KU LEUVEN
FACULTEIT
INGENIEURSWETENSCHAPPEN

Optimal timetables for temporarily unavailable tracks

Sander Van Aken

Thesis voorgedragen tot het behalen
van de graad van Master of Science
in de ingenieurswetenschappen:
verkeer, logistiek en intelligente
transportsystemen

Promotor:
Prof. dr. ir. Pieter Vansteenwegen

Assessoren:
Prof. dr. ir. Herman Bruyninckx
Prof. dr. ir. Greet Vanden Berghe

Begeleider:
Sofie Van Thielen

Academiejaar 2016 – 2017

Preface

Although it lasted longer than initially planned, my master's degree has been an amazing, challenging and rewarding time, during which I got opportunities I never expected to get upon starting it. One of them, is this thesis research. This document marks the end of an intensive semester, during which I learnt a lot of things, not least about myself. With the gathered skills and knowledge, I feel ready to tackle the unknown future challenges. I would like to grasp this opportunity to express my sincere gratitude to a number of people.

Primarily to my supervisors Prof. dr. ir. Pieter Vansteenwegen and Sofie Van Thielen, who agreed to conduct this thesis within one semester, which required a significant effort also from their side. Their constructive feedback during the numerous meetings, comments on the initial versions of this text, and the positive collaboration, allowed me to improve the work, and my own attitude along several aspects. It was tough, but rewarding to work together with you!

Next, I would like to thank Marc Lambrechts of Infrabel to provide me with answers on all of my questions on current dispatching practice during a visit to the local signal box of Leuven. My thanks also go to Prof. dr. ir. Dirk Cattrysse, who gave me more insights on the computational complexity of the models developed here, but also supported me in all my choices throughout the whole master's programme.

Of course I would like to thank my family and friends, and especially my parents and sister, who have been standing by me through the whole of my study career. To admit, it has been a bumpy ride with ups and downs, and sudden turns, but you kept your belief in me, and allowed me to develop myself.

Finally, I would like to express my hope that this thesis research may contribute to the Belgian railway system, and above all, may improve the effects on its passengers.

Sander Van Aken

Contents

Abstract

Train passengers expect a high level of service under all circumstances, while disruptions occur on a daily basis on the Belgian railway network, and dispatchers at Infrabel still have to rely on their experience to tackle them. This thesis develops mathematical models that can support dispatchers during disruptions resulting in capacity reduction, on three types of frequently encountered parts of the infrastructure (AIPs), outside station areas.

A first AIP consists of a double-track corridor with a double switch on both sides, modelled as a single-machine scheduling problem with rejection, which accounts for train re-timing, reordering, rerouting and cancellation. Next, the second AIP incorporates stops along the corridor. The model is extended with discretely controllable process times to account for stop-skipping as an additional measure. Finally, disruptions on multi-track corridors are modelled as a parallel-machine scheduling problem, thereby accounting for conflicts on the aligning junctions.

Applying these models on different practical case studies allows to formulate service constraints, which dispatchers could consider. Additionally, all models are subjected to a theoretical analysis of their parameters' impact. The results obtained by solving the mathematical models are compared to two heuristics, representative for dispatcher's decision making.

The models outperform both heuristics by effectively balancing between all possible measures. Original timetable structure is identified as the major determinant for the generated disruption timetables. Results also suggest that interactions between measures are much more complex than one could intuitively expect, even in absence of additional service constraints. This clearly illustrates that these models could improve current dispatching practice.

Future research could evaluate the impact of the generated disruption timetables on the complete network, as conflicts might arise outside of the considered AIPs. To deal with the uncertain nature of disruptions and additional delays, the models could be embedded in a rolling horizon approach, rerunning them upon information updates. The AIP concept presents a powerful approach to model a wide range of disruptions, and future work could aim at identifying new ones, thereby developing models for them.

Dispatchers could employ the models by characterizing the disruption and the infrastructure around it with a limited set of parameters. This approach allows to combine the models' merit of efficiently balancing between measures, with the dispatcher's knowledge on the surroundings of the corridor, and passenger expectations.

Samenvatting

Treinreizigers verwachten in alle omstandigheden een goede dienstverlening, ook tijdens de dagelijkse storingen op het Belgische treinnetwerk. Dispatchers bij Infrabel vertrouwen echter volledig op hun ervaring tijdens het herplannen. Deze thesis ontwikkelt wiskundige modellen om hen te ondersteunen tijdens storingen die leiden tot capaciteitsreductie, op drie veelvoorkomende stukken infrastructuur (AIP's), buiten stationsgebieden.

Het eerste AIP bestaat uit een dubbelsporige lijn met een dubbele wissel aan beide zijden, gemodelleerd als een één-machine probleem met volgorde-afhankelijke omsteltijden en annulering, waarbij treinen vertraagd, herordend, omgeleid en afgeschaft kunnen worden. Vervolgens neemt een tweede AIP haltes langs de lijn in rekening. Het model wordt uitgebreid met controlleerbare procestijden om het overslaan van haltes te modelleren. Ten slotte laat een parallel-machine probleem toe om storingen op meersporige segmenten te modelleren, rekening houdend met de toegenomen complexiteit van de wisselzones.

Aan de hand van praktische casussen worden ook richtlijnen ontwikkeld die dispatchers zouden kunnen toepassen om de alternatieve dienstregeling te verbeteren voor de passagiers. Daarnaast wordt de impact van de modelparameters onderzocht. De resultaten van het model worden vergeleken met twee representatieve heuristieken.

Door alle mogelijke maatregelen op een effectieve manier ten opzichte van elkaar af te wegen, genereert het model betere resultaten dan de heuristieken. Uit alle parameters blijkt de structuur van de oorspronkelijke dienstregeling de sterkste invloed te hebben op de alternatieve dienstregeling. Daarnaast zijn interacties tussen verschillende maatregelen zeer complex, en soms contra-intuïtief. Dit toont aan dat deze modellen de huidige methodes aanzienlijk kunnen versterken.

Verder onderzoek kan zich toeleggen op het evalueren van de impact van de alternatieve dienstregelingen op de rest van het netwerk, aangezien zich buiten de AIP's mogelijke conflicten kunnen voordoen. Om tegemoet te komen aan de onzekerheid tijdens storingen en eventuele bijkomende vertragingen, zouden de modellen in een rollende horizon-aanpak kunnen worden gebruikt. Ten slotte lijkt het AIP-concept een krachtige manier om storingen op een groot deel van het netwerk aan te pakken, en zou verder onderzoek nieuwe AIP's kunnen identificeren en modelleren.

Dispatchers kunnen de modellen gebruiken door een beperkt aantal parameters omtrent de storing en de omliggende infrastructuur in te geven. Deze aanpak combineert de kracht van de modellen om een goede balans tussen maatregelen te vinden, met de kennis van dispatchers over passagiersverwachtingen.

List of Figures

List of Tables

List of Abbreviations and Symbols

Abbreviations

AIP	Archetypical infrastructure piece
DRP	Disruption Recovery Program
DSS	Decision support system
IC	Intercity train
L	Local train
MILP	Mixed integer linear programming
MPC	Model predictive control
ROC	Railway Operations Centre
RS	Rolling stock
SSP	Schematic Signalling Plan
TMS	Traffic Management System
TTR	Train Traffic Rescheduling problem

Symbols

β	Set of block sections along the corridor.
β_D	Set of block sections within the disrupted area.
δ^{swap}	Threshold for relaxing orders between trains in the same direction.
Δr_{tbk}^{stop}	Increase in running time of train t on track k over block section b associated with performing the stop, taking into account braking, dwelling and re-acceleration.

Δr_{tb}^{stop} Increase in running time of train t over block section b associated with performing the stop, taking into account braking, dwelling and re-acceleration.

Δs_{ijk} Difference in set-up time associated with one or both trains i and j on track k performing its/their scheduled stops.

Δs_{ij} Difference in set-up time associated with one or both trains i and j performing its/their scheduled stops.

$\omega_{b,stop}^{dir}$ Multiplier to determine the stop-skipping penalty for trains in direction $dir \in \{0, 1\}$ with a scheduled stop at o_s in block section b.

π_k Parameter indicating whether a platform is located along track k ($\pi_k = 1$) or not ($\pi_k = 0$).

σ_t Binary variable indicating whether train t performs all of its stops within the disrupted area ($\sigma_t = 1$) or none of them ($\sigma_t = 0$).

$\tau_{i,b}^e$ End of the blocking time of train i on block section b.

$\tau_{i,b}^s$ Start of the blocking time of train i on block section b.

A, B Parts of the corridor located between the switches and its border points, e.g. stations or junctions, representing the first occurrence of trains running in direction 1 and 0 respectively.

a_t Acceleration rate of train t.

b_t Braking rate of train t.

B_{diff} Buffer time added in the disruption timetable between trains running in opposite directions on the same track.

B_{same} Buffer time added in the disruption timetable between trains running in the same direction on the same track.

C_t Completion time of train t, i.e. departure time from the disrupted area.

c_{rob} Percentage increase in objective function value when inserting buffer times into the disruption timetable.

D Part of the corridor on which the disruption happens, may consist of one or more block sections.

D_t Delay of train t upon leaving the disrupted area.

d_t Due date of train t.

D_t^{max} Maximum acceptable delay for train t.

dir_t Direction of train t, either 0 when running from right to left in all plots, or 1 when running from left to right.

h^{after} Increased headway needed between trains i and j if train i does not run on its originally assigned track.

h^{before} Increased headway needed between trains i and j if train j does not run on its originally assigned track.

k_t Track k assigned to train t in the disruption timetable for the third AIP.

k_t^{orig} Originally assigned track k of train t for the third AIP.

M Big-M: a large value used to relax constraints in case they are not needed.

M^D Machine representing the disrupted area.

M_T Set of tracks on an N-track corridor.

m_{tk} Binary track assignment variable, indicating whether train t runs on track k ($m_{tk} = 1$) or not ($m_{tk} = 0$).

n_D Number of block sections in segment D.

n_t Ordinal number of train t in its direction, i.e. if it is the fifth train running in direction 1, $n_t = 5$.

N_{batch}^{max} Maximum number of trains allowed in the same batch.

$N_{dev,dir}^{max}$ Maximum number of trains per hour in direction $dir \in \{0, 1\}$, that can be rerouted over alternative corridors.

$N_{dir,LOS}^{min}$ Minimum number of trains required to run in direction $dir \in \{0, 1\}$ during one hour.

$N_{stop,b}^{min,dir}$ Minimum number of stops per hour for the stop located in block section b to serve passengers travelling in direction dir.

O Set of open-track stops present along the disrupted area.

OB Maximum allowed off-balance in number of cancelled trains per hour between both directions.

p_t Process time of train t, i.e. the total running time over the disrupted area.

p_{tk} Process time of train t on track k.

q_{ijk} Binary order variable of trains i and j on track k. Following the Manne concept [33], $q_{ijk} = 1$ if train i is scheduled before train j on track k and not necessarily right before.

r_t Release time of train t.

R_{tbk} Running time of train t over block section b of track k without performing a stop.

$R_{tbk}^{stopping}$ Running time of train t over block section b of track k if it performs a scheduled stop with it.

R_{tb} Running time of train t over block section b without performing a stop.

$R_{tb}^{stopping}$ Running time of train t over block section b if it performs a stop.

S Set of originally scheduled stopping operations (t, b) by train t at stop o_s within block section b

S_1, S_2 Switches aligning the disrupted area (segment D), mostly consisting of one block section, unless a stop is present on the switch.

$s_{ijk}^{disrupted}$ Set-up time between trains i and j, both running on track k and without performing any stopping operation.

$s_{ijk}^{stopping}$ Set-up times associated with one or both trains i and j performing its/their scheduled stops when both are running on track k.

s_{ij} Set-up time between trains i and j for the disrupted area.

$s_{ij}^{disrupted}$ Set-up time between trains i and j without performing any stopping operation.

$s_{ij}^{stopping}$ Set-up time between trains i and j associated with one or both performing its/their scheduled stops.

T Set of trains t.

$t^{clear\ signal}$ Technical time required to change the aspect of a signal, used to calculate blocking times.

$t^{clearing}$ Clearing time, i.e. time elapsed between the head and the rear-end of the train passing a signal.

