vehicles available to RhB (RhB, 2021a), summarised in Table 5. Note that we assign the rolling stock in form of predefined configurations, which provides the capacity to transport passengers. Furthermore, we define $\mu_{\text{access}} = 180$ sec and $\mu_{\text{egress}} = 180$ sec for all vehicles.

Table 5: Available rolling stock, based on RhB (2021a).

category	type	$\max(r)$	capacity
<i>express</i>	engine with carriage set	20	280
<i>regional</i>	electrical multiple unit	25	300
<i>local</i>	electrical multiple unit	8	150
<i>bus</i>	replacement bus	0 / 5 / 10	45
<i>cargo</i>	auto train	∞	-
<i>cargo</i>	freight train	∞	-

4.2.5 Defining the Parameters for Generalised Travel Time

As introduced in Section 3.1.3 we use the the total generalised travel time to evaluate the system-wide performance. We use the same values as Zhu and Goverde (2020b), such that c_s for in-vehicle-time of $l \in \mathcal{L}^{\text{train}}$ is weighted with $c_s = 1$, while any other l , used for *waiting*, *accessing* and *excessing* is weighed with $c_s = 2.5$. To calculate the cost of a bypass for one $od \in \mathcal{OD}$, given by c_{\max}^{od} , we adapt the minimal generalized travel time required for this relation (c_{\min}^{od}), which corresponds to the shortest path from v_o to v_d in the SSN_{pax}^{od} . To obtain c_{\max}^{od} , we use the following,

$$c_{\max}^{od} = c_{\min}^{od} \cdot \eta \quad (26)$$

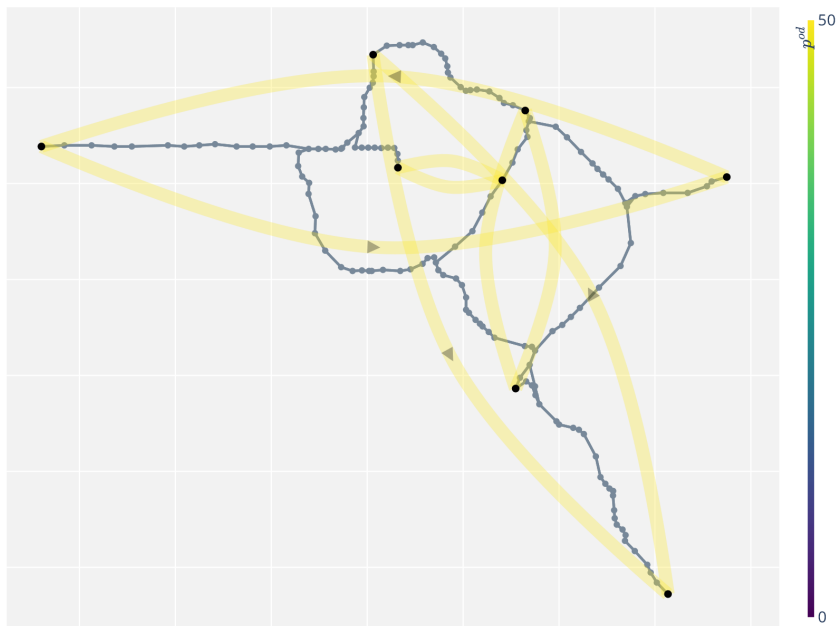
which limits the maximum accepted detour, such that it is lower than c_{\min}^{od} scaled by η . Similar to [Zhu and Goverde \(2020b\)](#), we choose $\eta = 2$. Consequently, passengers leave the system if they have to spend more than twice the generalised travel time compared with the best possible connection.

4.2.6 Demand Scenarios

We consider three demand scenarios, an artificial one ($\mathcal{OD}_{\text{artificial}}$) and two ($\mathcal{OD}_{\text{subset}}$, $\mathcal{OD}_{\text{full}}$) which are derived from real-life demand provided by [RhB](#).

As indicated by its name, $\mathcal{OD}_{\text{artificial}}$ is a completely artificial demand scenario, which we crafted ourselves, and its purpose is to provide a simple case to assess the methodology. As visible in [Fig. 17](#), $\mathcal{OD}_{\text{artificial}}$ contains 8 \mathcal{OD} relations, each with a demand of 50 passengers. Furthermore, all these \mathcal{OD} -relations are symmetric, i.e., the number of passengers travelling from and to some o and d are equal.

Figure 17: $\mathcal{OD}_{\text{artificial}}$ visualised with a macroscopic view of the infrastructure.

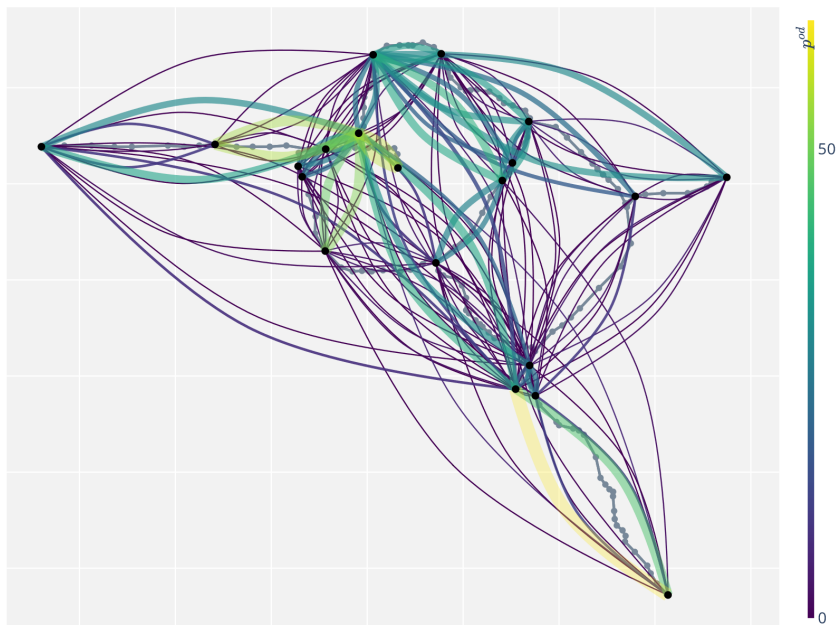


$\mathcal{OD}_{\text{full}}$ on the other hand is a real-life demand scenario provided by [RhB](#) and is the same as the $\mathcal{OD}_{\text{full}}$ scenario used by [Fuchs et al. \(2021\)](#). Following [Fuchs et al. \(2021\)](#), we scale the daily demand by 0.1 to get an approximation of the average demand during T . The resulting $\mathcal{OD}_{\text{full}}$ consists of 101 origin/destination locations and contains 1747 $od \in \mathcal{OD}$.

Finally, the third scenario, $\mathcal{OD}_{\text{subset}}$ is a subset of $\mathcal{OD}_{\text{full}}$, as it only contains 20 of the 101 origin/destination locations of $\mathcal{OD}_{\text{full}}$. We select the subset of origin and destination locations to

maximise the total number of passengers over the remaining nodes. Thus, we aim to extract a network where the 20 most *important* nodes remain.

Figure 18: $\mathcal{OD}_{\text{subset}}$ visualised with a macroscopic view of the infrastructure.



Given that $\mathcal{OD}_{\text{full}}$ contains more than 1000 relations, we do not provide a visualisation. However, Fig. 18 provides a visualisation of $\mathcal{OD}_{\text{subset}}$. $\mathcal{OD}_{\text{subset}}$ contains 173 \mathcal{OD} -relations. Furthermore, the sum over all $p^{od} \in \mathcal{OD}_{\text{subset}}$ is 1831.13 passengers, compared to $\mathcal{OD}_{\text{full}}$, where the sum amounts to 3300.89 passengers. Thus, $\mathcal{OD}_{\text{subset}}$ contains less than 10% of the od relations but more than 50% of the total demand of $\mathcal{OD}_{\text{full}}$.

4.3 An Overview on the Computational Experiments

We use the case study to investigate the usefulness and applicability of our approach defined in Section 3. Following our intentions stated in Section 2.3, the proposed approaches should provide a timetable where resilience towards the critical disruption is increased. Besides, we also guide planners towards critical infrastructure elements. As introduced in Section 4.2, we allow short turning and cancellation of all trains as responses, since this is common practice at RhB. Furthermore, depending on the chosen setting, we allow the AM to use either **no-buses**, **five-buses** or **ten-buses** to respond to a disruption, as this is also a frequent measure of RhB.

Our experiments are structured in two sections. Initially, we conduct individual assessments to evaluate the OM (proposed in Section 3.3) and AM (proposed in Section 3.4). Subsequently, we assess both modules combined in the proposed iterative approach given in Section 3.2. To provide readers of this thesis with additional material, several figures that visualise the results of the computational experiments are available as interactive *html* files. Any figure whose description is linked to Appendix A.3 is provided as *html*-file. Following the instructions in Appendix A.3, the *html*-files can be inspected interactively, providing the combined information of a table and figure.

4.4 Computational Experiments for Individual Assessments

The aim of the individual assessments is to identify if each module performs as intended. Besides, we can provide some insights on the results. The order of the assessments follows the sequence of steps in one iteration of the iterative approach given in Section 3.2.

Initially, in Section 4.4.1, we examine the OM's capability to optimise timetables and evaluate, if using heuristics is appropriate. Next, we see if we may shrink the SSSP in Section 4.4.2. Subsequently, we assess the AM. Initially, we assess if the primal-dual algorithm is capable of detecting the critical disruption in Section 4.4.3. We further test the iterative approach in Section 4.4.4 by completing one iteration, assessing the result of the adapted SSSP in the OM once more with the AM. Lastly we try to enhance the resilience of a timetable against two simultaneous disruptions in Section 4.4.5.

4.4.1 Using the OM to Optimise Timetables

In our very first investigation, we assess the capability of the SSSP within the OM to provide a timetable. Besides, we aim to assess the **fix-and-div** heuristic (Section 3.3.6) integrated in the heuristic architecture (implemented in Section 4.1.1). Thus, we compare the heuristic against solving the SSSP directly. We solve the case study for each OD scenario in three configurations. The first configuration emphasises on the M-SSSP alone, as we use the **passenger** trains but disable the S-SSSP. In this **no-conflict** case, no conflict detection is active and any feasible M-SSSP solution is also feasible in the S-SSSP. The second configuration contains all **passenger** trains and the third all **passenger** and **cargo** trains. Naturally, we do not disable the S-SSSP in the second and third configuration.

Consequently, we end up with nine models to solve, where the first three should be easiest to solve while the challenge gradually increases. To compare the heuristic against the regular **MIP** solve, we run each model twice, once as **full-solve** and once as **heuristic**. For the **full-solve**, we limit the **MIP**-solver to 3600 sec of computation time for all $\mathcal{OD}_{\text{artificial}}$ and $\mathcal{OD}_{\text{subset}}$, while we allow 7200 sec for $\mathcal{OD}_{\text{full}}$, as it is considerably larger. On the other hand, the **fix-and-div** heuristic may use up to 200 sec per step. Furthermore, we allow the improvement heuristic to use another 400 sec. Note that we supply both approaches with a warm-start, such that the results are comparable. We obtain the $\mathcal{Y}_{\text{active}}^*$ used as warm-start by querying the **S-SSSP** for a feasible $\mathcal{Y}_{\text{active}}$ before starting the solving processes. For each configuration, we report the required computation time, C_{nominal} and the optimality gap. Note that we use C_{ideal} for the **heuristic** to calculate the gap, while for **full-solve**, we can use the gap reported by the **MIP** solver.

Table 6: Results for solving the **SSSP** with **passenger** trains and **no-conflict**.

		$\mathcal{OD}_{\text{artificial}}$	$\mathcal{OD}_{\text{subset}}$	$\mathcal{OD}_{\text{full}}$
full-solve	time [sec]	5.4	145.7	7200.0
	C_{nominal} [h]	972.5	1583.1	2378.5
	gap [%]	0.00	0.00	2.09
heuristic	time [sec]	141.0	755.0	2336.4
	C_{nominal} [h]	972.5	1583.1	2335.0
	gap [%]	0.00	0.00	0.28

Table 6 summarises the results for the **no-conflict** run with **passenger** trains. A first observation is that for the $\mathcal{OD}_{\text{artificial}}$ and $\mathcal{OD}_{\text{subset}}$ cases, it is apparent that the **heuristic** approach requires considerably longer computation time. This is most likely due to the fact, that the **fix-and-div** heuristic solves the **SSSP** for each κ once. **full-solve** on the contrary requires only one **MIP** to be solved, and is therefore faster. However, all four results yield an optimal result, where the gap to C_{ideal} is 0.0%. A second observation is, that for $\mathcal{OD}_{\text{full}}$, **full-solve** yields a considerable worse result than **heuristic**, while neither provides a solution with a gap of 0.0%. A reason for **full-solve** performing worse could be, that in order to allow **M-SSSP** and **S-SSSP** to interact via a callback, we have to set the *Lazy* parameter in the Gurobi-solver (Gurobi Optimization, LLC, 2022). This deactivates certain pre-solve procedures of the **MIP**-solver and thus might degrade its performance (Gurobi Optimization, LLC, 2022).

Table 7: Results for solving the **SSSP** with **passenger** trains.