$t^{release}$ Technical release time of the signals, used to calculate blocking times.

t^{sight} Time between the moment a driver observes an aspect and when the train enters the next block section.

T_{dir} Subset of trains running in the same direction $dir \in \{0, 1\}$

T_{dir}^{h} Set of trains running during the h^{th} hour of the scheduling horizon, with $h \in \mathscr{H}$, the set of all 1 h-intervals.

$t_{dwell,s}^{min}$ (Minimum) dwell time of each train at stop o_s.

t_{end} Time denoting the end of the scheduling horizon.

$t_{j,b}^{approach}$ Time a train t requires to reserve a block section b before it physically enters it.

$t_{stop}^{approach}$ Approach time to be included in blocking time calculations after a train has performed a stop at the previous block section.

v_D Average speed over segment D during the disruption, typically lower than the reference speed.

v_r Maximum allowed speed on track r during a disruption on an N-track corridor.

v_t^{max} Maximum allowed speed of train t, regardless of the infrastructure.

v_{max} Speed restriction imposed by the infrastructure, may differ for each block section.

v_{sight} Reduced maximum allowed speed if the train driver has to stop-on-sight, e.g. when performing a stopping operation within the block section.

v_{tb} Maximum allowed speed of train t over block section b.

w^{change} Penalty for assigning trains to tracks differing from a "desired" one.

w_t^{cancel} Penalty for cancelling train t.

w_t^{delay} Penalty for delaying train t by 1 s.

$w_t^{deviation}$ Penalty for rerouting train t outside of the disrupted area.

$w_{t,b}^{stop-skip}$ Penalty for train t skipping its stop at o_s, which is located in block section b

X_i^j Term including the cancellation and deviation of all trains scheduled between trains i and j in the same direction.

z_{tk} Parameter indicating whether train t can reach track k using the switches aligning an N-track corridor ($z_{tk} = 1$), or not ($z_{tk} = 0$).

Chapter 1

Introduction

Railway transport is considered to be one of the main alternatives in battling the increasing level of car traffic congestion on the Belgian road network. Its operation is preceded by a succession of different planning steps, including a precise and extensive timetable design. Despite these efforts, delays are unavoidable and may render the timetable infeasible to operate as planned. However, reliability is one of the main passenger satisfaction determinants [20]: passengers especially remember those large delays following disruptions, shaping the general opinion on service quality. Dispatchers at Infrabel should aim at providing the highest possible level-of-service during them. However, at Infrabel, no automated decision support suited for this task, is at hand. During unexpected, large disruptions and their large-scale consequences, they mostly have to rely on their own knowledge and expertise. Therefore, this thesis aims at suggesting the best possible timetable adjustments in case of large disruptions, in order to support dispatchers' decisions.

Research has brought forward a range of models and solution approaches to deal with both small perturbations and larger disruptions. Each of them has its own merits and limitations, focusing on timetable adjustments and/or rolling stock and crew rescheduling. They present promising results, some of them within acceptable computation times for real-life applications. However, implementation is still lacking and current practice often resorts to predefined plans. As not each possible situation can be covered by these plans, they (almost) always have to be adapted to prevailing circumstances.

The Belgian infrastructure manager Infrabel and train operator NMBS are subject to reaching targets for punctuality within their management contract [39]. Infrabel provides punctuality statistics on a monthly basis, thereby distinguishing delays and train cancellations [24]. In 2015, punctuality reached 89.3%, and 22,947 trains (1.9%) had to be cancelled. After these strong improvements compared to 2014 [39], in which 86.2% of trains were on time and 2.4% had to be cancelled, punctuality figures have decreased again in 2016. Nevertheless, these numbers are considered to be unacceptable, and disruptions still occur frequently. Although no accurate information on their frequency and type is available, estimates are that on average one disruption is encountered each day [30]. To identify what or who causes them, cancellations can be considered to be an indicator. Figure 1.1 displays such statistics together with a number of possible causes per actor. Interestingly, third parties were the main contributor, a difficult factor to control.

The 2014 yearly report of NMBS [39] states that “punctuality is the most essential performance indicator”, and identifies several major working domains, among which close monitoring of traffic and a better flow of information within and between actors. Although not perfect, research shows that predefined plans can contribute to the latter goal. No such plans exist at Infrabel, reflecting the idea that every disruption is unique [30]. Without much doubt, the occurrence of disruptions can be considered unique due to the wide range of causes (Figure 1.1), locations, timings and uncertain nature.

However, it can be questioned whether the effects of a disruption on traffic are always unique and no structure exists within possible solutions. Consider for example the breakdown of a train on a double-track corridor between two double switches. Figure 1.2a illustrates how trains in both directions can still use the remaining track: the disruption merely results in a capacity reduction. Examining possible locations of this disruption on part of the Belgian network (Figure 1.2b), large similarities in terms of infrastructure seem to exist. Here, the distance between both switches could be the main determinant for the solution. Within this thesis, parts of the infrastructure which are omnipresent on a railway network, such as the one shown in Figure 1.2a, are called *archetypical infrastructure pieces* (AIP).

This thesis identifies a number of such AIPs outside of station areas and develops models to generate disruption timetables based on the arising situation. Hence, it bypasses the disadvantages of predefined plans not being applicable or available in all circumstances. After entering a limited number of parameters about the AIP, dispatchers should retrieve a suggestion on how to tackle the problem in the best possible way.

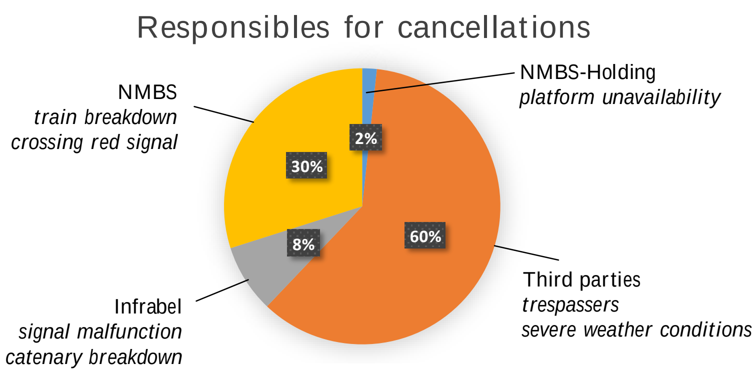
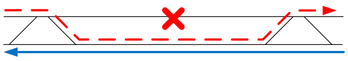
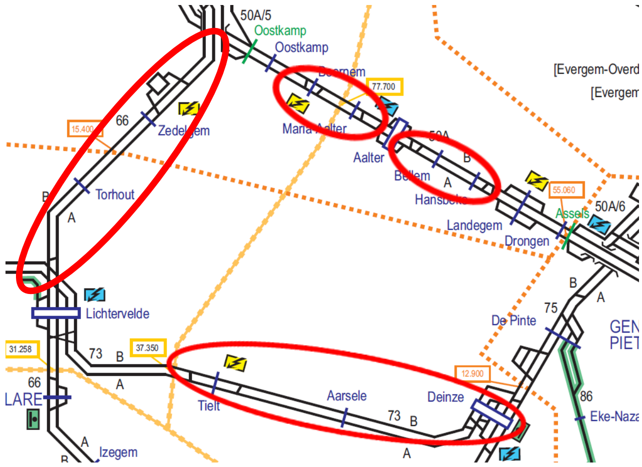


Figure 1.1: Attribution of train cancellations towards several responsible actors for the 22,947 cancelled trains in 2015. For each actor, possible (disruption) causes are mentioned. Data obtained from [24].



(a)



(b)

Figure 1.2: (a) Sketch of the impact of a train breakdown on a double-track corridor between two double switches as example to illustrate the concept of AIPs. (b) Several occurrences of this AIP on a part of the Belgian network. Adapted from [26].

Such an approach has not been developed in scientific research yet. Scope is limited to timetable rescheduling during disruptions with capacity reduction: trains can still be operated over the remaining track(s). Following this idea, this thesis' central research question is formulated as follows:

> How can models with few parameters support dispatchers' decision-making by providing solutions during disruptions with partial blockage on archetypical infrastructure pieces of the Belgian railway network?

First, Chapter 2 introduces the Belgian railway system and basic theoretical concepts in railway signalling and blocking time theory. Next, Chapter 3 limits the scope and specifies the approach taken within this thesis research by formulating subquestions. Chapter 4 continues with a literature study on railway disruption management and models to deal with them. It situates this thesis within current state-of-the-art research and identifies possible approaches to deal with the studied problem.

Next, Chapter 5 develops the mathematical model and its framework using a practical case study of the first AIP: a double-track corridor aligned by two double switches. Several parameters influence the model's results. Hence, an extensive theoretical assessment of their impact is conducted by means of an artificial case study (Chapter 6). Chapters 7 and 8 elaborate on the models for two additional AIPs of increased complexity by respectively incorporating stops along the corridor, and considering AIPs consisting of more than two tracks.

Finally, Chapter 9 reflects on the obtained results and the work conducted during this thesis research. To conclude, Chapter 10 summarizes the answers to the research questions and identifies possibilities for future research.

Chapter 2

Basic Concepts

The aim of this chapter is twofold. First of all, Section 2.1 sketches the context of the Belgian railway system, including its actors and their interactions, together with passenger expectations. Secondly, a thorough understanding of all concepts developed and presented within this thesis, requires some specific knowledge. Section 2.2 briefly discusses railway signalling, blocking time theory and timetabling. These aspects will be used frequently throughout the remainder of this thesis.

2.1 Introduction to the Belgian railway system

The Belgian railway system holds two major players: infrastructure manager Infrabel and (domestic) passenger train operator NMBS. For freight traffic, an open-access market exists, in which multiple operators are present, e.g. incumbent B-logistics. [58]

Infrabel owns, plans, designs, builds and maintains the track infrastructure. Possibly, subcontractors are hired to conduct (sub)tasks. Concerning daily operations, it is responsible for timetabling and more specifically path allocation. Timetables are constructed in coordination with both NMBS and the freight operators [58]. On the other hand, NMBS acquires, owns, schedules and maintains the rolling stock (RS) for passenger operations. It is also responsible for crew planning and management. NMBS collects fares from the passengers and pays infrastructure charges to Infrabel for using the tracks [58]. According to the yearly report of NMBS [39], train traffic accounted for 7.38% of all passenger kilometres travelled in 2012.

The Belgian situation in terms of both infrastructure and passenger demand is characterized by a very specific pattern with a strong focus towards Brussels. A large number of train lines run through or end near the Brussels' North-South connection, being the network's main bottleneck. This follows from the passenger demand pattern: during morning peak, passengers travel towards Brussels to go to work and in the evening, the opposite movement occurs. As a result, demand during weekdays strongly concentrates around morning and evening peaks as illustrated by Figure 2.1.

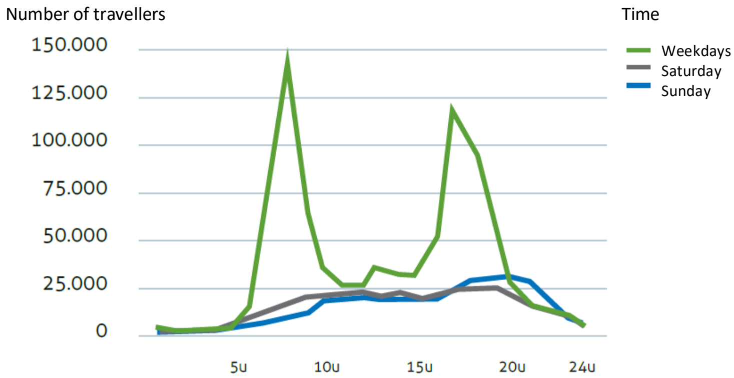


Figure 2.1: Number of passengers transported by NMBS along the day, specified for weekdays, Saturday and Sunday. On weekdays, demand is especially concentrated during morning and evening peak. Translated from [39].

Timetable development aims to satisfy both the management contract with the government and passenger demand. NMBS [28] operates four main types of trains. Intercity (IC) and local (L) trains run all day long and focus on long- and short-distance travellers respectively. Additionally, S trains are operated in the surroundings of Brussels to satisfy suburban demand. On top of these (rather) regular schedules, peak (P) trains are planned to serve the specific demand pattern. These may either provide additional capacity on IC connections or present new services. Neither absolute passenger numbers are known, nor good estimates exist [52]. Next to these, also international trains such as Thalys and ICE are operated.