		$\mathcal{OD}_{\text{artificial}}$	$\mathcal{OD}_{\text{subset}}$	$\mathcal{OD}_{\text{full}}$
full-solve	time [sec]	3600.0	3600.0	7200.0
	C_{nominal} [h]	1343.3	1627.1	2440.1
	gap [%]	27.6	2.70	4.57
heuristic	time [sec]	738.7	1254.6	3971.5
	C_{nominal} [h]	982.5	1602.0	2381.6
	gap [%]	1.03	1.26	2.28

Table 7 summarises the results for the run with **passenger** trains. Given that in these runs, the **S-SSSP** is no longer ignoring conflicting trains, we can expect the **SSSP** to be more challenging. When we compare the results in Table 7, it is observable, that **heuristic** excels in both, required time and optimality gap for any \mathcal{OD} . Besides, none of the **full-solve** approaches terminated within the time limit. Further noticeable is the difference between the gaps of the two $\mathcal{OD}_{\text{artificial}}$ runs, as **fill-solve** yields a result far from optimal, whereas **heuristic** leads to a result that is almost within 1% of C_{ideal} . Finally, it is observable, that for a bigger \mathcal{OD} , generally more computation time is required.

Table 8: Results for solving the **SSSP** with **passenger** and **cargo** trains.

		$\mathcal{OD}_{\text{artificial}}$	$\mathcal{OD}_{\text{subset}}$	$\mathcal{OD}_{\text{full}}$
full-solve	time [sec]	3600.0	3600.0	7200.0
	C_{nominal} [h]	1433.8	1672.0	2452.1
	gap [%]	32.1	5.32	5.04
heuristic	time [sec]	1373.7	2134.0	4867.9
	C_{nominal} [h]	985.0	1615.9	2434.9
	gap [%]	1.28	2.02	4.57

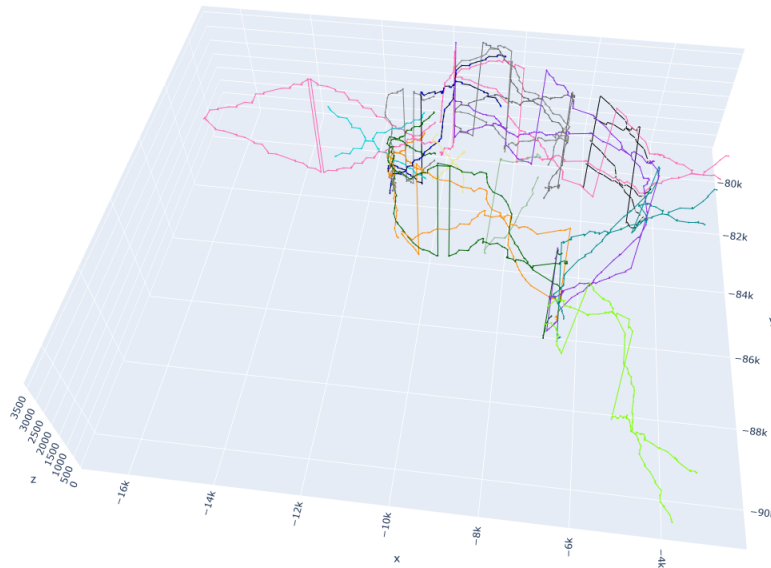
Table 8 summarises the results for the run with **passenger** and **cargo** trains. As these instances contain more trains, the challenge to solve the **SSSP** is likely increasing once more. When we compare the results in Table 8, it is observable, that once more, **heuristic** excels in both, required time and optimality gap for any \mathcal{OD} . However, while for $\mathcal{OD}_{\text{artificial}}$, the difference of the optimality gap is remarkable, it is considerably smaller for $\mathcal{OD}_{\text{full}}$. When we compare the results with the previous runs, it is apparent that the computation time for **heuristic** generally increases. Furthermore, none of the **full-solve** runs stops within the time limit.

These insights provided by Tables 6, 7 and 8 match with similar findings in literature, where integrated timetabling problems are frequently tackled with heuristics, see for example Schiewe (2020); Polinder (2020); Zhu and Goverde (2020b). These results indicate that directly solving the **MIP** is likely not a viable strategy for our case study. Thus, we restrict the **OM** to rely on **heuristic** for every successive experiment. Using **heuristic** consistently has the advantage that the results stem from the same approach and are comparable. Furthermore, since **heuristic** yields also a lower bound (C_{ideal}), we can assess the quality of a solution.

Before we conclude, we briefly investigate why the last cases with **passenger** and **cargo** trains given by Table 8 are challenging for both **full-solve** and **heuristic**. We therefore examine a resulting **EAN** provided by the **S-SSSP**, provided in Fig. 19. Since the case study consists of a network and not a corridor, we opt for a three-dimensional representation, where x and y are the coordinates, while z is the time in seconds. Fig. 19 reveals that infrastructure utilisation increases, especially in the more central network parts. Recall that most of the sections between stations only consist of one track. Consequently, passing/overtaking is often limited to suitable

stations. Fig. 19 reveals some recurring patterns, which are likely due to the limited possibilities to pass/overtake. Following this assumption, inserting cargo trains to the passenger further restricts the room for adjustment. Subsequently, finding an timetable with C_{nominal} close to C_{ideal} is challenging, if even possible. However, all listed hypotheses are based on assumptions.

Figure 19: Example timetable as 3D-time-space diagram, where z is time in seconds and x and y are coordinates in space (available as *html* (Appendix A.3)).



4.4.2 The effect of Shrinking the Instance Sizes in the M-SSSP

In section Section 3.3.7, we describe some procedures to reduce the number of variables in the M-SSSP required to route the passengers. While the pruning procedures are straightforward, the idea of bundling *od* relations by origin (Bull *et al.* (2019)) or departure locations has not been adopted widely yet. Hence, we specifically focus on the effect that bundling *od* has on the number of required variables in the M-SSSP.

Table 9: Number of required variables in the M-SSSP in the OM.

Bundling method	None	by origin	by destination	MIP heuristic
$OD_{\text{artificial}}$	72'143	72'143	72'143	72'143
OD_{subset}	895'846	275'113	299'173	251'991
OD_{full}	4'170'854	940'998	1'000'368	854'578

To provide some insights, we assess the impact of the outlined bundling procedures on the M-SSSP

used in the previous experiments, summarised in Table 9. The effect of bundling depends on the \mathcal{OD} scenario and the bundling method. However, our proposition to use an **MIP** heuristic to bundle the od relations outperforms any other method. That none of the bundling approaches yields a reduction for $\mathcal{OD}_{\text{artificial}}$ can be expected, as no od pairs share an origin or destination. However, for the other two cases, the impact is considerable. The biggest reduction is achieved with the **MIP** heuristic for $\mathcal{OD}_{\text{full}}$, where 79.5% of variables are removed.

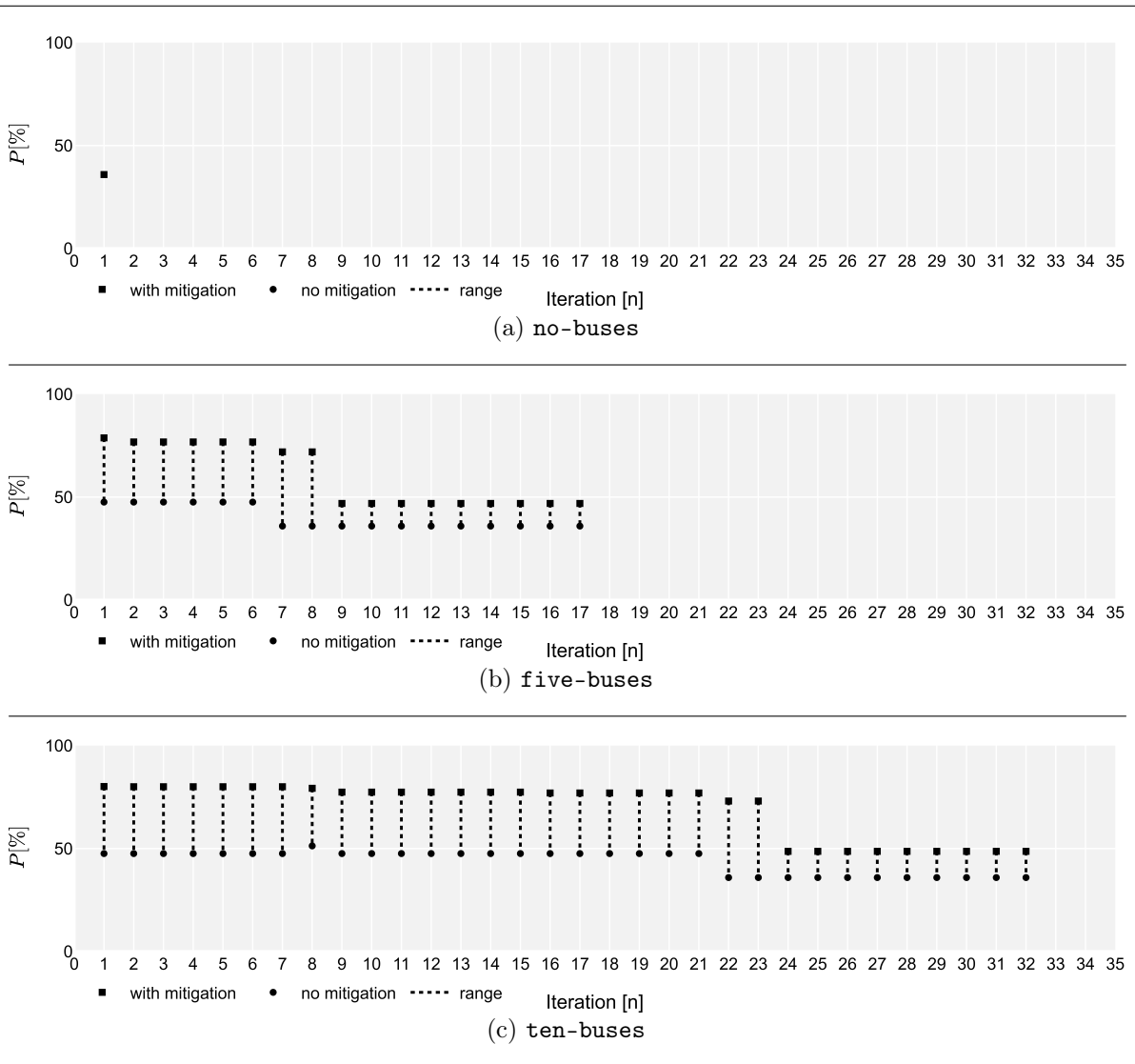
4.4.3 Using the **AM** to Detect $\mathcal{H}_{\text{critical}}$: One Disrupted Link

Complementary to the investigation of the **OM**, we succeed to assess **AM**, which contains the **Primal-SSSP** and **Dual-SSSP** in the form of the **primal-dual** algorithm. Our assessment consists of two stages. Initially, we start with a detailed example for one case, allowing a more in-depth understanding of how the algorithm works. We select a test case with $\mathcal{OD}_{\text{artificial}}$, **passenger** trains and **no-conflict** for the detailed investigation of the **primal-dual** algorithm. We specifically chose this configuration to minimise any side effects affecting the interpretation of the result. With the same intention, we set we set $K = \{1\}$ for the **primal-dual** algorithm. To obtain some contrast, we use three different responses. In the first one, **no-buses**, only short turning and cancellations of scheduled trains are allowed. The second configuration, **five-buses** may also use up to five replacement buses to mitigate the impact of a disruption. Analogously, the last configuration is **ten-buses**. After optimising the configuration with the **OM**, we submit the instances to the **AM**.

Fig. 20 visualises the results of the three assessments. While the x-axis provides the number of required iterations, the y-axis yields the values for $P_{\text{disrupted}}$ in P , which we obtain from the **Primal-SSSP** and the **Dual-SSSP** in the **primal-dual** algorithm. Since **Dual-SSSP** has fewer response measures, its values correspond to the *no mitigation* entries. Contrarily, the **Primal-SSSP** may have additional responses to reduce the impact of a disruption, thus we label these with *with mitigation*.

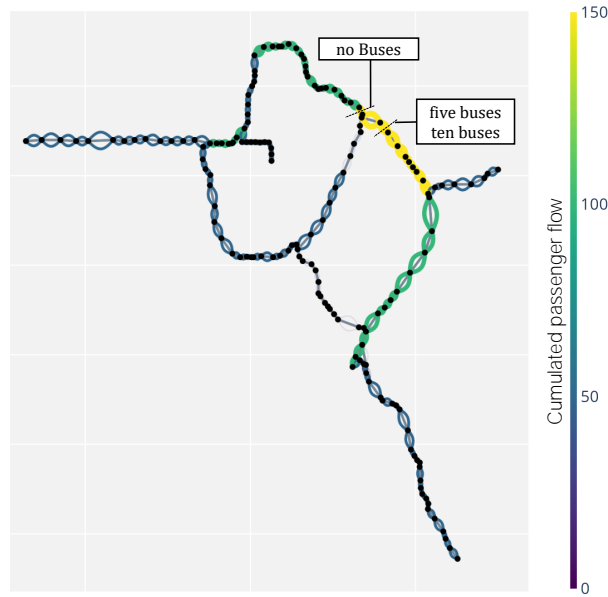
It is apparent, that in Fig. 20 the number of iterations depends on the number of buses available. If no buses are available (**no-buses**), the first candidate for $\mathcal{H}_{\text{critical}}$ found by the **Dual-SSSP** is the actual $\mathcal{H}_{\text{critical}}$, as the **Primal-SSSP** can not apply any further responses. Thus, P is equal for both models. Besides, as one is the dual of the other, P must be equal according to strong duality (Conforti *et al.*, 2014).

Figure 20: Results of the primal-dual algorithm (available as *html* (Appendix A.3)).



A different picture is given, when **five-buses** are available in Fig. 20. Depending on the $\mathcal{H}_{\text{disrupted}}$, the **Primal-SSSP** is able to reduce the impact considerably and the gap between the **Dual-SSSP** and the **Primal-SSSP** ranges from 10.9% up to 30.0%. It takes 17 iterations, until the **Dual-SSSP** can no longer provide any $\mathcal{H}_{\text{disrupted}}$, where P is higher than the $\mathcal{P}_{\text{critical}}$ for the current candidate $\mathcal{H}_{\text{critical}}$. If more buses are available, the number of iterations increases up to 32. Similarly, some of the observed gaps between the **Primal-SSSP** and **Dual-SSSP** are further amplified, with the biggest gap being up to 34.2%. However, such an improvement is not possible in all cases, as the shortest gap only spans 12.7%.

In general, Fig. 20 shows that more buses increase the number of iterations required to find $\mathcal{H}_{\text{critical}}$. On the other hand, when looking at $\mathcal{P}_{\text{critical}}$, it is apparent that more buses reduce the vulnerability, as the impact on P is reduced by 11% between **no-buses** and **five-buses**, while between **five-buses** and **ten-buses**, a further 1.7% of P can be recovered.

Figure 21: Location of $\mathcal{H}_{\text{critical}}$ depending on no-buses/five-buses/ten-buses.

To conclude the detailed assessment of the AM and its primal-dual algorithm, we examine the location of $\mathcal{H}_{\text{critical}}$, depending on the chosen configuration as given in Figure 21. It is apparent, that the number of buses also affects the location of $\mathcal{H}_{\text{critical}}$. In the case of **no-buses**, $\mathcal{H}_{\text{critical}}$ is blocking a section with the highest passenger flow. However, with **five-buses** or **ten-buses**, the location of $\mathcal{H}_{\text{critical}}$ is in the Veraina tunnel, a section where no bus bridging service can be provided (see Fig. 14 for the location of the tunnel).

4.4.4 Using the OM to increase Resilience against one Critical Link

To round off the individual and detailed investigations, we once more assess the OM, but this time we focus on its capability to adapt a given timetable to be more resilient against $\mathcal{H}_{\text{critical}}$. Again, we focus on the test case with $\mathcal{OD}_{\text{artificial}}$, **passenger** trains and **no-conflict**, to avoid any side effects that might obscure the solution. As an input, we use the respective $\mathcal{H}_{\text{critical}}$ for **no-buses**, **five-buses** and **ten-buses** as detected in the previous chapter. Furthermore, we set $\lambda = 1/3$, which means that we reroute one third of the p^{od} which can choose an alternative connection.

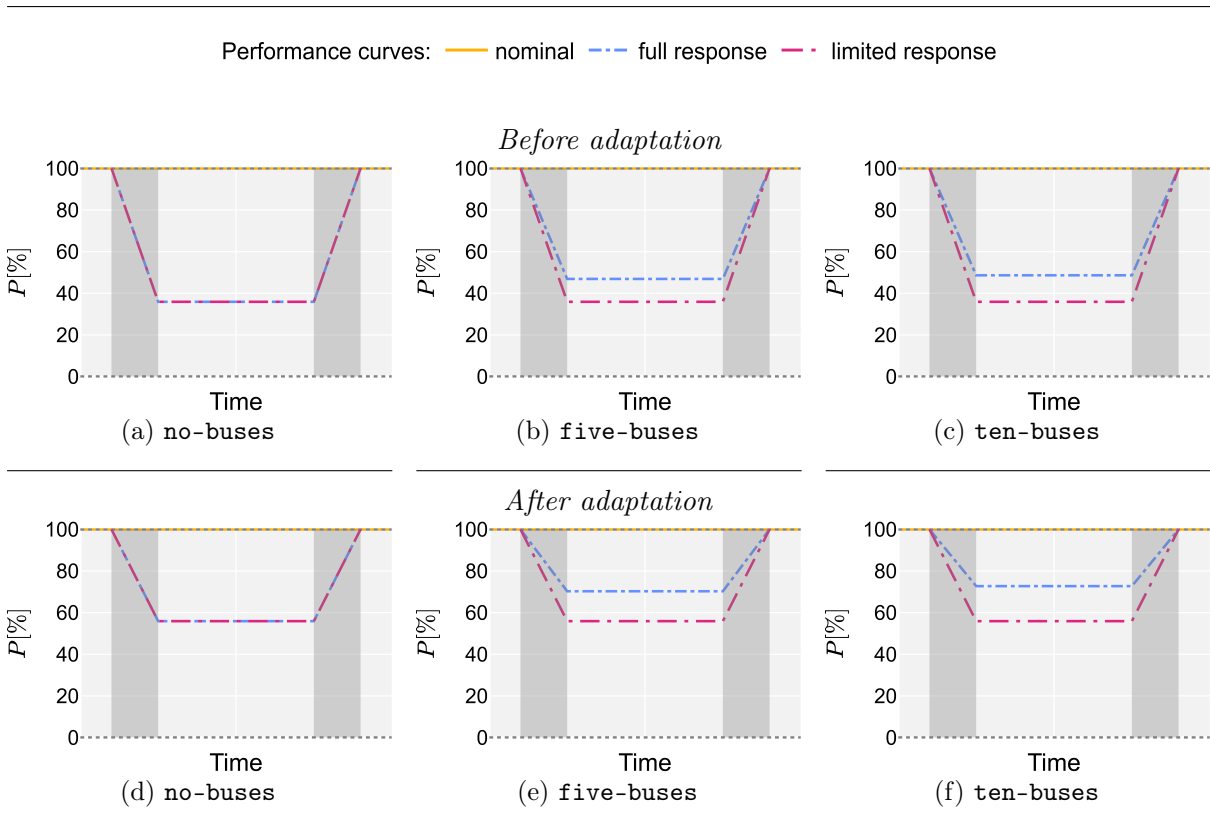
After adapting the SSSP in the OM, we assess the resilience of the adapted solution once more in the AM. Fig. 22 provides the bathtub curves before and after the adaptation. While the x-axis provides the time, the y-axis yields the performance P , such that we can plot three curves, P_{nominal} for the non-disrupted state and P_{critical} once for the case of *limited response* as given by Dual-SSSP and once as *full response* as given by the Primal-SSSP.

For the **no-buses** case, P_{critical} improves from 35.9% to 58.4% Recall, that we observed that $\mathcal{H}_{\text{critical}}$ for the **no-buses** case is located differently than as in the other two cases. In the other

two cases, including alternative connections lead to an improvement of P_{critical} from 46.9% to 73.3% and from 48.6% to 75.9% for **five-bus** and **ten-bus** respectively.

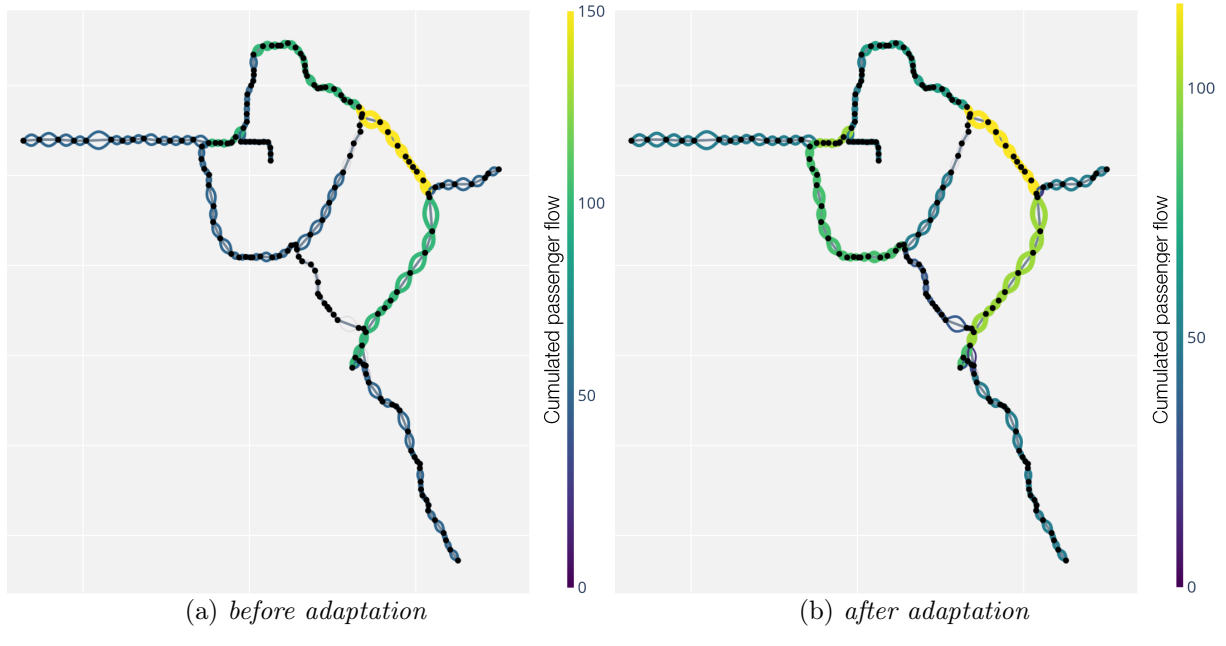
Besides, it is worth mentioning that in none of the adapted cases, an increase in P_{nominal} is observable. Thus, even though we adapt the **M-SSSP** to route some passengers via redundant connections, the resulting timetable provides high-quality connections when passengers can choose their route freely. If they do so, we obtain the same P_{nominal} as initially. Consequently, these solutions represent the ideal case, where we could increase the resilience against $\mathcal{H}_{\text{critical}}$ without sacrificing some of P_{nominal} .

Figure 22: Bathtub curves for $\mathcal{H}_{\text{critical}}$ (available as *html* (Appendix A.3)).



For further insights, we investigate the effect on the flow of passengers, as provided in Fig. 23 for the **five-bus** cases. Fig. 23(a) provides the cumulative flow for all p^{od} passing over one section $h \in \mathcal{H}$, separated by direction which results from the first solution of the **SSSP**. It is worth pointing out, that in this solution, no passengers travel via the Albula tunnel (for the location of the tunnel, we refer to Fig. 14). Fig. 23(b) yields some contrast, as after forcing some passengers on redundant connections and re-solving the **SSSP**, we can observe an increase of passengers on the western parts of the two center loops.

Figure 23: Effect of enforcing redundant connections on passenger flow



4.4.5 Using the OM to increase Resilience against two Simultaneously Disrupted Links

As the last computational assessment part of the detailed examinations, we briefly consider two simultaneous disruptions. In contrast to the previous experiments, we consider a realistic scenario for OD_{subset} with `passenger` trains and set $K = \{2\}$ for the `primal-dual` algorithm. Similar to the previous experiment, `no-buses`, `five-buses` and `ten-buses` are available to respond to the disruption. λ is again set to $1/3$.

The results of the assessment are given in Fig. 24. An initial remark is that the approach is capable of finding the critical disruption even in cases where they occur pairwise. Secondly, as we are considering `passenger` trains and prevent conflicts, we can not find a solution where P_{nominal} is equal to P_{ideal} . However, with a difference in P of 2.0% before and 1.0% after the adaptation, both gaps remain small and acceptable from the viewpoint of `RhB` (`RhB`, 2022). Furthermore, these findings match with the observations in Section 4.4.1. An additional insight given by Fig. 24 is that the impact of a disrupted H_{critical} on P is remarkable, since P_{critical} for the `no-buses` case degrades to 36.1% of P_{ideal} . Adding buses in `five-buses` and `ten-buses` allows a considerable reduction of the impact, as 57.3% and 61.2% of P remain.

Fig. 24 also indicates, the adaptation of the timetable yields no improvement in P_{critical} for `no-buses` and the `five-buses` case, while in the `ten-buses`, a minor gain from 61.2% to 61.4% results. Thus, these results indicate that the approach fails to improve the resilience of the timetable for the detected $\mathcal{H}_{\text{critical}}$. To investigate a conceivable reason for this finding, we briefly consider the locations of $h \in \mathcal{H}_{\text{critical}}$ for the `ten-buses` case, before the timetable is adapted.

Figure 24: Bathtub curves for $\mathcal{H}_{\text{critical}}$ with $K = \{2\}$ (available as *html* (Appendix A.3)).