In case of delays or disruptions, the separation of responsibilities between Infrabel and NMBS still holds. Infrabel has the final decision in path allocation, NMBS reschedules the rolling stock and crew. As it gains its revenues from passengers, NMBS also has to take care about the commercial aspect: serve the passenger as good as possible. Despite their specific responsibilities, a strong interaction between both actors remains. Section 4.3.5 presents a more in-depth discussion.

It is within this operational context and specific demand pattern that this thesis is situated. Both aspects have to be taken into account when considering the case studies and discussing their results.

2.2 Blocking time theory and timetabling levels

This section presents a number of definitions and theoretical concepts used throughout the thesis. First, Section 2.2.1 discusses the need for signalling and its effects on train operation. Next, Section 2.2.2 explains the concept of blocking time theory, which is used in standardized methods for railway line capacity assessment by the International Union of Railways (UIC [55]). *Lines* connect stations and *junctions*, i.e. “points at which at least two lines converge” [55]. Finally, Section 2.2.3 distinguishes the macroscopic from the microscopic level.

2.2.1 Signalling systems

Because of their long braking distances, trains running at a high speed cannot stop in time when the driver can only rely on sight. Safety and signalling systems have been designed to ensure safe train separation and prevent train collision. Nonetheless, they also aim at efficient infrastructure exploitation, i.e. without requiring extensive time separation between trains [48]. Fixed block signalling systems are omnipresent on current railway systems. They rely on signals located at fixed locations along the track to transmit information on downstream track occupation towards the train driver, who is expected to react accordingly. These signals split a track into several so-called *block sections*, "a section of a track [...] which a train may only enter when it is not occupied by another train or vehicles [sic]" [21].

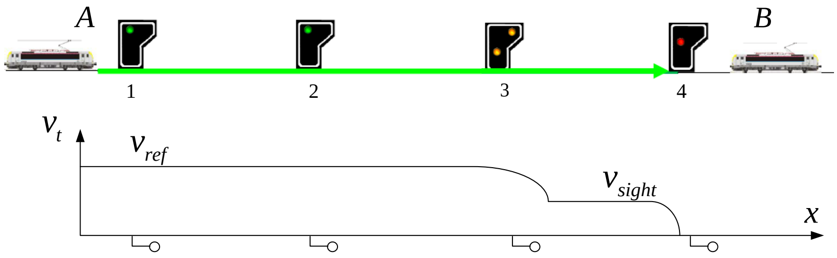
Except for single-track lines, a track mostly accommodates trains running in a specific direction. On a Belgian double-track line, trains are expected to run on the left track (*normal track regime*). Nonetheless, all tracks of the Belgian railway network used on a daily basis, have signals in both directions. This allows to operate trains running in so-called *counter track regime* also in an efficient manner. [18]

The Belgian signalling system is categorized as a *3-aspect-2-block signalling system*, meaning that every signal can show three aspects providing information about the occupation of up to two block sections ahead. A green aspect indicates that the next two block sections are free and reserved for this train. Hence, it is allowed to run at at most the *reference speed* v_{ref}, being "the maximum permissible speed" [18] imposed by the infrastructure. When the second block section following the signal is occupied or reserved for another train, the signal shows a *double yellow* aspect and the train has to slow down upon entering the block section. Commonly, this speed equals 40 km/h allowing the driver to stop within sight distance of the next signal[1]. Finally, a red aspect indicates that the next block section is occupied and the train has to stop in front of it. Figure 2.2a shows the resulting allowed speed profile for train A following train B with two block sections in-between. The moment train B leaves its current block section, signals 3 and 4 turn green and double yellow respectively and the speed profile is updated. Efficient train operation prevents braking and acceleration as much as possible. Hence, it is desirable that a train driver only meets green signals in-between two stops, commonly referred to as a *green wave (GW) policy*. [18, 43]

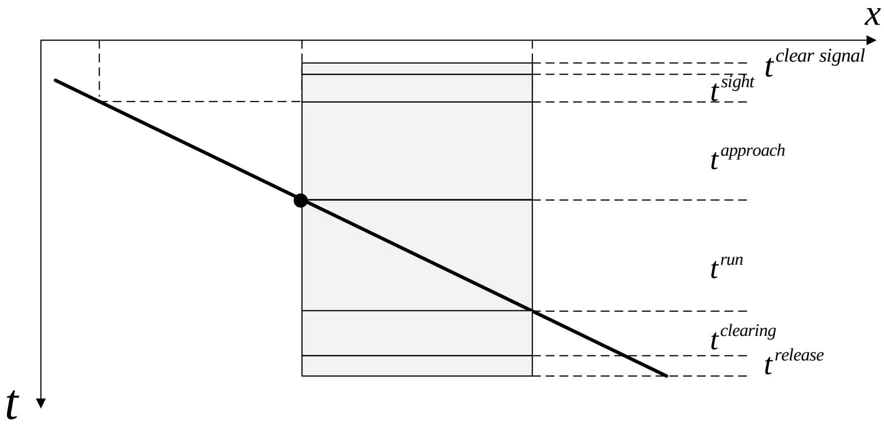
2.2.2 Blocking time theory

Figure 2.2b shows a time-distance diagram with the horizontal and vertical axes representing respectively distance along the line, and time. These diagrams are used to visualize the journey of one or multiple trains over a railway line. The bold line represents the *train path* of train A resulting from the speed profile for the first two block sections, which "describes the usage of infrastructure for a train movement on a track and in time" according to Hansen and Pachl [21].

[1] Also other speed limitations can be communicated towards the drivers, possibly in combination with green aspects. The reader is referred to [18] for a comprehensive overview.



(a)



(b)

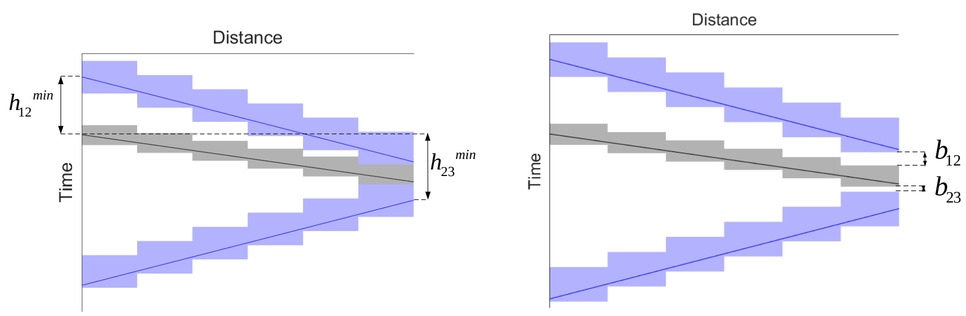
Figure 2.2: (a) Signalling logic on an open-track line for train A running from left to right behind train B (adapted from [18]) and the associated (maximum) speed profile. (b) Time-distance diagram indicating the blocking time components of train B running over the block section between signals 2 and 3. Based on [43] and [55].

Additionally, time-distance diagrams may show how a train reserves and uses the infrastructure. The *blocking time* is the "time interval during which a section of track is allocated for exclusive use of one train and is therefore blocked to all other trains" [21], represented by the shaded area in Figure 2.2b for the block section between signals 2 and 3. The black dot indicates the passing of the head of train A at signal 2. For a driver to meet only green signals, a block section has to be reserved a certain time before and after it physically enters and leaves it, resulting in six major constituents of the blocking time. A green aspect requires a block section to be reserved before the train enters the previous one. The *approach time* ($t^{approach}$) therefore equals the running time over the preceding block section. The driver observes signal 2 already some time before passing it and the signal should show a green aspect before the train is within sight distance of it, resulting in the *sight and reaction time* (t^{sight}). Additionally, allocating the route and switching the aspect also requires some time ($t^{clear\ signal}$). After a certain running time (t^{run}) the head of the train leaves the block section again, but its rear end still has to pass signal 3. The clearing time ($t^{clearing}$) required to do so depends on the train's speed profile and length. Finally, releasing the block section takes an additional technical time ($t^{release}$).

Figure 2.3a shows the train paths of three trains over a stretch of six block sections: a slow and a fast one running in the same direction, followed by a slow one in the opposite direction. For a specific train, the combination of blocking times for each block section along its train path constitutes its *blocking time stairways*, indicated by the shaded areas. These blocking time stairways represent how a train reserves and uses each block section when it only meets green signals, i.e. within a GW policy. In case blocking times (partially) overlap, the occupation by the first train results in the second train meeting a yellow or red aspect somewhere along the line and it has to either slow down or stop. These occurrences are referred to as *conflicts*. Hence, when developing a timetable, one should aim at a conflict-free schedule and a minimum time and spatial train separation has to be respected. *Minimum headway times* represent the time required between the arrival of two trains on a line section, for example h_{12}^{min} between the trains running from left to right[2]. With this definition, headway times are much larger for trains in opposite directions than for those running in the same direction. The *critical block section* refers to the location with minimum time separation, i.e. the extreme right block section for both pairs in Figure 2.3. [43]

However, real-life operations are prone to schedule deviations. In case the blocking time stairways for the first train in Figure 2.3a is altered, e.g. a longer running time in one of the block sections, this immediately causes a conflict with its successor

[2] Pachl [43] presents several alternatives to define minimum headway times, e.g. between arrivals and departures at stations aligning the section. Essentially, they boil down to the same foundation: avoiding overlap of blocking time stairways.



(a) (b)

Figure 2.3: Time-distance diagrams representing a timetable with two trains running from left to right, followed by one in the opposite direction. (a) Time-distance diagram in which blocking time stairways touch each other, used to determine minimum headway times h_{ij}^{min}. (b) Alternatively, some buffer time b_{ij} may be present between subsequent trains.

as blocking time stairways become overlapping. Hence, train 2 will experience a delay caused by the delay of train 1, referred to as *knock-on delay*. Train 2 has to brake and re-accelerate due to this conflict, resulting in a longer running time and transferring an even bigger delay to train 3, which results in a snowball effect. To prevent small delays from influencing subsequent train paths, *buffer times* are added to the timetable. The latter can be seen as the smallest time separation between two trains in a time-distance diagram, e.g. b_{12} and b_{23} in Figure 2.3b. [43, 47, 55]

Here, time-distance diagrams are frequently used both to illustrate certain concepts and to visualize the obtained results. Appendix A provides an overview of how those shown within the thesis should be interpreted.

2.2.3 Macroscopic versus microscopic models

Timetables can be defined at either the macroscopic or microscopic level. At the *macroscopic level*, all tracks within a station are represented by a single node. Stations, stops and junctions are connected by arcs representing the railway line without considering the number of tracks. Hence, macroscopic timetabling models define arrival, passing and departure times at nodes without explicitly taking into account the exact train path and signalling system constraints. Radtke [47] states they "are not suitable for all detailed railway operation planning tasks".

For example, to calculate running times and investigate conflict-freeness of a timetable, the *microscopic* level has to be considered. At this level, the timetable incorporates the exact routes used by the trains, i.e. tracks and block sections, and details such as the passing times at each signal. Such models take into account detailed train (e.g. acceleration), track (e.g. gradient) and signalling characteristics. Blocking time theory then allows to determine the minimum headway times. As stated before, a timetable is conflict-free when none of the time-distance diagrams shows overlaps. [47]

Chapter 3

Scope and Methodology

This chapter limits the scope of this thesis (Section 3.1) and splits the central research question into three subquestions (Section 3.2). Each of them is translated to a number of goals. Section 3.3 presents the methodologies to achieve these goals, and provides a structure for the thesis research.

3.1 Scope

Chapter 1 formulated the central research question as:

> How can models with few parameters support dispatchers' decision-making by providing solutions during disruptions with partial blockage on archetypical infrastructure pieces of the Belgian railway network?

Answering this research question results in two main goals. First, models are built to generate disruption timetables which result in a suitable solution that could be further specified by the dispatcher if required. Hence, focus is on providing a timetable for the remaining disruption duration. Next, these models are exploited to solve multiple scenarios for a range of parameters expected to influence the final solution. Potentially, general rules to deal with a disruption of a certain type, can be derived from these experiments.