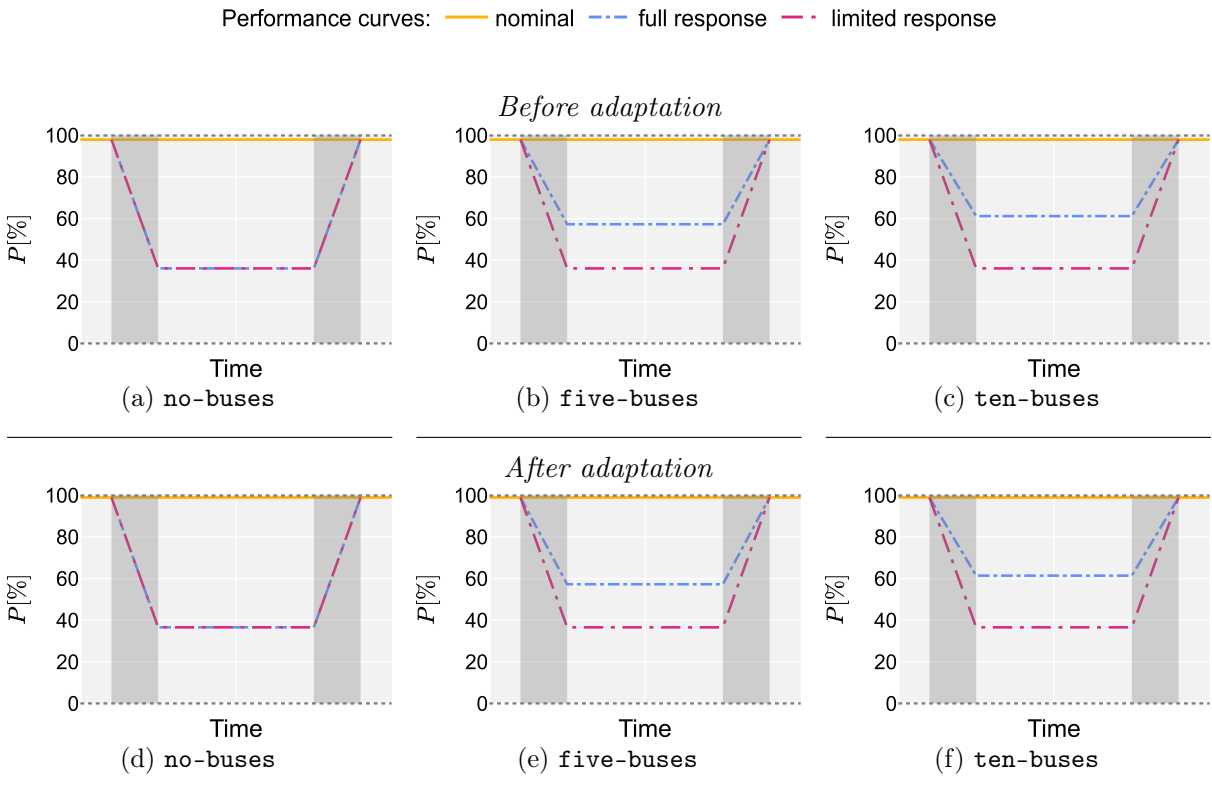
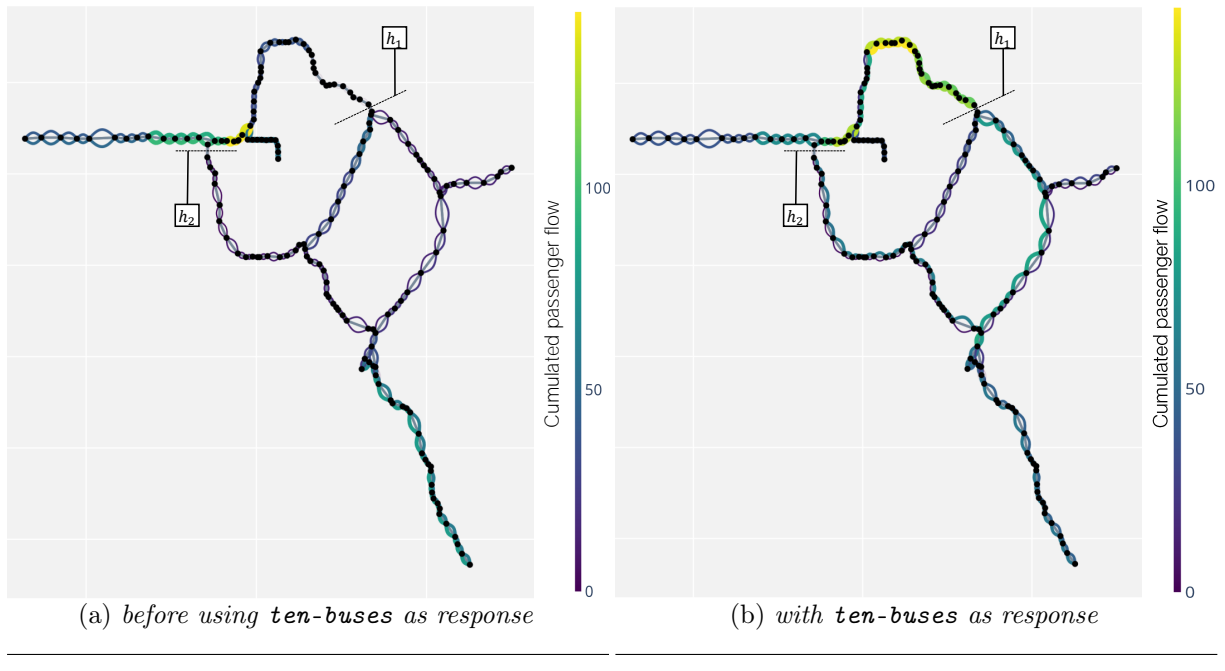


Fig. 25 depicts the locations of $h \in \mathcal{H}_{\text{critical}}$ for the **ten-buses** with the cumulative flow of passengers before (Fig. 25(a)) and after allowing the **Primal-SSSP** to respond with **ten-buses** to the disruption. It is observable, that with **ten-buses**, the flow of passengers generally increases, especially at sections close to one of the $h \in \mathcal{H}_{\text{critical}}$. Furthermore, we can also find a possible deduction, why increasing the resilience of the timetable failed. When we remove the sections $h \in \mathcal{H}_{\text{critical}}$, the network separates in two completely disconnected parts. Thus, rerouting some passengers via redundant connections is not possible.

Figure 25: Passenger flow and locations of $h \in \mathcal{H}$ for two simultaneous disruptions.

We already introduced the possibility of such a scenario when we introduced our approach in Section 3.2. Note that although we can not increase the resilience of the timetable, we still can provide essential insights. One such insight is that using buses can be highly beneficial. Further, we managed to further improve the P_{nominal} in the second run of the OM. However, we cannot increase resilience due to the location of the disruptions, splitting the network in two. When we discussed this result with our partners at RhB, they confirmed that they are aware that such scenarios of $\mathcal{H}_{\text{critical}}$ can occur. Indeed, this is why RhB frequently uses buses as a response to disruptions. Nevertheless, as our approach allows using buses to increase $P_{\text{disrupted}}$ in such a case, RhB confirms its value, even if improving the resilience with redundant routes is not possible (RhB, 2022).

4.5 Computational Experiments to Assess the Iterative Methodology

The second set of assessments expands on the insights of the first set, as we apply both the **OM** and **AM** in the iterative approach. We conduct a total of nine experiments. We allow the *primal-dual* algorithm in the **AM** to select a $k \in K$ with $K = \{1, 2, 3\}$. To allow comparison over all experiments, we limit the scope to **five-buses** as response. We split the experiments in groups of three, where each single experiment in such a group corresponds to a \mathcal{OD} scenario.

In the first group, we use the **passenger** trains and again disable the **S-SSSP** in the **OM**. This configuration is chosen to replicate an instance with low infrastructure utilisation. Consequently, this configuration allows us to assess the methodology under almost *ideal* circumstances, with few side effects to consider. In the second group, we use the instance with only **passenger** trains scenario. Finally, in the third group we include all **passenger** and **cargo**. From this arrangement, we expect the instances to be gradually more challenging to optimise in the **OM**. Hence, we can assess the effect of increasing difficulty in the **OM** on the overall approach.

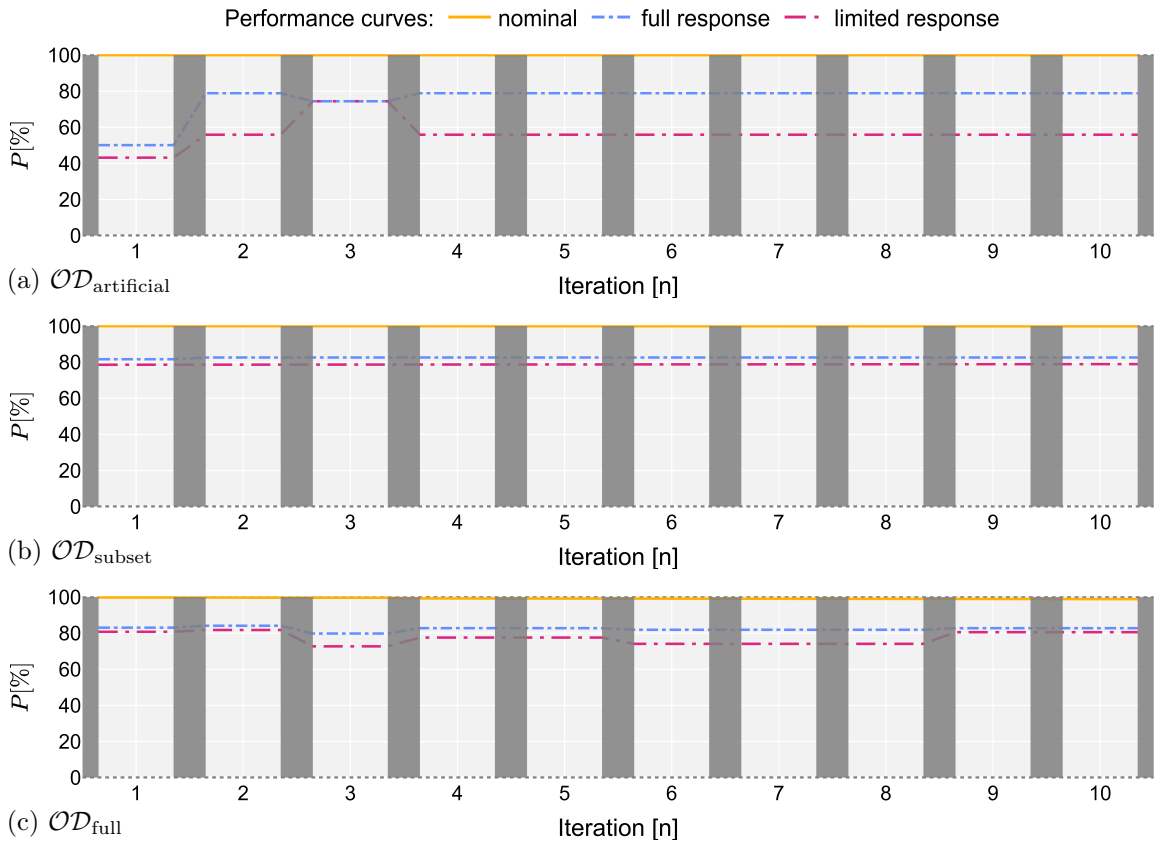
Provided that in previous experiments (Section 4.4.4), $1/3$ for λ has been suitable, we again use $\lambda = 1/3$. Besides, we run each experiment for ten iterations. We opt for ten steps based on two considerations. Firstly, we aim to include enough iterations to capture how P_{nominal} and P_{critical} evolve. Secondly, we expect larger instances with $\mathcal{OD}_{\text{full}}$ to be computationally challenging in the **OM** and **AM**. Given that we aim to provide an approach applicable in practice, a total time of less than 24 hours to set up and run each approach for ten iterations is desired. A time frame of 24 hours has also been approved by **RhB** (**RhB**, 2022).

4.5.1 Assessment with passenger Trains and no-conflict Detection

As introduced, the first assessment of the iterative approach tackles all three \mathcal{OD} scenarios without checking for any conflicts in the **S-SSSP** in the **OM**. We run each scenario for ten iterations. Fig. 26 provides the results in the form of adjacent bathtub curves, where after each assessment in the **AM**, we denote the results of the current iteration. As introduced in Section 3.1.3, $P_{\text{ideal}} = 100\%$ and $P_{\text{worst-case}} = 0\%$ limit the performance range of any bathtub. For each $\mathcal{H}_{\text{critical}}$, we report the P_{nominal} for the non-disrupted timetable with the *nominal* performance curve. In addition, the plots provide the curves for P_{critical} for $\mathcal{H}_{\text{critical}}$ with the *limited response* and the *full response*. While the *full response* yields P_{critical} when all **five-buses** are used in the response, *limited response* holds P_{critical} with only short turning and cancellation as responses, but without the **five-buses**. Note that we display *limited response* mainly for illustrative purposes, as it allows us to assess the effect of using **five-buses** on P_{critical} . However, we generally focus on P_{nominal} and P_{critical} for the *full-response*.

In Fig. 26, we can see that it is possible to solve any of the 20 instances for $\mathcal{OD}_{\text{artificial}}$ and $\mathcal{OD}_{\text{subset}}$ in the **OM** to a difference of less than 1% between P_{nominal} and P_{ideal} . Initially, the solutions of $\mathcal{OD}_{\text{full}}$ do not exceed the 1% difference either. However, the final iteration yields a difference of 1.1%.

Figure 26: Performance P during 10 iterations with **no-conflict** and **five-buses** (available as *html* (Appendix A.3)).



Interestingly in Fig. 26, all three OD yield a solution with the highest P_{critical} at the second iteration when using a full response. The subsequent evolution of P strongly differs for each OD scenario. We see an absolute increase for $OD_{\text{artificial}}$ in Fig. 26(a) of 28.8%, an increase of 1% for OD_{subset} (Fig. 26(b)) and 1.1% OD_{full} (Fig. 26(c)). Although, the improvement for $OD_{\text{artificial}}$ clearly outperforms the two other OD , all three result indicate that we are able to improve the resilience in all three cases. Even in the case of OD_{subset} , with the lowest absolute improvement of 1%, we reduce the vulnerability by more than 1/20 (Fig. 26(b)). Consequently, we also have to consider the initial P_{critical} , as a higher value leaves less room for an absolute improvement in P .