One can imagine a complete railway network with a large number of AIPs, which are not all as recurrent as others. For example, station areas may be characterized by very specific infrastructure configurations depending on factors such as the number of platforms or incoming corridors. The scope of this thesis is limited to three AIPs of increasing complexity, frequently encountered on *corridors* of the network. The latter are defined as (parts of) railway lines between stations with more than two platforms and junctions, where trains can switch between two railway lines. Furthermore, only timetable rescheduling is considered as rolling stock and crew rescheduling is mostly conducted on a network basis. Obviously, timetable changes could lead to infeasible rolling stock or crew schedules. However, while minimizing the delays of the involved trains, one aims to automatically limit these infeasibilities as much as possible.

The final goal of the thesis is not to provide a perfect, ready-to-implement disruption timetable, as this would require fully detailed information on each possible disruption. Rather, it aims at developing models which dispatchers can employ by providing only a limited amount of input data in order to calculate high-quality solutions for many parts of the infrastructure. In some cases, these solutions might require some fine-tuning to take into account all details about delays and the specific available infrastructure. Despite some simplifications, the solutions should prevent conflicts as much as possible, thereby increasing the likelihood of being conflict-free on the microscopic level. Hence, blocking time theory is used.

Unfortunately, no comparison of the model's solution with current practice is possible for two main reasons. On the one hand, public data on infrastructure and timetable are not sufficiently detailed for exact blocking time calculations. Moreover, only limited confidential data was at hand, requiring assumptions and simplifications for several aspects of the model's input. Wherever possible, this thesis provides recommendations on how to use real-life data to overcome (some of) these assumptions. On the other hand, Infrabel does not structurally report and file measures taken during disruptions. Hence, comparison with real-life solutions is not rendered feasible within the limited time available for this thesis research.

3.2 Subquestions and goals

Considering these elements, the central research question is split into three subquestions with their specific goals:

Q1 Which disruption management approaches exist both in scientific literature and current practice?

- Gather and present knowledge on disruptions, ways to conceptualize them, and how researchers try to deal with them.
- Description of current practice at Infrabel and in other countries.
- Motivation for, and framing of this thesis in both scientific research and current practice.

Q2 How can disruptions on AIPs be modelled and which situation-specific parameters have to be taken into account?

- Identification of AIPs on the Belgian railway network.
- Selection of parameters possibly influencing disruptions and disruption management and type of model to be developed.
- Development of a model for each AIP considered.

Q3 Which parameters strongly influence the results of the model, and how do they do this?

- Assessment of additional constraints on the model's results.
- Development and assessment of parameter impact for each AIP.

3.3 Methodology

Splitting the central research question into several subquestions provides a structure for the remainder of this thesis. Each of these goals require one or multiple methodologies in order to provide an answer to the research questions.

To answer question Q1, a literature study is conducted: Chapter 4 presents a synopsis thereof. A visit to the local signal box in Leuven [30] provided the necessary information for a description of current practice in Belgium. Next to covering the goals of Q1, theoretical concepts behind the model's development which were not yet introduced in Chapter 2, are presented.

Chapter 5 provides an answer to Q2 by example of a disruption on a corridor between two stations: in-between two double switches, a train breaks down. Based on the insights presented in Chapter 2 and additional field-specific knowledge gathered throughout the master's programme, parameters potentially influencing the solution are identified. A limited number of these parameters are selected. Next, the chapter continues with the development of a framework for the optimization model based on a practical case study of the first AIP. Finally, additional service requirements as they may occur in real-life are modelled as constraints and applied to the case study, partly answering Q3. Results obtained by the model are contrasted with two heuristics, representing expected dispatchers' decisions.

Using this model, Chapter 6 analyses the effect of several of the selected parameters on the generated disruption timetables. Parameters include both timetable and infrastructure characteristics. For reasons of simplicity and cross-scenario comparison, an artificial case study is used. Solving several scenarios allows to draw more general conclusions on which measures can be considered to be useful in practice.

Chapter 7 extends the first AIP by adding stops along the corridor, increasing the complexity of the disrupted situation. Next, the double-track corridor is generalized to an N-track one, potentially with stops located along the corridor (Chapter 8). Both chapters elaborate on the necessary extensions and adaptations of the model and framework presented in Chapter 5, by using practical case studies of the respective AIPs. Additionally, the analysis of parameters in Chapter 7 and 8 is limited to the new parameters not discussed in Chapter 6.

Finally, Chapter 9 critically reflects on the research questions and the used methodologies, after which Chapter 10 concludes this thesis research.

For each AIP, the framework is developed in Matlab® version R2016a [35] and passes the optimization model on to CPLEX® v12.6 [22] by means of the YALMIP toolbox [31].

Chapter 4

Literature Study

This chapter presents an extensive literature review to answer the first subquestion in Section 3.2. Section 4.1 provides several definitions of the terms *disruption* and *disruption management* to limit the scope of the remainder of this chapter. Next, Section 4.2 explains the bathtub model to conceptualize the evolution of disruptions, thereby identifying several challenges to handle them efficiently. Section 4.3 describes the current practice of disruption management in several countries, contrasted with a discussion on how Infrabel deals with this.

State-of-the-art models for the *Train Traffic Rescheduling* problem (TTR) are discussed in Section 4.4, constituting the main portion of this chapter. First, it discusses a number of models for disruption management, illustrating the innovativeness of the approach in this thesis. Next, an overview of both machine scheduling problems and simulation in light of train (re)scheduling is provided, as these may be used further on. Finally, Section 4.5 concludes the chapter and situates this thesis within both current practice and the state-of-the-art.

4.1 Definition of disruptions

Deviations in railway traffic show a high variability in terms of spatial and temporal occurrence, duration, their effect on railway traffic and service quality. In general, schedule deviations are called *delays* [21]. Depending on their size, delays may impact on each of the planning phases from timetabling to rolling stock and crew scheduling. If delays are relatively small, timetable rescheduling is sufficient to get back to regular operations without affecting the original rolling stock and crew schedules. Cacchiani et al. [9] categorize these as *disturbances*.

More severe are *disruptions*, but it is difficult to distinguish disruptions from disturbances, and no commonly agreed guidelines or criteria to identify them exist in literature. Moreover, definitions may even be conflicting when comparing two articles. Cacchiani et al. [9] define disruptions as “relatively large incidents, requiring both the timetable and the duties for rolling stock and crew to be modified”. This is in line with the definition by Hansen and Pachl [21], who also emphasize the large-scale effect. On the other hand, both Besinovic et al. [6] and Jespersen-Groth et al. [27]

do not set timetable rescheduling as a requirement. According to them, the incident "renders any of the resource schedules (infrastructure, rolling stock, and/or crew) infeasible" [6]. Narayanaswami and Rangaraj [37] limit disruptions to unavailability of tracks for (at least) a short time, whereas Louwerse and Huisman [32] consider the same situation but for a prolonged duration, e.g. more than 1 h. Although most research on disruption management only focuses on a single planning stage, the definition of a disruption by Cacchiani et al. [9] seems to be the most adopted one in recent literature, e.g. Dollevoet et al. [17].

Dealing with disruptions of any kind is referred to as *(railway) disruption management*:

> "the joint approach of the involved organizations to deal with the impact of disruptions in order to ensure the best possible service for the passengers. This is done by modifying the timetable, and the rolling stock and crew schedules during and after the disruption. The involved organizations are the infrastructure manager and the operators." [27]

This is a rather generic definition, which can be used regardless of the specific definition of a disruption. As cancellation of trains is a viable option during disruptions, rolling stock or crew schedules may become infeasible too. However, these are not considered within this thesis, only timetable rescheduling is considered.

4.2 Disruption evolution: the bathtub model

Louwerse and Huisman [32] compare the evolution of network utilisation in case of a disruption with a bathtub. Although no definition for *network utilisation* is given, the number of trains running seems to be an acceptable measure and in line with Ghaemi and Goverde [19]. Figure 4.1 distinguishes three phases based on the traffic level evolution. As most disruptions are unpredictable in terms of time, location and impact on traffic, their identification requires continuous monitoring by dispatchers [27], before their occurrence.

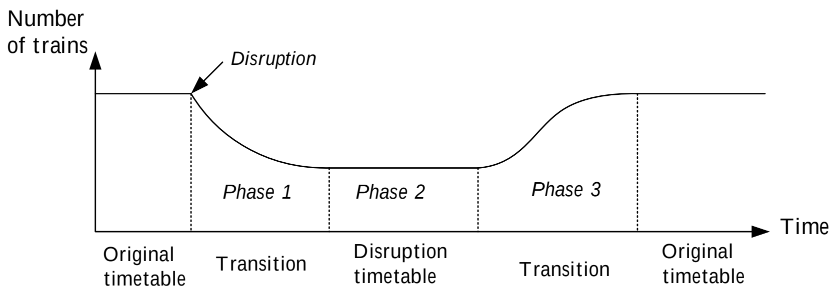


Figure 4.1: The bathtub model conceptualizes the evolution of traffic level during a disruption. It distinguishes three phases: the disruption timetable in phase 2 requires a transition phase before and after it. Based on [32] and [19].

After the start of a disruption, traffic has to evolve from the regular level to a lower, controllable, one: the *disruption timetable* in Figure 4.1. During this first transition phase, Chu and Oetting [11] and Ghaemi and Goverde [19] identify challenges such as estimating the duration, choosing appropriate plans for the second phase, communicating these plans to all parties involved and lowering the traffic level by removing trains from the network. Although the plan for the second phase may be the same, the reduction of trains highly depends on the current traffic, which complicates decision-making. Jespersen-Groth et al. [27] and Oetting and Chu [42] also point at issues arising when multiple operators are affected: the infrastructure manager has to coordinate the decision-making process, which may affect the response time.

In some countries, predefined plans are at hand for the second, stable phase (see Section 4.3). However, the duration of the disruption is highly variable and depends on multiple factors such as the incident causing it and availability of resources. Hence, not all situations are covered by these plans. In absence of a plan, dispatchers aim at isolating the disrupted area from the rest of the network. Near the end of the disruption, one has to prepare the return to the original timetable, which may include reinserting cancelled trains and clearing queues on a corridor. [19, 27]

Jespersen-Groth et al. [27] and Louwerse and Huisman [32] advocate the last transition phase to be less challenging than the first one. As a consequence of adjustments during earlier phases, rolling stock and crew may not be available at the right place at the right time, resulting in possible complications and prolonged duration of this third phase.

4.3 Current practice in disruption management

This section sketches the current practice in several countries based on available literature. Some infrastructure managers and operators have predefined plans to deal with disruptions, especially for the second phase in the bathtub model (Figure 4.1). These plans vary in terms of number of scenarios, considered measures and construction method. In some cases, decision support systems (DSS) are available for at least one of the planning stages, i.e. timetable, rolling stock or crew rescheduling.

4.3.1 The Netherlands: contingency plans

In the Netherlands, dispatchers have so-called *contingency plans* at hand, designed "to assist the traffic controllers in dealing with [...] disturbed traffic" [19]. Experienced dispatchers design the plans based on the original timetable and a certain disruption, allowing trains to be cancelled and short-turned. Figure 4.2 shows an example, with on top information specifying the situation for which the plan has been designed. Text within the red box lists two possible locations for which this plan is valid. A second part specifies whether a train series can be kept in operation or has to be cancelled. Here, one train series has to be cancelled due to the unavailability of one track on a four-track section. In general, designers do not consider delaying trains [32].

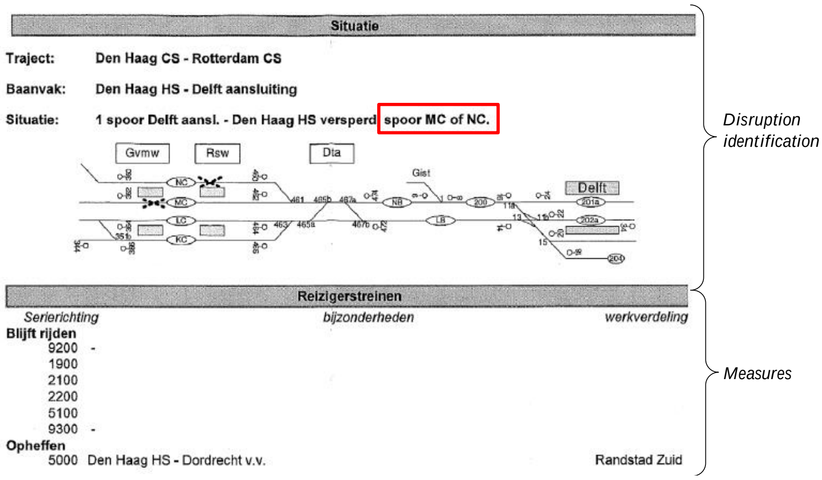


Figure 4.2: Structure of a contingency plan in the Netherlands, describing the situation(s) for which it applies and the measures to take. In Dutch, adapted from [12].