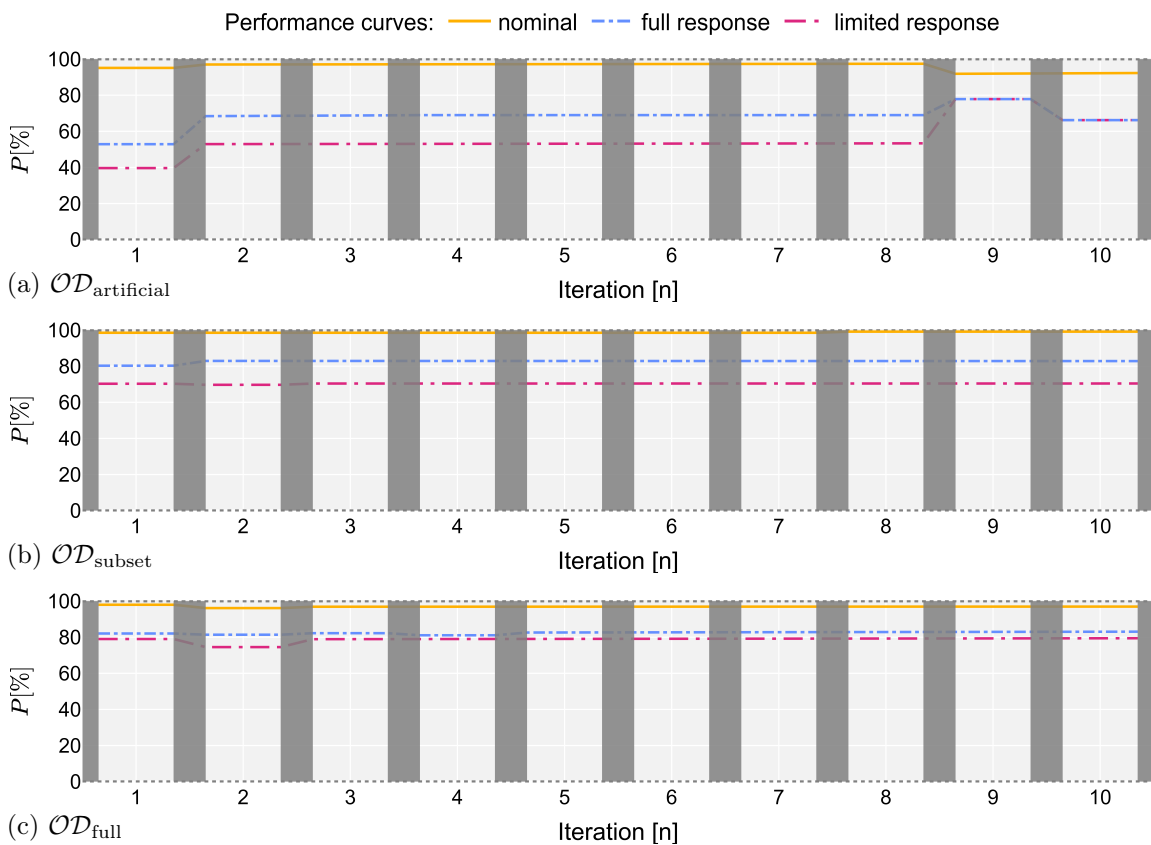
Using **five-buses** adds an additional 23.0%, 3.7% and 2.3% to P_{critical} for each OD scenario, when focusing on the second iteration. Finally, considering the development following the second iteration, the results vary. While for $OD_{\text{artificial}}$, P_{critical} briefly dips at iteration 3, it recovers and stagnates from iteration 4 on. Similarly, OD_{subset} stagnates from iteration 2 on and no more improvements in P_{critical} is observable. A contrasting impression is given by the OD_{subset} scenario in Fig. 26(c), where we observe a degradation of both P_{nominal} and P_{critical} over the iterations. A possible explanation for this observation is that forcing some passengers on redundant connections in the **SSSP** might have caused an over-correction of the passenger flows.

As a possible consequence, the adaptation of the timetable is too drastic and undermines our intentions. However, since we store and display the results for all iterations, we may select any of the available timetables. Thus, we would recommend to use the solution at iteration 2 for the $\mathcal{OD}_{\text{full}}$ in Fig. 26(c).

4.5.2 Assessment with passenger Trains

The second assessment of the iterative approach considers all **passenger** trains. As stated in Section 4.5.1 we iterate ten times and allow **five-buses** as response resources. Fig. 27 provides the results in the form of adjacent bathtub curves.

Figure 27: Performance P during 10 iterations with **passenger** trains and **five-buses** (available as *html* (Appendix A.3)).



It is apparent, that in contrast to the previous runs in Fig. 26, the differences between P_{nominal} and P_{ideal} increases in Fig. 27. This difference supports our insights in Section 4.4.1, and is likely due to the fact, that we now consider conflicts between trains. Presumably, it is more challenging to schedule the **passenger** trains such that P_{nominal} is closer to P_{ideal} .

Further observable is that the differences vary depending on the \mathcal{OD} . For $\mathcal{OD}_{\text{artificial}}$ in Fig. 27(a), the **OM** yields few solutions with P_{nominal} close to P_{ideal} , with the smallest difference being 2.6%.

For $\mathcal{OD}_{\text{subset}}$ in Fig. 27(b), the difference never exceeds 1.9%, with 0.9% as the minimal value at iterations 8 to 10. As we have seen in Section 4.4.1 and Section 4.4.2 $\mathcal{OD}_{\text{full}}$ leads to a more challenging model in **M-SSSP**, highly likely due to the higher number of *od* relations. Thus, a bigger gap between P_{nominal} and P_{ideal} can be anticipated for $\mathcal{OD}_{\text{full}}$ in Fig. 27(c). In fact, the initial P_{nominal} at iteration 1 has an absolute difference of 2% to P_{ideal} , offering a P_{nominal} of 98.0%. However, during the subsequent iterations, P_{nominal} decreases to 96.1% at iteration 2 before recovering to 97.0% from iteration 7 on. When assessing the P_{critical} , we can observe that we manage to improve all three \mathcal{OD} . For $\mathcal{OD}_{\text{full}}$ in Fig. 27(c), the initial P_{critical} is at 82.0%, while eventually at iteration 10, we report 83.0%. However, as also P_{nominal} is lower at iteration 10 (98% to 97%), this improvement comes with a *cost of resilience*.

For $\mathcal{OD}_{\text{artificial}}$ in Fig. 27(a), P_{critical} increases from 52.8% to 69.0% before peaking at 77.8% at iteration 9 and then eventually falling to 66.2%. Again, this erratic behaviour might be due to an over-correction when we modify the **M-SSSP** to include redundant connections. If the aim is to choose the iteration with the highest $P_{\text{disrupted}}$, iteration 9 is the choice. However, this incurs a considerable *cost of resilience*, hence solution 8 might be more preferable. Again, displaying and storing all intermediate solutions is highly beneficial, as it enables a planner to select the most appropriate one.

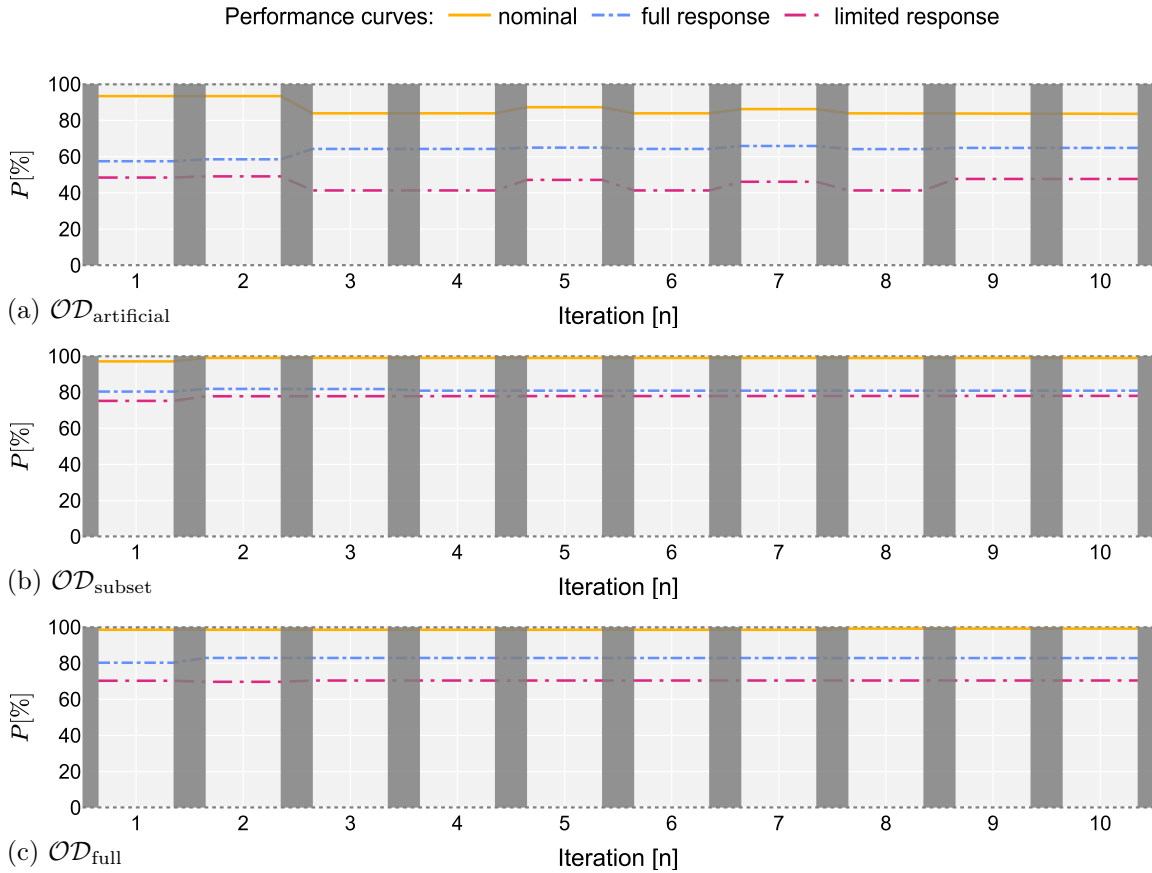
Finally, $\mathcal{OD}_{\text{subset}}$ in Fig. 27(b) provides us with a gradual improvement in P_{critical} of initially 80.3% to 82.9% at iteration 10. As iteration 10 also yields the result with the highest P_{nominal} of 99.1%, iteration 10 presents a solution where we managed to do both, increase P_{nominal} and P_{critical} .

Eventually, Fig. 27 again demonstrates that the response with **five-buses** frequently manages to increase P_{critical} for any \mathcal{OD} . However, there is one exception for $\mathcal{OD}_{\text{full}}$ at iteration 9 in Fig. 27(c), where adding buses did not improve anything. In this case, $\mathcal{H}_{\text{disrupted}}$ is likely affecting infrastructure sections, where no buses can be used to respond to a disruption (i.e, longer tunnels).

4.5.3 Assessment with passenger and cargo Trains

Eventually, we assess the iterative approach with **passenger** and **cargo** trains combined. Given the higher number of trains, we can expect these instances to impose a more significant challenge when compared to previous runs. This assumption is supported by the insights of our experiments in Section 4.4.1, the results of Fuchs *et al.* (2021) which use the same data set and report, that as a considerable fraction of the infrastructure network is single-tracked, finding conflict-free solutions is not trivial. Fig. 28 visualises the evolution of P during the iterations. As expected, the results differ from previous observations.

Figure 28: Performance P during 10 iterations with passenger, cargo and auto-trains (available as *html* (Appendix A.3)).



When we briefly consider P_{nominal} for all instances in Fig. 28, we can clearly distinguish $OD_{\text{artificial}}$ from the other two OD . As given by Fig. 28(a) $OD_{\text{artificial}}$ yields solutions, with a difference between P_{ideal} ($P_{\text{ideal}} = 100\%$) and P_{nominal} of initially 6.4%, evolving to 16.4%. Furthermore, with increasing number of iterations P_{nominal} generally decreases, however some fluctuations are observable. On the contrary, P_{critical} increases from initially 57.5% to 65.9% at iteration 7. Note that this increase comes with a *cost of resilience*, as after iteration 2, P_{critical} increases but P_{nominal} never reaches the initial P_{nominal} again. The decision, which iteration to select is not straightforward, but when prioritising P_{nominal} over $P_{\text{disrupted}}$, one would choose iteration 2. If prioritising vice versa, iteration 7 yields the best choice. However, any solution in between might also be appropriate, as for example iteration 5 offers a trade-off.

In contrast to $OD_{\text{artificial}}$ in Fig. 28(a), the result of OD_{subset} in Fig. 28(b) reveals a far less erratic picture. The iterative approach manages to improve P_{critical} from 80.3% to 81.7% at the third iteration. Interestingly, also P_{nominal} can be improved from 97.1% to 99.0% from the first to the third iteration. Subsequently, the efforts of enforcing alternative connections seems to lead to a decrease of P_{critical} to 80.9% at iteration 10. Interestingly, P_{nominal} is only slightly affected, falling by 0.1% to 88.9%.

Finally, $\mathcal{OD}_{\text{full}}$ in Fig. 28(c) provides a similar picture to $\mathcal{OD}_{\text{subset}}$ in Fig. 28(b). However, the difference between P_{nominal} and P_{ideal} is initially substantially bigger. P_{nominal} starts at 95.5% and gradually decreases until iteration 5 to a value of 93.2%, before recovering to 97.2% throughout the remaining iterations. $P_{\text{disrupted}}$ increases from 80.1% to 82.2% until iteration 3, before decreasing again. However, similar to P_{nominal} , P_{critical} also recovers from iteration 56 on, levelling off at 81.7% at iteration 10.

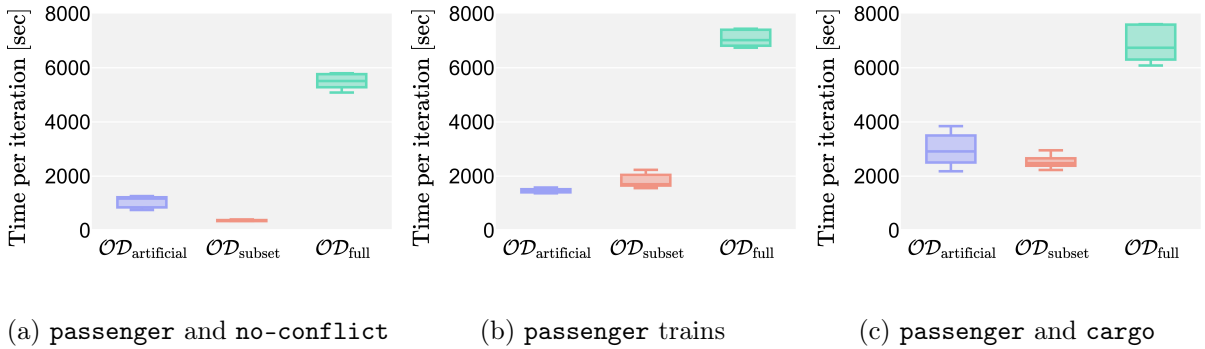
When comparing all three instances in Fig. 28, it is apparent that the instance with $\mathcal{OD}_{\text{artificial}}$ seems to be much more challenging, given the differences between P_{ideal} and P_{nominal} . One possible cause is the difference on how the *od*-relations of the considered \mathcal{OD} are structured. Recall that $\mathcal{OD}_{\text{artificial}}$ is specifically tailored to emphasise passengers travelling comparably long distances with multiple transfers. Thus, this instance is likely more affected by the higher number of trains since this reduces the freedom to arrange trains for minimal transferring times. On the contrary, the real-life based \mathcal{OD} have fewer passengers, which require more than one train to reach their destination. Therefore, those two instances are possibly less drastically affected by the higher number of trains. Furthermore, although we aimed to test our methodology with three more challenging instances, we see that it is still capable of providing solutions, where we manage to increase $P_{\text{disrupted}}$.

4.5.4 Computation Time per Iteration

To summarise the iterative assessment results, we provide some insights on the computation time. However, our approach consists of an ensemble of MIP-models. Thus, we do not report computation times individually for the specific models but focus on providing an overview by summarising the time required for one iteration in the experiments before. Note that these times also include any overhead, such as setting up the models when iterating between the OM and AM.

Fig. 29 provides an overview on the required computation time per iteration. One interesting fact is that even though, $\mathcal{OD}_{\text{artificial}}$ consists of only eight *od*-relations, it requires more computation time than $\mathcal{OD}_{\text{subset}}$ in some cases, although the latter has more than twice the number of *od*-relations. Again, the different structure of the demand might be a reason. A second observation is that the time per iteration generally increases with increasing difficulty (i.e., more conflicts between trains). Also apparent is that all instances with $\mathcal{OD}_{\text{full}}$ require considerably more time.

Figure 29: Box-plots to summarise the computation time per iteration (available as *html* (Appendix A.3)).



For a more precise comparison, we additionally report the average computation time in Table 10. Using these, we can also verify that we managed to remain within the defined time window of 24 hours in Section 4.5. Indeed, since none of the average computation times exceeds 2 hours, we can summarise that for 10 iterations, we remain within the 24 hours.

Table 10: Average computation time in seconds for one iteration.

	$OD_{artificial}$	OD_{subset}	OD_{full}
passenger & no-conflict	1044.7	358.8	5500.5
passenger	1469.3	1817.8	7084.9
passenger & cargo	2948.5	2528.2	6908.9

5 Conclusion and Future Work

As a last element of the thesis, we will draw a conclusion in Section 5.1 and address the RQs formulated in Section 1.3, before we provide some future research directions in Section 5.2.

5.1 Conclusion

We use the research questions posed in Section 1.3 as a guide to draw a conclusion. Initially, we address the RQ before we move on towards the SQ 1 to SQ 5.

(Main) Research Question (RQ) How can we integrate the assessment and enhancement of resilience in the strategical and tactical timetabling for railways?

Based on our literature review, we identified a research gap between the current state-of-the-art in the railway planning process and proposed *network-*, and *scenario-based* approaches to address disruptions. Consequently, we proposed a scheme that allows creating timetables *from scratch* while identifying and addressing the critical disruption. To unify the objective throughout the process, we opted for a demand centred performance metric based on generalised travel time. Furthermore, we included passenger routing, track assignment and rolling stock circulation. As a result, we faced a challenging task.

Consequently, we introduced the TSS, a structure allowing us to reduce the complexity drastically. The concept of TSS is to restrict a train to time windows, similar to the TPE (Wang *et al.*, 2020) and the SI (Caimi *et al.*, 2011b). Using the TSS, we proposed a decomposition scheme based on logical-Benders decomposition (Hooker and Ottosson, 2003), enabling us to split the task of timetabling in a master and a sub-problem. While we solved the former with a commercial MIP solver, we formulated the latter as SAT.

Furthermore, as we consider the context of strategical planning, we can not rely on predefined disruption scenarios. Thus, we proposed a *primal-dual* algorithm that identifies the critical disruption under the consideration of all available response methods. In doing so, we combined the advances in *network-*, and *scenario-based* approaches to obtain an accurate value of the system performance.

Eventually, we arranged our work in two modules, which interact iteratively during the timetabling process. To assess our proposition, we conduct a case study based on concrete, real-life data provided by RhB. Throughout the case study, we conducted several experiments organised in two groups.

In the first set of experiments, we evaluated each module individually. Our experiments concluded that solving the timetabling model directly with the commercial solver is only appropriate for small instances. However, we quickly obtained high-quality solutions by proposing and using a fix-

and-dive heuristic. For challenging models, the heuristic used only 2/3 of the given 7200 seconds computational time while providing equal or higher quality results compared to solving the model directly. Furthermore, for non-trivial instances, directly solving the model did not lead to termination within the time limit and yields solutions with an optimality gap.

In subsequent assessments, the case study shows that the proposed approach can increase the resilience of the timetable such that the performance during the critical disruption increases from 35.9% to 58.4%, increasing the remaining performance by a factor of 1.63. Although this result relies on an artificially generated demand, the insight is valuable. Another valuable insight is that our procedure to identify and assess critical disruption also works for simultaneously occurring disruptions. This capability is a criterion worth striving for, according to future research directions outlined by [Bešinović \(2020\)](#).

Eventually, we applied the iterative approach to nine different instances. In addition, we gradually increased the number of trains within the network for further insights. We observed that our proposition performs as intended under ideal circumstances, although the results are affected by the considered demand scenario. Frequently, the solutions differ in both resilience against the critical disruption and performance under regular (nominal) circumstances. Thus, selecting the appropriate solution sometimes requires a trade-off. Here, we can rely on an advantage our approach provides, because we store and present all solutions generated during the iterations. Based on this overview, a practitioner can select and implement the most suitable solution.

To summarise, we can successfully integrate the assessment and enhancement of resilience in railways' strategical and tactical timetabling with our proposed iterative approach, which finds and responds to the most critical disruption while solving the timetabling task. Thus, we proceed to the [SQ 1](#).

[SQ 1](#) What state-of-the-art metrics/measures are suitable to assess the resilience of a railway timetable during timetabling?

We settled on using generalised travel time, as it is suitable to optimise a timetable ([Caimi et al., 2017](#)) and also to assess its performance during disruptions ([Zhu and Goverde, 2020b](#)). Therefore, we developed a performance metric based on generalised travel time to evaluate the nominal and disrupted performance. Given that the metric is performance-based, we could follow the recommendations of both [Zhou et al. \(2019\)](#) and [Bešinović \(2020\)](#). Both emphasise that performance-based metrics are the preferable choice. Furthermore, since we were using a demand centred perspective, we addressed a future research direction outlined by [Bešinović \(2020\)](#), recommending to consider demand centred resilience. Consequently, we can state that our performance metric follows the state-of-the-art recommendations while being suitable to assess performance from a passenger centred perspective.

SQ 2 How can we characterise disruptions such that we can assess them during the timetabling process?

As we considered disruptions in the strategical and tactical timetabling process, we established that we could not rely on predefined scenarios. Therefore, we proposed detecting the most critical disruption with an optimisation model. Besides, we limited the scope to the complete blockage of sections between stations, similar to [Szymula and Bešinović \(2020\)](#) and [Borecka and Bešinović \(2021\)](#). Furthermore, we constrained our evaluation of disruptions to the response phase, when the system is in its steady disrupted state. A key argument for this limitation is the insight of [Zhu and Goverde \(2020a\)](#). [Zhu and Goverde \(2020a\)](#) show that the start and end of disruption affect the transformation of the system from the nominal to the disrupted state. Further, existing work that considers a similar time context also omits the phases of survivability and recovery ([Szymula and Bešinović, 2020](#); [Borecka and Bešinović, 2021](#)). To summarise, we managed to characterise disruptions by limiting the scope towards complete blockages and the response phase. Furthermore, the *primal-dual* algorithm developed in Section 3.4 allowed us to detect and accurately assess single or simultaneous critical disruption without relying on predefined scenarios.

SQ 3 How to mathematically model the requirements and resources of the different stakeholders to include them as constraints?

To provide a holistic indication of performance and resilience, we included the requirements of all concerned stakeholders. As stakeholders, we identified the passengers as users as well as the [TOC](#) as operator and the [IM](#) as an infrastructure provider. Consequently, we integrated routing passengers, circulating vehicles and assigning tracks into timetabling. Existing research demonstrates that integrating a selection of these tasks already leads to challenging models ([Polinder et al., 2021](#); [Schiewe, 2020](#); [Pätzold et al., 2017](#); [Fuchs et al., 2021](#)). Consequently, we proposed to break down the complexity by using [TSS](#), which restrict a scheduled train to run within predefined time slots. The idea is similar to [TPE](#) used by [Wang et al. \(2020\)](#) and the [SI](#) of [Caimi et al. \(2011b\)](#) and allows to formalise the requirements of [TOC](#). Furthermore, we could use the [TSS](#) to apply a logical-Benders decomposition to decompose the timetabling task into a master- and a sub-problem. To tackle the master problem, we developed a network structure. The structure allowed us to select a set of [TSSs](#) which minimise the generalised travel time for passengers while the vehicle circulation is ensured. The sub-problem receives a set of selected [TSSs](#) and solves the resulting timetabling problem, including track assignment. If no timetable for a given set of [TSSs](#) exists, we add a constraint to the master-problem, thereby banning the set of conflicting [TSSs](#). We used an iterative approach and iterated between optimising and assessing the timetable. To ensure a consistent perspective, all of the developed models in the iterative approach rely on the [TSS](#). Furthermore, [TSSs](#) are also capable of modelling buses as a response to a disruption.

SQ 4 What are the benefits when using the developed optimisation approach to enhance the resilience of an existing railway system?

We conducted a case study to evaluate our proposition. Our case study reflects the concrete, real-life network of a Swiss railway operator, the [RhB](#). Given that the network consists primarily of single track sections, it is known to be a challenging environment for finding conflict free timetables ([Fuchs et al., 2021](#)). Our initial experiments in [Section 4.4](#) confirmed this insight, as optimising with the proposed model was challenging with increasing problem size. However, as we also proposed a heuristic procedure to obtain high-quality solutions in a shorter time (see [Section 3.3.6](#) and [Section 4.1](#)), we can overcome this challenge. A further crucial insight of the initial experiments is that the proposed procedure to adapt a timetable to increase its resilience against the critical disruption was applicable ([Section 4.4.3](#)). Consequently, we conducted a second set of experiments, where we assessed the proposed iterative approach in its entirety. The experiments confirmed the insights of the first set of experiments, as generally, the approach enabled to increase the resilience of a timetable throughout the process. However, the results varied depending on the demand structure and infrastructure utilisation. While we managed to improve resilience in cases of low infrastructure usage, in some cases with more demand and higher infrastructure utilisation, finding a high-quality timetable imposed a more demanding challenge. Nevertheless, the iterative approach improved resilience for any of the submitted instances. However, in some cases, this gain came at the cost of sacrificed nominal performance. Thus, to maximise the value of our proposed approach, we provided all generated solutions to the planners. They may then decide, which solution would be most suitable. Apart from these benefits, our approach revealed insights into the location of critical disruptions. Furthermore, the approach was helpful to assess the effectiveness of different responses, i.e., varying number of buses. Thus, we can summarise that our proposition supports planners with timetables with improved resilience while also guiding their attention towards critical disruptions and providing insights into the effectiveness of specific measures to address a disruption.

With these answers to our research questions, we finalise the conclusion and complete the chapter with recommendations for future work.