Cacchiani et al. [9] reported on the existence of more than 1,000 contingency plans at that time, from which the implemented one is selected in a non-automated way. The *Operations Control Centre Rail* (OCCR) brings together representatives of, amongst others, the infrastructure manager (ProRail), operators and passenger organisations, who decide together on the contingency plan to be implemented. From this centralized body, the representatives steer their local colleagues such as the regional dispatchers of ProRail [19].

Train operators are responsible for rolling stock and crew rescheduling. At Netherlands Railways (NS), the main operator, DSSs for these steps are under development and have been tested to reschedule resources in case of disruptions which can be anticipated, e.g. bad weather conditions, or last several days, e.g. a train derailment. Kroon and Huisman [29] state that the resulting solutions are at least as good as those obtained by dispatchers, but the main benefit lies in the faster computations.

4.3.2 Copenhagen: dispatching rules and emergency plans at DSB S-tog

DSB S-tog operates the suburban train network around Copenhagen, Denmark. Jespersen-Groth et al. [27] describe how the operator deals with disruptions, distinguishing between disruptions with a low impact and more severe ones. For this first group, basic dispatching rules may be sufficient to modify the timetable and recover from the disruption. An example is switching stopping patterns between a slow and fast train, if the fast one cannot overtake the slow one. Another possibility is the (partial) cancellation of train lines, which increases buffer times between other trains.

Emergency scenarios are defined to handle more severe cases, such as the one mentioned in Figure 4.3 for the complete blockage of a segment. For lines directly involved in the disruption, i.e. running through the disrupted area, the emergency scenario defines train cancellations, number of trains running and where they should turn. Compared to the Dutch contingency plans, the emergency scenarios also include turnaround times at the turning stations, as shown in Figure 4.3 (right). Hence, one can claim these plans are more specific, while operations are also expected to be less complicated due to the smaller network. According to Jespersen-Groth et al. [27], these plans are needed to prevent trains queuing up in the heavily utilized network.

4.3.3 Germany: disruption recovery programs

Disruption recovery programs (DRP) are in use at several urban railway operators in Germany. Chu and Oetting [11] define these as “pre-defined dispatching measures in case of certain (infrastructural) disruptions [...] with the goal of ensuring stable operations in a disrupted situation”, suggesting that they are designed for the second phase of the bathtub model. After occurrence of a disruption, collecting sufficient information from the field, adapting the plans, and communicating them, requires much time. Regardless of the needed adaptations, the DRPs provide a starting solution and basis for communication. Dispatchers themselves recognize these advantages, but the transition phases remain problematic, just as for the Dutch and Danish cases. [11, 42]

4.3.4 Japan: turning patterns for complete blockages

In Japan, some operators have tools at hand to support dispatchers by proposing new train orders in case of disruptions. Nakamura et al. [36] describe how Japanese operators in general deal with disruptions. Note that in Japan, multiple operators exploit their own infrastructure [58]. This is not practised everywhere and differs from the European context where infrastructure managers and operators are separated and have to interact.

A dispatcher estimates the duration and decides on train cancellation and short-turning. Next, another dispatcher within the traffic control room decides on the order between trains. Dispatchers’ knowledge and experience are used to create so-called *train-rescheduling patterns*, which can be used and adapted during the real

Line	Changed from and to	Canceled from and to
A+	Køge to Østerport	Østerport to Buddinge
H	Frederikssund to Dyssegård Buddinge to Farum	Dyssegård to Buddinge
H+	Frederikssund to Svanemøllen	Svanemøllen to Farum

Line	Traffic south of blockage	Traffic north of blockage
A+	Køge to Østerport Turnaround time: 10 min. Trains necessary: 6	Canceled
H	Frederikssund-Dyssegård Turnaround time: 19 min. Trains necessary: 8	Farum-Buddinge Turnaround time: 13 min. Trains necessary: 3
H+	Frederikssund-Svanemøllen Turnaround time: 16 min. Trains necessary: 8	Canceled

Figure 4.3: Example of an emergency scenario for the suburban network around Copenhagen during a blockage between two stations. The plan specifies both the changes in the line plan (left), and the turnaround times and number of trains (right). Taken from [27].

operations. Only specific parts are used, such as turnaround patterns in stations. In this sense, the way of working is similar to the one in the Netherlands, where experienced dispatchers draw up the contingency plans, although it is more flexible. Finally, both sets of measures are documented and sent to rolling stock depots and crew bases, taking care of their respective resources. [36]

4.3.5 Belgium: current practice at Infrabel

The *Railway Operations Centre* (ROC), located in Brussels, opened in 2014 and accommodates representatives of both Infrabel and NMBS with the goal of improving real-time operations for the complete Belgian network. Figure 4.4 sketches the organisational set-up, based on the yearly report of NMBS [39]. The ROC supervises operations and interacts with the 31 local signal boxes, which hold dispatchers of Infrabel, and NMBS representatives. Infrabel's Traffic Control takes the final decision, but in reality closely cooperates with three expert groups of NMBS. Each group deals with one stage of the rescheduling process. *Traction controllers* and *"Centrale Permanentie Treinbegeleiders"* (CPT) manage respectively the rolling stock circulations and crew rescheduling for the complete network. *"Dispatching reizigers" (RDV)* communicates information to crew members directly in touch with passengers and represents the passengers' interests. [39]

This set-up differs from the Dutch OCCR (Section 4.3.1) in three ways. Whereas the Dutch OCCR operates only in case of disruptions, the ROC is a permanent cooperation to deal with any deviation. Secondly, the ROC holds only representatives of NMBS and Infrabel, not of other operators. Finally, no predefined plans are at hand and decisions have to be taken ad hoc [30].

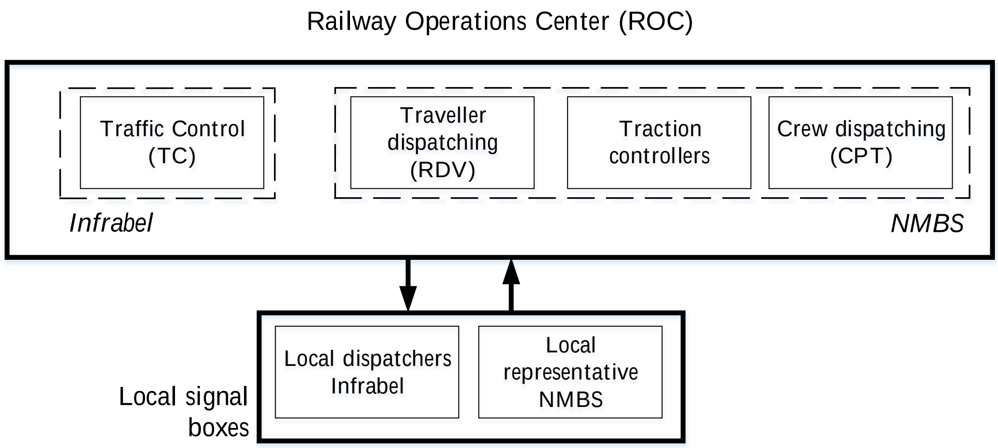


Figure 4.4: Organisational set-up for controlling railway traffic on the Belgian network. The centralized Railways Operations Centre interacts with 31 local signal boxes spread out over the country. Based on [30, 39].

When a disruption occurs, this is first noticed by a signaller of the local signal boxes. After identifying it as a disruption, the necessary information is sent to the centralized TC in Brussels, including type and location. Estimates of the disruption duration are experience-based, taking into account which resources may be required and where they are located.

TC coordinates train traffic at the network level, and negotiates with NMBS and potentially other operators involved, but not located within the ROC. Currently, no plans are at hand defining which measures to take. The goal of these negotiations is to keep the level of service as high as possible, i.e. limiting the effects on passengers, and to react as fast as possible. After agreement, plans are communicated to the local signal boxes.

Dispatchers take several types of measures and constraints into account. Main measures include re-timing, reordering of trains at stations, switching stopping patterns, globally re-routing trains, short-turning them at stations, implementing shuttle services and train cancellation. However, cancellation and stop-skipping are avoided as much as possible. Hence, all measures are taken ad hoc and depend on the dispatchers' knowledge of passenger numbers, timetable and network structure.

Lately, Infrabel took into operation its Traffic Management System (TMS), which is able to predict conflicts arising within the next 15 min [56]. However, it does not provide information on how to solve them. Currently, no tools are available at either the signal boxes or TC to evaluate effects of measures on the local and global traffic respectively. Moreover, no information on measures taken during past disruptions is at hand.

4.4 State-of-the-art models

In case of delays, timetable rescheduling has to be conducted. During the last decade, research has brought forward several Operations Research (OR)-based models for adjusting train traffic according to a predefined goal, such as minimizing the overall delay [45], minimizing the maximum delay [16], isolating the disrupted area, and minimizing the effects on passengers, including cancellations [6].

Cacchiani et al. [9] present an overview of models which can be applied to one or multiple planning stages: timetabling, rolling stock and crew rescheduling. Research is categorized according to the level of detail considered, i.e. microscopic versus macroscopic timetabling models, the number of planning stages, i.e. isolated versus integrated models, and the effects of delays, i.e. disturbances versus disruptions. They define the *Train Timetable Rescheduling* (TTR) problem as "adjusting in real-time an existing timetable that has become infeasible due to unpredicted disturbances or disruptions. The aim of TTR is to quickly re-obtain a feasible timetable of sufficient quality". Hence, TTR refers to both disturbances and disruptions, although most works in literature focus on a particular one.

First, this section presents several works relevant to railway disruption management. Both models for preparing for disruptions on beforehand, e.g. assessing contingency plans (Section 4.4.1), and handling them in real-time (Section 4.4.2), are discussed. Next, Section 4.4.3 discusses machine scheduling problems, possible extensions and how they can be used to solve TTR problems. Sections 4.4.4 and 4.4.5 present mixed-integer linear programming (MILP) models and rolling horizon approaches as alternatives to deal with general TTR, followed by simulation models in Section 4.4.6. Finally, Section 4.4.7 briefly discusses similarities between disruptions with capacity reduction and scheduling problems within other transport modes.

Table 4.1 shows an overview of literature discussed in Sections 4.4.1 and 4.4.2, categorizing them as models used for assessing (A) and building (B) contingency plans, or applicable for real-time disruption management (R). Note that the distinction between these last two is questionable. Here, models with short computation time are classified as R, while they could also be used to design plans. A second classification is based on severity of the considered disruptions: either complete (C) or partial (P) blockages. Additionally, Table 4.1 provides general information on the model, which measures are considered as decisions and which are fixed, time duration of the disruption and planning stage(s) taken into account. They serve as basis for the discussion in the remainder of the section.

The approach of dealing with the problem strongly varies, although most of them apply a common set of measures: train re-timing, reordering, local and global rerouting. When traffic is severely disrupted, also short-turning and cancellation come into play. Most models allowing train cancellation, use a weighted objective function to penalize train delays and cancellations. Possibly, also stop-skipping [60] is incorporated.

4.4.1 Models for assessing and designing contingency plans

Chu and Oetting [11] assess the feasibility of disruption recovery programs (DRPs) based on process (time) variability during the first transition phase. They identify the different steps required to implement a predefined DRP after the start of a disruption. For each of them, distributions for time durations are derived from dispatchers' experience or by process analysis [42]. Time variations may be due to train positions differing from the plan, or a central dispatcher as communication bottleneck. They result in increased blocking times and capacity consumption, as determined based on the UIC 406 method [55]. Oetting and Chu [42] conclude time-to-decision to be one of the main influencing factors, which is reduced when using DRPs as they ease up communication.