5.2 Future Work

As the final element of this thesis, we provide some directions for future work. We structure these directions in three sections. Initially, we propose some future work considering the [OM](#) in [Section 5.2.1](#), before we focus the [AM](#) in [Section 5.2.2](#) and the consider both jointly in [Section 5.2.3](#).

5.2.1 Future Work on the [OM](#)

Although the [OM](#) met our expectations, we identified some directions for future research. One of these directions is naturally improving the performance by introducing further heuristics that reduce the computation time. Another approach to improve its performance might be to use an

alternative formulation that relies on a path based formulation to route passengers, as suggested by [Szymula and Bešinović \(2020\)](#). Although we specifically chose an arc-based formulation, it remains to show if a path-based approach may be superior. One such case could be, when aiming to improve a given starting solution obtained by the [S-SSSP](#).

A second point worth investigating is to provide alternative means to adjust a timetable. Currently, we rely on alternative routes that are redundant under normal circumstances. However, as we have seen in [Section 4.4.5](#), introducing redundant routes is sometimes out of scope, as the disruptions separate the network in two components. Hence, introducing further means to increase resilience could help here. A promising approach might be integrating the system in the local bus network, as proposed by [Jin *et al.* \(2014\)](#).

Eventually, another path worth pursuing is assessing the sacrifice due to using [TSS](#), compared to an approach that considers integrated routing of passengers during timetabling, such that we can assess the difference. However, for a fair comparison, the approach to compare with should also route trains during timetabling, which is currently not done in any of the proposed models ([Schiewe, 2020](#); [Polinder *et al.*, 2021](#)). Thus, such an evaluation is not straightforward.

5.2.2 Future Work on the AM

The [AM](#) also provides some interesting future research. Firstly, one might want to explore further and utilise the possibilities to include more responses. Although we outlined some of these already in the [Primal-SSSP](#) in [Section 3.4.1](#), we did not assess them in the case study. Consequently, assessing further measures yields additional insights.

Another possible adaptation is to adjust the normalisation procedure used in the [primal-dual](#) algorithm. Currently, we use the number of simultaneously disrupted sections. However, this normalisation implied that the critical disruption was always one single disruption during the iterative assessments in the second round, likely due to the considered network and the normalisation factor combined.

Similarly, one could also try to include an alternative way of providing the constraint of how many links may be blocked. This alternative constraint could then also account for the probability of failure, as, for example, done by [Zhu and Goverde \(2017\)](#). Consequently, the criticality of disruption is characterised by performance and probability of occurrence. Naturally, it is also worth considering implementing other forms of disruptions. For instance, partial blockages could block some of the tracks but not the complete section. Likewise, one could introduce disruptions where trains have to reduce the speed and thus occupy the disrupted section for a longer time.

5.2.3 Future Work on the Iterative Approach using the OM and AM

Eventually, when considering the complete iterative approach, one future research direction is an extension of the [TSS](#) and [SSSP](#) to a non-periodic case. With this extension, one can then

assess the entire day rather than limiting the scope to the duration of one period. Similarly, one could try to adapt the **SSSP** such that both assessment and optimisation are included in one model. This way, one could also remove the need to specify a λ that forces some passengers on redundant routes. Since the value of λ has to be defined manually, investigating the effect of adjusting the λ on the iterative solutions is also a viable future working path.

6 References

- Albrecht, T., A. Binder and C. Gassel (2013) Applications of real-time speed control in rail-bound public transportation systems, *IET Intelligent Transport Systems*, **7** (3) 305–314.
- Audemard, G. and L. Simon (2018) On the glucose sat solver, *International Journal on Artificial Intelligence Tools*, **27**, 1840001, 02 2018.
- Bababeik, M., N. Khademi, A. Chen and M. M. Nasiri (2017) Vulnerability Analysis of Railway Networks in Case of Multi-Link Blockage, *Transportation Research Procedia*, **22** (2016) 275–284, ISSN 23521465.
- Bešinović, N., R. M. P. Goverde, E. Quaglietta and R. Roberti (2016) An integrated micro-macro approach to robust railway timetabling, *Transportation Research Part B: Methodological*, **87**, 14–32, ISSN 01912615.
- Bešinović, N. and C. Szymula (2021) Estimating impacts of covid19 on transport capacity in railway networks, *European Journal of Transport and Infrastructure Research*, **21** (1).
- Besinovic, N., Y. Wang, S. Zhu, E. Quaglietta, T. Tang and R. M. P. Goverde (2019) Integrated train and passenger disruption management for urban railway lines, *2019 IEEE Intelligent Transportation Systems Conference, ITSC 2019*, 3182–3187.
- Bešinović, N., B. Widarno and R. M. P. Goverde (2020) Adjusting freight train paths to infrastructure possessions, paper presented at the *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, 1–6.
- Bešinović, N. (2020) Resilience in railway transport systems: a literature review and research agenda, *Transport Reviews*, **40** (4) 457–478.
- Biere, A., M. Heule and H. van Maaren (2009) *Handbook of satisfiability*, vol. 185, IOS press.
- Borecka, J. T. and N. Bešinović (2021) Scheduling multimodal alternative services for managing infrastructure maintenance possessions in railway networks, *Transportation Research Part B: Methodological*, **154**, 147–174, ISSN 0191-2615.
- Borndörfer, R., M. Grötschel and M. E. Pfetsch (2007) A column-generation approach to line planning in public transport, *Transportation Science*, **41** (1) 123–132, ISSN 15265447.
- Borndörfer, R., H. Hoppmann and M. Karbstein (2017) Passenger routing for periodic timetable optimization, paper presented at the *Public Transport*, vol. 9, 115–135, 7 2017, ISSN 1866-749X.
- Borndörfer, R., N. Lindner and S. Roth (2020) A concurrent approach to the periodic event scheduling problem, *Journal of Rail Transport Planning and Management*, **15**, 140–159, ISSN 22109706.

- Bull, S., J. Larsen, R. M. Lusby and N. J. Rezanova (2019) Optimising the travel time of a line plan, *4or*, **17** (3) 225–259, ISSN 16142411.
- Burggraeve, S., S. H. Bull, P. Vansteenwegen and R. M. Lusby (2017) Integrating robust timetabling in line plan optimization for railway systems, *Transportation Research Part C: Emerging Technologies*, **77**, 134–160, ISSN 0968090X.
- Cacchiani, V., D. Huisman, M. Kidd, L. Kroon, P. Toth, L. Veelenturf and J. Wagenaar (2014) An overview of recovery models and algorithms for real-time railway rescheduling, *Transportation Research Part B: Methodological*, **63**, 15–37, ISSN 0191-2615.
- Caimi, G., M. Fuchsberger, M. Laumanns and K. Schüpbach (2011a) Periodic railway timetabling with event flexibility, *Networks*, **57** (1) 3–18, 1 2011, ISSN 00283045.
- Caimi, G., L. Kroon and C. Liebchen (2017) Models for railway timetable optimization: Applicability and applications in practice, *Journal of Rail Transport Planning & Management*, **6** (4) 285–312, ISSN 22109706.
- Caimi, G., M. Laumanns, K. Schüpbach, S. Wörner and M. Fuchsberger (2011b) The periodic service intention as a conceptual framework for generating timetables with partial periodicity, *Transportation Planning and Technology*, **34** (4) 323–339, ISSN 03081060.
- Conforti, M., G. Cornuéjols, G. Zambelli *et al.* (2014) *Integer programming*, vol. 271, Springer.
- Csardi, G. and T. Nepusz (2006) The igraph software package for complex network research, *InterJournal*, **Complex Systems**, 1695.
- de Fabris, S., G. Longo, G. Medeossi and R. Pesenti (2014) Automatic generation of railway timetables based on a mesoscopic infrastructure model, *Journal of Rail Transport Planning & Management*, **4** (1) 2–13, ISSN 2210-9706.
- Dekker, M. M., R. N. van Lieshout, R. C. Ball, P. C. Bouman, S. C. Dekker, H. A. Dijkstra, R. M. P. Goverde, D. Huisman, D. Panja, A. A. M. Schaafsma and M. van den Akker (2021) A next step in disruption management: combining operations research and complexity science, *Public Transport*, Feb 2021, ISSN 1613-7159.
- Friedrich, M., M. Hartl, A. Schiewe and A. Schöbel (2017) Integrating Passengers' Assignment in Cost-Optimal Line Planning, paper presented at the *17th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2017)*, vol. 59 of *OpenAccess Series in Informatics (OASICS)*, 5:1–5:16, Dagstuhl, Germany, ISBN 978-3-95977-042-2, ISSN 2190-6807.
- Fuchs, F. and F. Corman (2019) An open toolbox for integrated optimization of public transport, paper presented at the *2019 6th International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, 1–7.

- Fuchs, F., A. Trivella and F. Corman (2021) Enhancing the interaction of railway timetabling and line planning with infrastructure awareness, *SSRN Electronic Journal*, 1–29.
- Gattermann, P., P. Großmann, K. Nachtigall and A. Schöbel (2016) Integrating passengers' routes in periodic timetabling: a sat approach, paper presented at the *16th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2016)*.
- Gattermann, P., J. Harbering and A. Schöbel (2017) Line pool generation, *Public Transport*, **9** (1-2) 7–32, ISSN 16137159.
- Ghaemi, N., O. Cats and R. M. P. Goverde (2017a) A microscopic model for optimal train short-turnings during complete blockages, *Transportation Research Part B: Methodological*, **105**, 423–437, ISSN 0191-2615.
- Ghaemi, N., O. Cats and R. M. P. Goverde (2017b) Railway disruption management challenges and possible solution directions, *Public Transport*, **9** (1-2) 343–364, ISSN 16137159.
- Giacco, G. L., D. Carillo, A. D'Ariano, D. Pacciarelli and Ángel G. Marín (2014) Short-term rail rolling stock rostering and maintenance scheduling, *Transportation Research Procedia*, **3**, 651–659, ISSN 2352-1465. 17th Meeting of the EURO Working Group on Transportation, EWGT2014, 2-4 July 2014, Sevilla, Spain.
- Goemans, M. X. and D. P. Williamson (1997) The primal-dual method for approximation algorithms and its application to network design problems, *Approximation algorithms for NP-hard problems*, 144–191.
- Goerigk, M. and C. Liebchen (2017) An improved algorithm for the periodic timetabling problem, paper presented at the *17th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems*, 1.
- Goerigk, M. and M. Schmidt (2017) Line planning with user-optimal route choice, *European Journal of Operational Research*, **259** (2) 424–436, ISSN 03772217.
- Goossens, J. W., S. Van Hoesel and L. Kroon (2006) On solving multi-type railway line planning problems, *European Journal of Operational Research*, **168** (2 SPEC. ISS.) 403–424, ISSN 03772217.
- Großmann, P. (2016) Satisfiability and Optimization in Periodic Traffic Flow Problems, Ph.D. Thesis, Technische Universität Dresden.
- Großmann, P. (2011) Polynomial reduction from pesp to sat, *Technical Report*, Knowledge Representation and Reasoning Group, TU Dresden.

- Gurobi Optimization, LLC (2022) Gurobi Optimizer Reference Manual, <https://www.gurobi.com>.
- Herrigel, S., M. Laumanns, J. Szabo and U. Weidmann (2018) Periodic railway timetabling with sequential decomposition in the PESP model, *Journal of Rail Transport Planning and Management*, **8** (3-4) 167–183, ISSN 22109706.
- Hooker, J. N. (2007) Planning and scheduling by logic-based benders decomposition, *Operations research*, **55** (3) 588–602.
- Hooker, J. N. and G. Ottosson (2003) Logic-based benders decomposition, *Mathematical Programming*, **96** (1) 33–60.
- Ignatiev, A., A. Morgado and J. Marques-Silva (2018) PySAT: A Python toolkit for prototyping with SAT oracles, paper presented at the *SAT*, 428–437.
- Jin, J. G., L. C. Tang, L. Sun and D. H. Lee (2014) Enhancing metro network resilience via localized integration with bus services, *Transportation Research Part E: Logistics and Transportation Review*, **63**, 17–30, ISSN 13665545.
- Kroon, L. G., L. W. Peeters, J. C. Wagenaar and R. A. Zuidwijk (2014) Flexible connections in PESP models for cyclic passenger railway timetabling, *Transportation Science*, **48** (1) 136–154, ISSN 15265447.
- Kümmling, M., P. Großmann, K. Nachtigall, J. Opitz and R. Weiß (2015) A state-of-the-art realization of cyclic railway timetable computation, *Public Transport*, **7** (3) 281–293, ISSN 16137159.
- Liebchen, C. and R. H. Möhring (2004) The Modeling Power of the Periodic Event Scheduling Problem: Railway Timetables — and Beyond, in *Computer-aided Systems in Public Transport*, 117–150, Springer Berlin Heidelberg, ISBN 9783540733119.
- Liebchen, C. and L. Peeters (2009) Integral cycle bases for cyclic timetabling, *Discrete Optimization*, **6** (1) 98–109, ISSN 15725286.
- Liffiton, M. H. and J. C. Maglalong (2012) A cardinality solver: More expressive constraints for free, paper presented at the *International Conference on Theory and Applications of Satisfiability Testing*, 485–486.
- Louwerse, I. and D. Huisman (2014) Adjusting a railway timetable in case of partial or complete blockades, *European Journal of Operational Research*, **235** (3) 583–593, ISSN 0377-2217.
- Luan, X., J. Miao, L. Meng, F. Corman and G. Lodewijks (2017) Integrated optimization on train scheduling and preventive maintenance time slots planning, *Transportation Research Part C: Emerging Technologies*, **80**, 329–359, 7 2017, ISSN 0968-090X.