Contingency plans can also be evaluated by their effect on network traffic, e.g. to compare several variants. Corman et al. [12] assess disruption handling scenarios with partial blockage of a double-track line, using several performance indicators such as passenger waiting time and output delay. They use existing contingency plans as input for a tool called *Railway traffic Optimization by Means of Alternative graphs* (ROMA)[1]. Scenarios differ in the number of trains being cancelled, globally or locally rerouted. Hence, the remaining decisions include orders of trains and delays. The ROMA tool adapts these disruption scenarios to the current positions of

[1]More explanation on alternative graphs is provided in Section 4.4.3.

Table 4.1: Overview of research on railway disruption management described within Section 4.4. Models are classified by type (A = assessment, B = building contingency plans, R = real-time application) and disruptions considered (P = partial, and C = complete blockages). A short description of the model is provided, together with which measures it considers. Finally, the time horizon for the case studies considered and the planning stages on which the model acts, are reported.

	Type / disruption	Model description	Measures		Time horizon	Timetabling level/ planning stages
			Variable	Fixed		
[13]	A / P	Evaluation of several scenarios Alternative graph approach Minimize the maximum delay	Re-timing Reordering	Cancellations Local rerouting Global rerouting	1 h	Microscopic, periodic
[14]		Distributed vs centralized MPC			0.5 - 1.5 h	
[11]	A / C	Evaluation of time variability in first phase Assess feasibility DRPs	-	DRP	(first phase)	Microscopic
[32]	B / PC	Event-activity network Weighted objective function	Re-timing Reordering Short-turning Cancellation	Routes	2 h	Macroscopic, including details on capacity
[36]	B / C	Provide dispatcher with initial solution Algorithmic construction of timetable	Reordering Short-turning Cancellation	Rescheduling patterns	-	Macroscopic
[60]	R / P	MIP model to reschedule high-speed traffic	Re-timing Reordering Stopping pattern	Routes Strategy	24 h	Macroscopic
[57]	R / PC	All phases of the disruption Weighted objective function Event-activity network	Re-timing Rerouting Short-turning Cancellation	Rerouting options	-	Macroscopic
[17]	R / PC	Iterative framework Timetabling: minimize cancellation and delay RS + crew: cover as much as possible	idem Assignment RS Assignment crew	idem	1 - 2 h	Macroscopic RS + crew
	A / PC	RS and crew feasibility of contingency plans		Contingency plan	-	RS + crew
[6]	B / C	Iterative framework Feasibility on microscopic level	idem [17]	-	2 h	idem [17] Microscopic

trains, resulting in certain values for the performance indicators. In a follow-up work, Corman et al. [13] assess the performance of both distributed and centralized model predictive control (MPC). By splitting up the network in several parts, in line with common practice of having local signal boxes, the problem size reduces. A global dispatcher coordinates measures at the border of the control areas. This method succeeds in finding more feasible solutions quickly, but may result in suboptimal results.

The Dutch contingency plans (Section 4.3.1) are designed without considering re-timing of trains. Louwerse and Huisman [32] present a MILP model to draw disruption timetables for the second phase, starting from an existing timetable. They include aspects of rolling stock circulation and microscopic routing in their macroscopic model by means of inventory constraints for train units and tracks respectively. As their model allows delays, resulting disruption timetables have more trains running compared to existing contingency plans. However, increasing maximum delay leads to extensive computation times and renders the model unsuitable for real-time application. Nonetheless, it could be used to design improved contingency plans.

Nakamura et al. [36] present an algorithm to build a disruption timetable in case of complete blockage of a railway line, using three building components: train types (TG), partial cancellation of a train (TCS) and turning patterns (TBP). These *train-rescheduling patterns* are based on dispatchers' knowledge and used by a heuristic algorithm to draw up a disruption timetable at run-time. The framework in Figure 4.5 is used to evaluate the generated plans. After the incident, the dispatcher collects information about the disruption, e.g. location and an estimate of duration, and provides this information to a DSS. Next, the DSS' algorithm uses the predefined building components to draw up a solution, which has to be verified by the dispatcher. Finally, a microsimulation model is used to simulate the plan to come up with the actual train re-timing.

These authors consider the framework to be applicable to any part of the network, given that the building components have been defined for those locations. Their

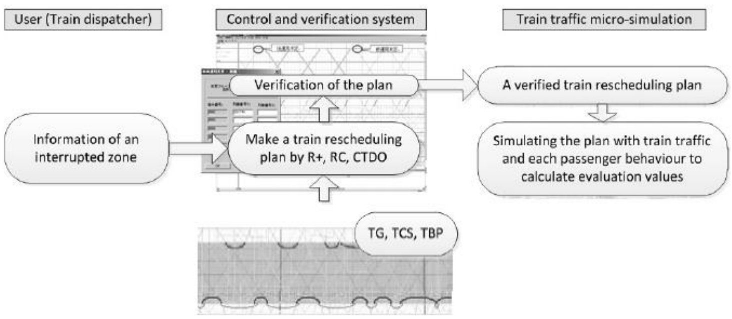


Figure 4.5: Evaluation framework for the algorithm developed by Nakamura et al. [36], in which a dispatcher characterizes the disruption and retrieves a disruption timetable shortly after. Taken from [36].

algorithm does not aim at providing the optimal solution by considering a wide range of measures. Its primary goal is to support dispatchers by returning understandable and intuitive plans within a short computation time.

The framework sketches a possible implementation for the models developed in this thesis. However, their approach is fundamentally different: the rescheduling patterns serve as fixed input to the algorithm. This thesis aims at determining a disruption timetable at run-time for given input data on the timetable, infrastructure and disruption, by considering a wider range of possible measures. Hence, the found solutions could be further specified on the microscopic level, and for the remainder of the network, by Infrabel's traffic management system (TMS) (see Section 4.3.5).

4.4.2 Models for real-time disruption management

Zhan et al. [60] develop a model to deal with the partial blockage of a high-speed line under several strategies, one of which to let both directions use the remaining track. Other strategies only allow one train at a time on the affected section, or forbid trains in the blocked direction from running. The latter strategy shows similarities with current practice at Infrabel [30]: trains in the less-crowded direction only run over the area if gaps exist in the timetable. First, Zhan et al. [60] present a MILP model which can mimic all strategies. For long disruptions, during which multiple trains have to be cancelled, large optimality gaps existed after 10 min. Hence, they switch to a two-stage heuristic approach if no good solution has been found after 10 min. Their experiments show that solutions obtained within 180 s score at most 5% worse than the ones after 600 s.

Veelenturf et al. [57] build on the work by Louwerse and Huisman [32] and manage to reduce computation times to levels acceptable for real-time application. However, they severely limit the maximum allowable delay to 10 min. One may question whether this is a valid assumption as delays can easily accumulate to larger values during disruptions. It may be too restricting for real-life situations. Additionally, it is a macroscopic model, not guaranteeing feasibility on the microscopic level. Their macroscopic MILP model has been embedded into two iterative frameworks for disruption management by Besinovic et al. [6] and Dollevoet et al. [17].

Models considering multiple planning stages at the same time, can be categorized as either integrated models or iterative frameworks. The former develop a single model which contains constraints and variables for all considered planning stages. However, it may be difficult to define a consistent objective function dealing with goals of all planning stages. Iterative frameworks solve the problem in sequential steps, with feedback loops to former planning stages as shown in Figure 4.6. They may require multiple iterations and one has to ensure that results are consistent and feasible at every planning stage.

Besinovic et al. [6] and Dollevoet et al. [17] incorporate existing models for each phase of rescheduling into the framework of Figure 4.6. For the macroscopic timetabling component, the model by Veelenturf et al. [57] is used, with a high train cancellation penalty. Next, rolling stock and crew rescheduling modules take care of

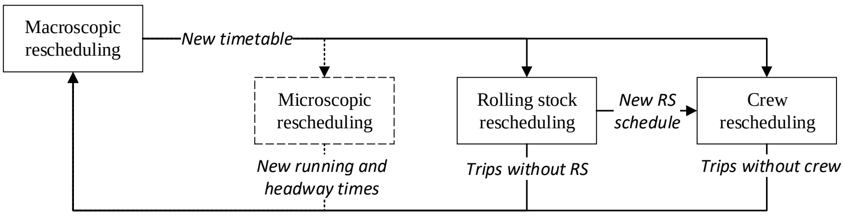


Figure 4.6: Iterative framework for timetable, rolling stock and crew rescheduling during disruptions, used by Besinovic et al. [6] and Dollevoet et al. [17]. The latter does not include the microscopic rescheduling module (dashed parts). The framework includes feedback loops between all planning stages to ensure consistency and feasibility. Based on [6, 17].

covering as many trips as possible within the generated timetable. Uncovered trips are cancelled, and the process is repeated.

They differ in the use of a microscopic model. Dollevoet et al. [17] does not incorporate such a component, and their model solved 99% of instances to feasibility within two iterations and 10 min. On the other hand, Besinovic et al. [6] uses heuristic approaches for local train rerouting and updates running times. In case the macroscopic timetable is not conflict-free at the microscopic level, headway times are calculated and returned to the macroscopic timetabling module. The advantage is that the resulting timetable is feasible on the microscopic level. However, it requires more iterations, which consumes more computation time compared to the framework by Dollevoet et al. [17].

Dollevoet et al. [17] report that their tool is currently applied as a way to evaluate whether feasible rolling stock and crew schedules can be found for the existing contingency plans in the Netherlands. Hence, the framework can also be classified as "assessing" in Table 4.1.

Theoretically speaking, the models developed within this thesis could also be incorporated in such iterative frameworks. Although the goal is to come up with conflict-free disruption timetables, further detailing of the solutions found, may still lead to (minor) conflicts, especially outside the considered corridor. Rolling stock and crew rescheduling are also not considered, as the models act on a small portion of the complete network.

4.4.3 TTR as a job-shop-scheduling problem

Some models that deal with the TTR are based on machine scheduling problems, e.g. the alternative graph approach used in ROMA by Corman et al. [13, 14] and D'Ariano et al. [16]. This section introduces basic concepts of the machine scheduling problem, together with additional constraints required by the specifics of train operations.

Machine scheduling problems

Machine scheduling problems consist of scheduling a set of jobs $J = 1, ..., n$ on a set of machines $M = 1, ..., m$ in an optimal way according to some evaluation criterion. The processing of a job j on a machine k is considered to be a task i, and requires a process time p_{ijk}. If t_{jk} denotes the starting time of job j on machine k, its *completion time* on this machine is $C_{jk} = t_{jk} + p_{ijk}$. [2]

Problems are classified based on the number and type of machines and tasks. Single-machine scheduling problems deal with scheduling jobs, each with one task, on one machine. In case multiple machines can perform this task, it becomes a parallel-machine scheduling problem. Scheduling jobs with multiple tasks to be conducted in the same order on the same set of machines, is called flow-shop scheduling. Finally, the most general one, job-shop scheduling (JSSP), allows tasks to follow any path along the machines.

Together with a criterion such as minimizing maximum or total *flow time*, i.e. the "time job j spends in the system" [2], these components constitute the most basic machine scheduling problem. Solving real-life problems however requires several additional constraints and problem extensions.

Each job j may have a release time r_j, i.e. the earliest time its first task can be started, and a due date d_j, i.e. the time it should be ready. This allows to define additional criteria such as minimum lateness, i.e. "time by which the completion time of job j exceeds its due date". [2]

Additionally, some (sequence-dependent) set-up time s^k_{jl} may be required before machine k can start with job l after processing of job j. Research has brought forward several alternatives to formulate these constraints. Nogueira et al. [41] and Rocha et al. [49] conduct computational analyses of several MILP formulations for respectively single- and parallel-machine scheduling problems. Both conclude that formulations based on the Manne concept [33] are the most efficient in terms of computation times. Nogueira et al. [41] present the following MILP formulation for the single-machine scheduling problem:

$$C_j \geq C_i + s_{ij} + p_j - M_{ij}(1 - \gamma_{ij}) \qquad \forall i, j \in J, i \neq j \tag{4.1}$$
$$\gamma_{ij} + \gamma_{ji} = 1 \qquad \forall i, j \in J, i < j \tag{4.2}$$
$$C_j \geq r_j + p_j \qquad \forall j \in J \tag{4.3}$$
$$C_j \geq 0 \qquad \forall j \in J \tag{4.4}$$
$$\gamma_{ij} \in \{0, 1\} \qquad \forall i, j \in J, i \neq j \tag{4.5}$$

The binary decision variable $\gamma_{ij} = 1$ if job i is processed (not necessarily right) before job j and 0 otherwise. Constraints (4.1) enforce minimum set-up times between the completion times of both jobs. For the parallel-machine version, Rocha et al. [49] use the starting times. Constraints (4.2) express symmetry: if job i is processed before job j, i.e. $\gamma_{ij} = 1$, then job j is not processed before job i ($\gamma_{ji} = 0$). Rocha et al. [49] replace this one by using two constraints with γ_{ij}, implicitly exploiting symmetry. A job cannot be completed before a certain process time after its release time has expired (Constraints (4.3)). Constraints (4.4) and (4.5) impose bounds on the decision variables. This formulation can be supplemented with a scheduling criterion. [41]

When more resources can be assigned to a task, its process time can be reduced. These resources come at a cost, e.g. support tools or higher energy consumption. The problem of balancing the extra costs with the standard objectives of a machine scheduling problem, is referred to as *scheduling with controllable process times*. Two categories exist: continuously varying and discrete process times. Chen et al. [10] describe the single-machine scheduling problem with discrete process times, and present algorithms for several criteria. Essentially, this problem can be interpreted as process times themselves becoming decision variables, next to the order and starting times of jobs on the machines.

Finally, the *scheduling problem with rejection* extends the basic models with the possibility to reject a job. Minimizing, for example, the total lateness of a set of jobs, may render this beneficial for jobs with long process times and limited profits. Hence, other jobs can be processed faster without violating their due date. Bartal et al. [3] introduce the *scheduling problem with rejection* allowing the decision of not scheduling a job for a certain penalty. The decision represents balancing between worsening the scheduling criterion and paying penalties for rejected jobs. Zhang et al. [61] consider the single-machine variant with release times, minimizing total makespan of the accepted jobs on the one hand, and penalties for the rejected jobs on the other hand. They prove the problem to be NP-hard.

TTR as a job-shop-scheduling problem: the alternative graph approach

Rescheduling railway traffic can be considered as a machine scheduling problem with trains being jobs, machines being block sections, resulting in tasks representing the running of a train over a block section. As train routes may differ, i.e. different set of tasks for each job, TTR can be interpreted as a JSSP as follows. Release times correspond to the earliest arrival on a block section, e.g. the scheduled departure time from a station. Process times relate to the minimum running time of a train over a block section. Train separation is taken into account by the set-up time. However, this interpretation does not necessarily take into account all possible operational restrictions yet.

Additional constraints are vital to thoroughly capture the block section operation: trains cannot wait in-between two subsequent block sections. Mascis and Pacciarelli [34] present an extension of the basic JSSP in case no buffer capacity is present to store jobs in-between subsequent tasks on different machines. As a result, a job cannot leave a machine before its next task can be started (*no-wait constraints*). Hence, completion of the task is postponed and the machine (block section) can also not start on another job (train), resulting in so-called *blocking constraints*. To solve the problem, Mascis and Pacciarelli [34] develop the alternative graph approach.

D'Ariano et al. [16] adopt this approach for train rescheduling purposes and to solve conflicts in the ROMA tool. Rerouting trains alters the alternative graph. Therefore, Corman et al. [14] develop a tabu-search algorithm to include train rerouting in the ROMA tool: the output of the alternative graph model is used to find possible improvements in routes by searching different neighbourhoods of the existing routes. It improves the performance of the ROMA tool, even within shorter computation times.

4.4.4 Mixed-integer linear programming models for the TTR

Törnquist and Persson [54] present a model able to handle general infrastructure configurations consisting of a corridor with segments differing in the number of tracks. They incorporate re-timing, reordering and local rerouting to deal with minor traffic disturbances. Local rerouting boils down to choosing one of several parallel tracks in each segment. As they only consider small disturbances, the main decision is on the order of trains. They apply several strategies to limit the search space: the more freedom the strategy provides to the order decision, the more time it takes to find the optimal solution. Computation times reach unacceptably high figures for higher initial delay values, and longer scheduling horizons. The authors claim that corridors with any kind of infrastructure can be considered by their model.

The work by Törnquist and Persson [54] shows similarities with what this thesis aims at: providing flexible models to reschedule traffic. However, they consider less severe and small disturbances and still run into problems with computation times. Nonetheless, their way of modelling may be interesting to this thesis.

Pellegrini et al. [45] describe the RECIFE model, which is a MILP formulation to re-time and reorder trains on the microscopic level during disturbances. The model uses the prevailing network circumstances as input, including speed and position of all trains. In a second step, local rerouting is taken into account by providing a set of alternative routes for trains. Due to the time limit restrictions for both steps, this is a heuristic approach. The authors incorporate their model within a closed-loop optimization, meaning that every x seconds (*optimization frequency*), the problem is solved over the coming y seconds (*optimization horizon*). Rodriguez et al. [50] focus on fine-tuning these parameters.

Pellegrini et al. [45] also test the RECIFE-MILP heuristic on disruptions involving the unavailability of small parts of infrastructure within a station area or junction. Due to the need for local rerouting and the resulting large search spaces, computation times for solving the model to optimality increased drastically with the number of trains considered. Hence, it is not suited for real-life application yet.

4.4.5 Dealing with changing conditions: rolling-horizon approaches

Quaglietta et al. [46] present the ON-TIME framework for real-life application of both the ROMA tool [14, 16] and RECIFE model [45, 50], incorporating interactions with changing traffic states, dispatchers, route-setting systems and train drivers. The traffic management part implements a rolling-horizon approach. At regular intervals, it collects information on current train speed and position, solves the model over the next time horizon, and presents the solution to the dispatcher. Upon his or her approval, routes are automatically set and information is communicated to train drivers. Care is taken to ensure solution consistency during subsequent intervals. Based on results for three case studies, they claim that the "framework strongly reduces delays independently of the network topology, the traffic pattern, [and] the perturbation considered".

For the models developed in this thesis, the dispatcher has to provide a rescheduling horizon. However, estimates on disruption duration are highly uncertain. After using the (predicted) traffic state as input for the model, additional process variations may result in the calculated solution becoming suboptimal or even infeasible. Hence, the rolling-horizon approach presents an interesting method to deal with such uncertainties.

4.4.6 Use of simulation in traffic rescheduling

Next to optimization models, also simulation tools have been developed to reschedule train traffic.

Van Thielen et al. [56] present a discrete event simulation tool which incorporates microscopic details on a large part of the Belgian network. The tool allows to predict and detect conflicts, and to embed a heuristic to solve them. Heuristics include a first-come-first-served (FCFS) and priority-based approach, next to a newly developed heuristic, which minimizes the total duration of secondary conflicts created by solving the prevailing one. Scenarios included trains entering with minor delays and resulted in the FCFS outperforming other heuristics in terms of average (weighted) delay. This was attributed to low saturation levels and large buffer times for the trains running. The final goal is to embed the tool within Infrabel's TMS, so as to propose solutions to dispatchers.

The problem of rescheduling traffic in case of a disruption shows similarities with adjusting train traffic according to maintenance works. Both result in the current timetable to become infeasible, requiring adjustments to the original timetable. Whereas the spatial effect of a broken train and the closure of one track on a double-track corridor is inherently the same, the temporal one differs due to the uncertainty associated with disruption duration.

Bošković et al. [7] develop a simulation tool for line capacity assessment in case of maintenance on one track on a double-track corridor between two stations. The main question is whether all traffic can still be operated on the remaining track at a lower speed, given incoming trains may have delays. They consider global rerouting, cancellation and train coupling as measures. Moreover, they assess the premise of *train batching* as possibility to increase line capacity, i.e. trains in the same direction pass right after each other. Within the batch, trains are ordered according to a FCFS strategy based on the simulated arrival times. Stating that the "complex problem [...] cannot be resolved by merely applying analytical methods", they simulate a wide range of possible sets of measures and assess line capacity.

4.4.7 Similarities with other problems

The partial blockage of a double-track corridor of a railway network results in the need to schedule trains arriving from both ends of the disrupted area on the remaining single track. Similar problems arise in different areas and for other transport modes, such as scheduling single-lock operations on a canal, or scheduling take-off and landing operations of aircraft.

Passchyn [44] presents a model for the *lockmaster's problem* at a single lock with ships arriving from both sides. During a *lockage*, the lock moves from one side to the other with a number of ships confined within it. The basic problem exists of assigning ships to a set of lockages with an unlimited capacity and lockage duration T, which does not depend on the number of ships. The goal is to minimize the total (weighted) waiting time of all ships. The author models the problem by means of a graph, with nodes representing the time at which the lock is moved and arcs carrying costs equal to the additional waiting time. A shortest path algorithm is used to solve the problem. Results are compared with several (intuitive) heuristics, based on current practice, among which the "wait until threshold". Waiting a certain time after the last arrival before moving the lock, may allow an additional ship to be processed. However, Passchyn [44] states this may result in difficulties handling strong unidirectional traffic.

Airport runways may also serve aircraft in two directions, and those taking off and landing require sufficient time separation. The problems of scheduling one of both, are referred to as the Aircraft Take-off Problem (ATP) and Aircraft Landing Problem (ALP). Bennell et al. [5] present an extensive review and conclude that most models at that time only dealt with the isolated problems. A solution to either the isolated or combined problems contains decisions on the sequence, timings and runway assignment. However, due to fuel and speed constraints, holding aircraft in the air is difficult, resulting in narrow time windows. Machine scheduling formulations have been proposed to solve the combined ATP/ALP. Beasley et al. [4] interpret it as a job-shop scheduling problem with sequence-dependent set-up times. As for the TTR, D'Ariano et al. [15] develop an alternative graph approach to combine the runway scheduling problems with the scheduling of other capacity-restricted resources within the airport such as taxi-lanes and gates.

Both problems show a high degree of similarity with scheduling bidirectional train traffic on a single-track line, although differences exist. First of all, cancelling ships or aircraft is not considered in any of the models, whereas it becomes a valuable measure during a railway disruption. Second, overtaking of trains in the same direction is not always possible and requires physical infrastructure. Although aircraft are restricted to follow certain paths, it is easier to allow overtaking. Third, rescheduling railway traffic is not confined to short time windows as in the ATP/ALP, possibly increasing the likelihood of finding feasible solutions, but also enlarging the solution space. Finally, locks can simultaneously move multiple ships without changing lock duration. For railways, increasing the batch size leads to a longer processing time.

4.5 Conclusion

Disruption management is concerned with providing the best possible service during unexpected events and the consequences arising from it, such as a partial blockage. However, these disruptions are unpredictable in terms of occurrence and impact and lead to a chaotic phase before obtaining control over the traffic level again.

A number of railway operators control this second phase by implementing pre-defined contingency plans. However, several disadvantages exist. They have been designed based on dispatchers' knowledge for a specific, possibly outdated, timetable. During real-life operations, plans have to be adapted to the prevailing situations such as delays. Secondly, dispatchers mostly cannot grasp the effect of delaying one train on all others, which may be either beneficial or adverse. Finally, not all situations are being covered. Nonetheless, these contingency plans ease up communication during disruptions and shorten the first chaotic phase. At Infrabel, such decision support does not exist yet: rescheduling is done on an ad hoc basis. The goal of this thesis is to support their dispatchers by providing a high quality solution a short time after observation of the disruption. The concept of AIPs renders the developed models suitable for multiple locations on the network.

In research, several models have been developed, which in general render good results, but most of them still lack in computational efficiency. Hence, search space is reduced by for example limiting the set of alternative routes, imposing a maximum delay and time horizon, or by allowing only certain strategies. Satisfactory results may also be obtained by heuristic approaches or limiting computation time. A main stream of research develops and applies variants of machine scheduling problems for timetable rescheduling purposes. Most interesting is the flexibility of these models to increase the similarities with railway operations. On the other hand, also simulation can be an interesting approach within this thesis.

Several promising frameworks have been proposed to interact with dispatchers, and to integrate timetable rescheduling with other planning stages. Additionally, a rolling-horizon framework may effectively deal with the uncertainties associated with disruptions. Each of these frameworks can serve as inspiration for future applications of models developed in this thesis research.

Although the amount of research has shown a strong increase over the last decade, the approach taken within this thesis research has not been explored yet. Therefore, this thesis presents an innovative addition to existing state-of-the-art models, while being complementary to current practice at Infrabel, i.e. the absence of pre-defined plans, but existence of the TMS.