- Lusby, R. M., J. Larsen and S. Bull (2018) A survey on robustness in railway planning, *European Journal of Operational Research*, **266** (1) 1–15, ISSN 0377-2217.
- Lübbecke, M., C. Puchert, P. Schiewe and A. Schöbel (2018) Integrating line planning, timetabling and vehicle scheduling, *Proceedings of the Conference for Advanced Systems in Public Transport - CASPT*, **15** (12).
- Mattsson, L.-G. and E. Jenelius (2015) Vulnerability and resilience of transport systems – a discussion of recent research, *Transportation Research Part A: Policy and Practice*, **81**, 16–34, ISSN 0965-8564. Resilience of Networks.
- Michaelis, M. and A. Schöbel (2009) Integrating line planning, timetabling, and vehicle scheduling: A customer-oriented heuristic, *Public Transport*, **1** (3) 211–232, ISSN 1866749X.
- Pätzold, J., A. Schiewe, P. Schiewe and A. Schöbel (2017) Look-ahead approaches for integrated planning in public transportation, *OpenAccess Series in Informatics*, **59** (17) 1–16, ISSN 21906807.
- Peeters, L. W. P. (2003) *Cyclic railway timetable optimization*, vol. 22, Erasmus University Rotterdam, Rotterdam, ISBN 90-5892-042-9.
- Peeters, M. and L. Kroon (2008) Circulation of railway rolling stock: a branch-and-price approach, *Computers and Operations Research*, **35** (2) 538–556, ISSN 03050548.
- Polinder, G.-J. (2020) New models and applications for railway timetabling, Ph.D. Thesis, Erasmus University Rotterdam.
- Polinder, G. J., M. Schmidt and D. Huisman (2021) Timetabling for strategic passenger railway planning, *Transportation Research Part B: Methodological*, **146**, 111–135, ISSN 01912615.
- RhB (2005) Geschäftsbericht 2004, *Technical Report*, Rhätische Bahn (RhB), Chur.
- RhB (2021a) *Rolling Stock*, Rhätische Bahn (RhB), <https://www.rhb.ch/en/company/key-figures/rolling-stock> (accessed 2021-12-07).
- RhB (2021b) *Streckennetz*, Rhätische Bahn (RhB), <https://www.rhb.ch/de/service-souvenirs/streckennetz> (accessed 2021-12-04).
- RhB (2022) technical discussion, Feb. 13 2022.
- Rijden de Treinen (2022) *Statistics*, Rijden de Treinen, <https://www.rijdendetreinen.nl/en/statistics>, (accessed 2022-25-25).
- Robenek, T., Y. Maknoon, S. S. Azadeh, J. Chen and M. Bierlaire (2016) Passenger centric train timetabling problem, *Transportation Research Part B: Methodological*, **89**, 107–126, ISSN 01912615.

- SBB (2021) *Timetable*, Schweizerische Bundesbahnen (SBB), <https://www.sbb.ch/en/timetable.html> (accessed 2021-12-07).
- Schiewe, P. (2020) *Integrated optimization in public transport planning*, vol. 160, Springer, ISBN 9783030462697.
- Schmidt, M. and A. Schöbel (2015a) The complexity of integrating passenger routing decisions in public transportation models, *Networks*, **65** (3) 228–243, ISSN 10970037.
- Schmidt, M. and A. Schöbel (2015b) Timetabling with passenger routing, *OR Spectrum*, **37** (1) 75–97, ISSN 14366304.
- Schöbel, A. (2017) An eigenmodel for iterative line planning, timetabling and vehicle scheduling in public transportation, *Transportation Research Part C: Emerging Technologies*, **74**, 348–365, ISSN 0968090X.
- Schöbel, A. and S. Scholl (2006) Line Planning with Minimal Traveling Time, paper presented at the *ATMOS 2005 5th Workshop on Algorithmic Methods and Models for Optimization of Railways*.
- Schöbel, A. (2012) Line planning in public transportation: models and methods, *OR Spectrum*, **34** (3) 491–510, 7 2012, ISSN 1436-6304.
- Serafini, P. and W. Ukovich (1989) A Mathematical Model for Periodic Scheduling Problems, *SIAM Journal on Discrete Mathematics*, **2** (4) 550–581, ISSN 0895-4801.
- Smith, J. C., M. Prince and J. Geunes (2013) *Modern Network Interdiction Problems and Algorithms*, 1949–1987, Springer New York, New York, NY, ISBN 978-1-4419-7997-1.
- Smith, J. C. and Y. Song (2020) A survey of network interdiction models and algorithms, *European Journal of Operational Research*, **283** (3) 797–811, ISSN 03772217.
- Sparing, D. and R. M. P. Goverde (2017) A cycle time optimization model for generating stable periodic railway timetables, *Transportation Research Part B: Methodological*, **98** (434) 198–223, ISSN 01912615.
- Szymula, C. and N. Bešinović (2020) Passenger-centered vulnerability assessment of railway networks, *Transportation Research Part B: Methodological*, **136**, 30–61, ISSN 0191-2615.
- Tréfond, S., A. Billionnet, S. Elloumi, H. Djellab and O. Guyon (2017) Optimization and simulation for robust railway rolling-stock planning, *Journal of Rail Transport Planning and Management*, **7** (1-2) 33–49, ISSN 22109706.
- Van Aken, S., N. Bešinović and R. M. P. Goverde (2017) Designing alternative railway timetables

- under infrastructure maintenance possessions, *Transportation Research Part B: Methodological*, **98**, 224–238, ISSN 0191-2615.
- Veelenturf, L. P., M. P. Kidd, V. Cacchiani, L. G. Kroon and P. Toth (2016) A railway timetable rescheduling approach for handling large-scale disruptions, *Transportation Science*, **50** (3) 841–862.
- Veelenturf, L. P., L. G. Kroon and G. Maróti (2017) Passenger oriented railway disruption management by adapting timetables and rolling stock schedules, *Transportation Research Part C: Emerging Technologies*, **80**, 133–147, ISSN 0968090X.
- Wang, P., A. Trivella, R. M. P. Goverde and F. Corman (2020) Train trajectory optimization for improved on-time arrival under parametric uncertainty, *Transportation Research Part C: Emerging Technologies*, **119** (January) 102680, ISSN 0968090X.
- Wüst, R., S. Bütikofer, S. Ess, C. Gomez, A. Steiner, M. Laumanns and J. Szabo (2019a) Maintenance timetable planning based on mesoscopic infrastructure and the transport service intention, *Journal of Rail Transport Planning & Management*, **11**, 100146.
- Wüst, R., S. Bütikofer, S. Ess, C. Gomez, A. Steiner, M. Laumanns and J. Szabo (2019b) Periodic timetabling with ‘track choice’-pesp based on given line concepts and mesoscopic infrastructure, in *Operations Research Proceedings 2018*, 571–578, Springer.
- Yan, F. and R. M. P. Goverde (2019) Combined line planning and train timetabling for strongly heterogeneous railway lines with direct connections, *Transportation Research Part B: Methodological*, **127**, 20–46, ISSN 01912615.
- Zhang, X. and L. Nie (2016) Integrating capacity analysis with high-speed railway timetabling: A minimum cycle time calculation model with flexible overtaking constraints and intelligent enumeration, *Transportation Research Part C: Emerging Technologies*, **68**, 509–531, ISSN 0968090X.
- Zhang, Y., Q. Peng, Y. Yao, X. Zhang and X. Zhou (2019) Solving cyclic train timetabling problem through model reformulation: Extended time-space network construct and Alternating Direction Method of Multipliers methods, *Transportation Research Part B: Methodological*, **128**, 344–379, ISSN 01912615.
- Zhou, Y., J. Wang and H. Yang (2019) Resilience of Transportation Systems: Concepts and Comprehensive Review, *IEEE Transactions on Intelligent Transportation Systems*, **20** (12) 4262–4276, ISSN 15580016.
- Zhu, Y. and R. M. P. Goverde (2017) System-based vulnerability measures for railway systems, paper presented at the *7th International Conference on Railway Operations Modelling and Analysis: RailLille2017*.

Zhu, Y. and R. M. P. Goverde (2019) *Dynamic Passenger Assignment for Major Railway Disruptions Considering Information Interventions*, vol. 19, Networks and Spatial Economics, ISBN 1106701909.

Zhu, Y. and R. M. P. Goverde (2020a) Dynamic and robust timetable rescheduling for uncertain railway disruptions, *Journal of Rail Transport Planning & Management*, **15** (March) 100196, ISSN 22109706.

Zhu, Y. and R. M. P. Goverde (2020b) Integrated timetable rescheduling and passenger reassignment during railway disruptions, *Transportation Research Part B: Methodological*, **140**, 282–314, ISSN 01912615.

Zhu, Y. and R. M. P. Goverde (2021) Dynamic railway timetable rescheduling for multiple connected disruptions, *Transportation Research Part C: Emerging Technologies*, **125** (January 2019) 103080, ISSN 0968090X.

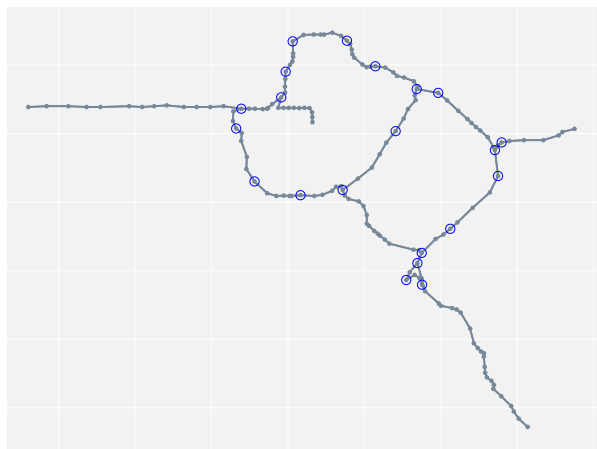
A Supporting Material for the Case Study

Here we list all supporting material.

A.1 Subset of Stations for Specific Purposes

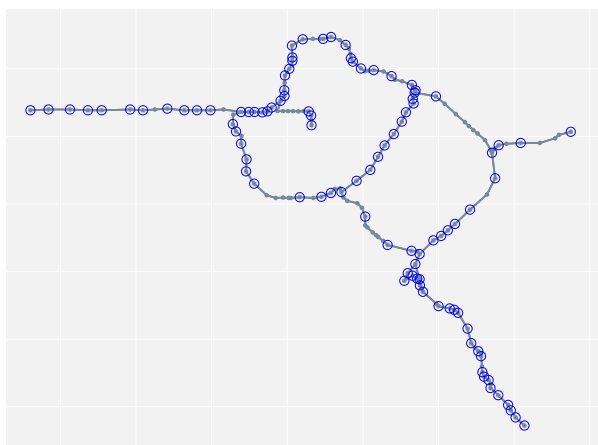
Fig. 30 designates all stations, where transferring between trains is allowed in the **OM**.

Figure 30: Stations suitable for transferring in the **OM** are marked blue.



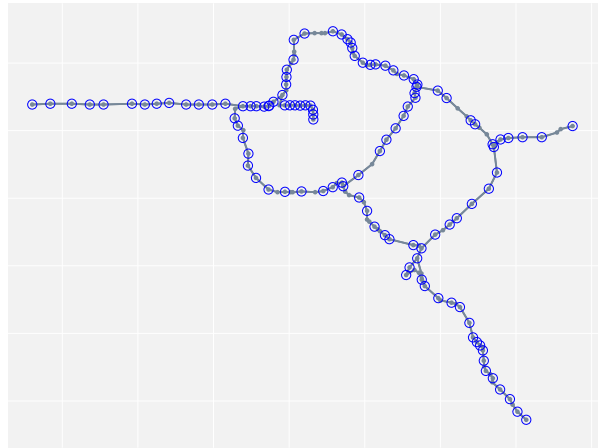
Appendix A.1 designates all stations, where transferring between trains is allowed in the **AM**.

Figure 31: Stations suitable for transferring in the **AM** are marked blue.



Appendix A.1 designates all stations, where two or more station tracks are available, such that short turning is allowed in the **AM**.

Figure 32: Stations suitable for short turning trains in the OM are marked blue.



A.2 Considered SAT Solvers

To solve the [Boolean Satisfiability Problem \(SAT\)](#) instances, different solvers are available. Since the [M-SSSP](#) requires feedback in form of unsatisfiable cores from the [SAT](#)-solvers for conflicts in the [S-SSSP](#), we only use solvers that yield unsat-cores for unsatisfiable [SAT](#)-instances. All eleven solvers included in the `python-sat` package provided by [Ignatiev et al. \(2018\)](#) allow extraction of unsatisfiable cores. Consequently, we compared all the solvers in the package. We found, that [Glucose 4.1](#) provided by [Audemard and Simon \(2018\)](#) and [MiniCard 1.2](#) by [Liffiton and Maglalang \(2012\)](#) perform best, with the latter slightly outperforming the former.

A.3 Interactive Figures as *html*-files

To provide the readers of this thesis with additional material, several figures that visualise the results of the computational experiments are available as interactive *html* files. Consequently, any figure whose description is linked to this section is provided as *html*-file. These can be opened with any browser and allow a more detailed inspection of the results. Thus, we provide *html*-files instead of tables. The files can be found in the `InteractiveHtmlFigures` folder, submitted together with the thesis. The name of the file corresponds to the figure id.