Chapter 5 continues with the development of the model for the first AIP. The knowledge presented here supports several aspects of this process.

Chapter 5

Partial Blockage of a Double-Track Corridor - Model Development

Chapter 4 presented a wide range of disruption management models, none of them developed for a concept such as AIPs. Using the acquired knowledge, this chapter deals with all steps from selecting parameters and model type until model development and application. As such, it (partly) answers the second and third subquestion of this thesis (see Section 3.2) by illustration of a specific, practical case study.

Section 5.1 describes the first AIP: a double-track corridor with double switches on each side. The disruption results in a closure of one track. Next, Section 5.2 elaborates on all parameters influencing operations during the considered disruption. It selects the most important ones, and describes assumptions for others. Section 5.3 selects the type of model to be developed in this thesis based on the literature study. Thereafter, Section 5.4 continues with sketching the model framework, i.e. how both available and dispatcher's inputs are translated to a disruption timetable, and introduces the case study. Section 5.5 interprets the rescheduling task as an extended single-machine scheduling problem, and describes the required pre-processing steps. After developing a first model and applying it on the practical case study, a number of additional "service" constraints are presented to verify their influence on the obtained results. To evaluate the model's performance, Section 5.6 presents two heuristics resembling dispatcher's behaviour. Section 5.7 concludes the chapter, and indicates how the model could be employed in a practical set-up.

5.1 Archetypical infrastructure piece description

Figure 5.1 shows the first AIP considered, which consists of a double-track corridor between two main stations, with at least two double switches along it. Between these switches, a disruption occurs and one of both tracks is unavailable for operation during a certain time. Causes can include a signal breakdown or a broken train. As a result, it remains possible to operate traffic over the other track, potentially at a lower speed: drivers may be asked to pass the disruption's location at a reduced speed. In the remainder, *disrupted area* refers to the block sections on which operations are affected, i.e. segments S_1, D and S_2. To keep the model as general as possible, the stations[1] at both ends are not incorporated in the model (dashed lines in Figure 5.1).

The resulting corridor can be divided into five segments, each consisting of a number of block sections. The outer two, A and B, connect the stations with the disrupted area. However, certain occurrences of the first AIP do not hold these segments: switches S_1 and S_2 directly connect segment D with the respective station areas. Assume the disruption occurs on the top track of segment D. Trains running in direction 0, from right to left in Figure 5.1, can still use their normal route. Trains in direction 1, have to be rerouted locally over the other track in segment D, and run over it in counter-track regime.

This first AIP is used to develop the model framework and examine all parameters that come into play when rescheduling train traffic during disruptions. A case study on line 36, connecting Brussels and Liège (see Figure 5.2), serves as illustration for the different steps in the reasoning. Consider a train breaking down between stations Tienen and Landen, respectively the left and right ends in Figure 5.1, blocking one of both tracks. One can see that between Tienen and the disrupted area (indicated in red in Figure 5.2), no block sections are present. Segment B holds at least one block section and two stops, i.e. Ezemaal and Neerwinden. For this AIP, these stops are disregarded. After choosing the model and developing its framework, Section 5.4.1 provides more information on the exact situation.

[1] Note that the stations in Figure 5.1 may also be replaced by a junction of merging or diverging corridors. Hence, the model is more generally applicable than merely on a corridor between two stations.

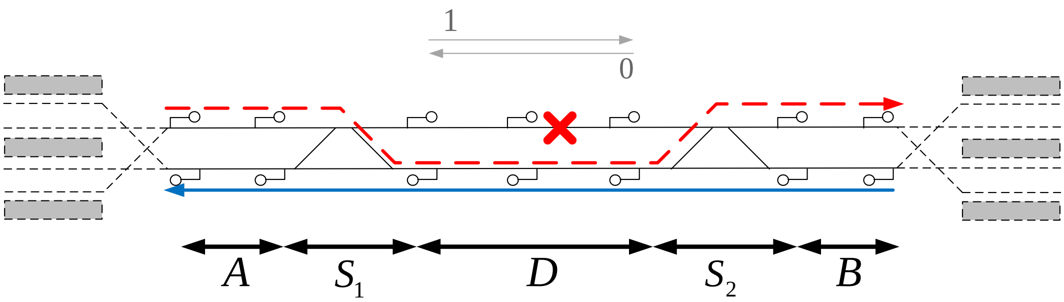


Figure 5.1: Sketch of the first AIP considered, consisting of a double-track corridor between two border points, e.g. stations (as displayed here) or a branching junction. Along the corridor, two double switches S_1 and S_2 allow trains to change tracks during a blockage of one track on segment D. Trains running in direction 1 (red dashed line) normally run on the upper track, those in direction 0 (full blue line) on the lower track. Marks along the tracks represent signals, which can show aspects in both directions.

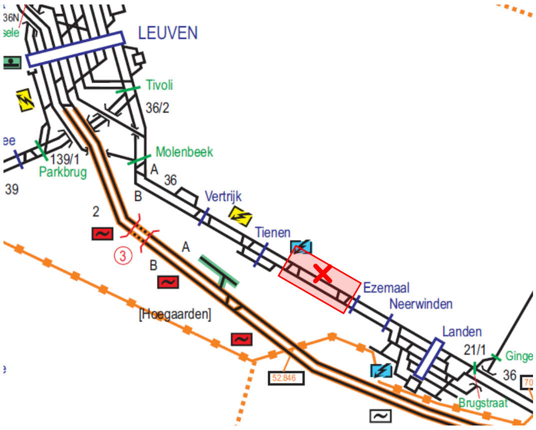


Figure 5.2: Indication of the Tienen-Landen case study on line 36, running from Brussels (top left) to Liège (bottom right), for the first AIP. A train breaks down between the switches east of station Tienen and west of stop Neerwinden, representing S_1 and S_2 respectively. The red rectangle indicates the disrupted area. Adapted from [26].

5.2 Parameters influencing disrupted operations

Parameters related to several aspects such as the infrastructure and timetable, influence the disruption timetable to a greater or lesser extent. Although in reality strong interrelations exist between them, all parameters are related to one of five categories: infrastructure, trains, timetable, disruption characteristics and passengers. Most of them influence results by altering (components of) blocking times. Several have an impact on running times, which can be either calculated by the model or provided as input. The latter is advisable for real-life application. Unfortunately, they were not at hand and running times have to estimated.

Infrastructure and signalling

Infrastructure parameters are characteristics of those elements physically present, regardless of trains. The safety and signalling system is the main determinant for blocking time calculation and closely related to the infrastructure through signals. However, the safety system is fixed and cannot be varied: the three-aspect-two-block signalling (Section 2.2.1) forms the foundation for blocking time calculation.

Parameters related to the infrastructure outside of the corridor, cannot be taken into account when developing a general model. As station capacity depends on the specific configuration, their capacity is assumed not to be a limiting factor in this thesis. One assumes it is feasible to build up an infinite queue of trains outside the disrupted area, which is considered to be a valid assumption in many, if not all, practical cases. If not, trains can also be buffered in stations further downstream of the corridor. Note that stops may also be situated along the segment D, which is considered for the second AIP (Chapter 7).

Neighbouring corridors may allow to globally reroute a number of trains. This is modelled by a penalty representing the additional travel time of the alternative route. However, the number of trains that can be rerouted is limited since the trains have to fit in with regular traffic on the alternative route.

A first group of parameters arises from the configuration of the corridor and its block sections. For segment D, the main influencing characteristic is its length: running times increase with increasing length. Especially the number of block sections and their specific lengths are determining factors for minimum headway times between trains.

Infrastructure speed limitations influence the running times. Switches S_1 and S_2 may impose different speed restrictions depending on the direction a train crosses them: running straight allows a higher speed than changing tracks. Finally, gradients and curves impact on running times as they may slow down a train. Here, they are not taken into account when calculating running times.

Trains

Train dynamics come into play when calculating the blocking times of a specific rolling stock type: maximum train speed (v_t^{max}), acceleration (a_t) and braking rates (b_t). Note that the latter two depend on the presence of slopes and curves, and their values are expected not to differ too much among train types.

For segment D, trains are assumed to have a constant speed, resulting in only train speed to have a major impact. Moreover, a number of different rolling stock types may be in operation, even for the same *service type*, i.e. intercity (IC) or local (L). Abril et al. [1] prove that "in single-track lines, capacity is more affected by train speed than by heterogeneity". Hence, one could make abstraction of all possible rolling stock types and differ only between IC and L trains.

Freight traffic is not considered as its importance relative to passenger traffic depends on a large number of factors such as its load and safety restrictions. In general, these trains have a low priority and can be stored on a yard nearby for the duration of the disruption.

Timetable

Abril et al. [1] and UIC [55] report that timetable structure is a major determinant for capacity. During a disruption, original arrival times, train order and scheduled stops restrict the possible solutions. Additionally, scheduling a timetable originally built for a double-track corridor, on a single-track one, is likely to result in conflicts.

Frequency in both directions is expected to be one of the major influencing parameters, as at some point it is no longer possible to schedule all traffic without incurring high delays. Both the number of trains per direction as well as their distribution over time impact on the best possible solution. For example, a regular timetable with uniform headways between trains over a full hour versus one with all trains running within a 20 min interval, may lead to different results for the same set of measures.

Including buffer times in the timetable has two effects. On the one hand, those in the original timetable may mitigate the knock-on delays. On the other hand, requiring buffer time in the disruption timetable may decrease the number of trains still operating on it. Theoretically, disruption timetables can be determined without requiring buffer times to reduce the minimum headway time between trains. However, train arrivals may deviate from their scheduled time, resulting in a snowball effect if no buffer has been incorporated in the disruption timetable.

Disruption characteristics

Disruptions differ in terms of location, duration and effects on traffic. For the situation in Figure 5.1, no significant differences are expected between closure of the track in direction 0 or the other one. The sole difference is a (slight) increase in running times for trains which have to diverge to the other track. More impacting on running times is the maximum allowable speed next to the location of the disruption, influencing running and blocking times. This *disruption speed* v_D may be lower than maximum speeds of all rolling stock types. In those situations, trains of both types run over segment D at the same speed.

Finally, disruption duration significantly influences the effectiveness of measures, e.g. a cancellation may not be required for short disruptions. As duration is highly variable, a dispatcher considers a certain *scheduling horizon* over which rescheduling has to be done, e.g. based on estimates, rather than claiming perfect knowledge.

Passengers

During a disruption, the goal should be to minimize the effects on passengers. Both passenger numbers, and how they perceive different measures, influence the optimal disruption timetable. In general, waiting 30 min because of a cancelled train, is perceived worse than a delay of 30 min [20], and cancelling or delaying a crowded train is worse than taking the same measure for an empty one. Since no satisfactory estimates of passenger numbers exist [52], model parameters should reflect relative rather than absolute numbers.

5.3 Model type selection

This section discusses the advantages and disadvantages of both simulation and optimization approaches with regard to the considered problem, which allows to make a justified choice.

5.3.1 Simulation versus optimization

Rescheduling trains during disruptions with reduced capacity can be tackled with similar techniques as those to evaluate railway capacity presented by Abril et al. [1]. For this thesis, two options are at hand, each with their advantages and disadvantages:

1. Using a simulation tool for part of the Belgian network as developed within the work of Van Thielen et al. [56] (see Section 4.4.6), or constructing a new one.
2. Developing a mathematical model with several input parameters and possible measures as decision variables to optimize a defined objective function.

Table 5.1 distinguishes the most important input parameters based on the previous discussion, from measures frequently employed in both state-of-art models and current practice and relevant to this (and other) AIPs. These are important for a justified discussion and choice.

The merits of simulation lie within the power of representing the situation with a high level of detail, simultaneously incorporating effects of possible entrance delays [56, 59]. However, the level of detail depends on the quality and detail of input data, and the output exclusively depends on the set of measures used as input, or the applied heuristic. Only limited data was at hand for this thesis, and altering aspects such as infrastructure configuration may require significant effort.

Additionally, a certain set of measures results in a level of service without an indication on how to improve the result. Decreasing the number of necessary simulation runs by means of problem-specific knowledge is plausible for this first AIP. For example, trains are expected to run in larger batches with increasing running times. Cancellation of a train is preferably considered at the start of a batch rather than near its end. Despite this knowledge, differing timetable structures as well as exact cancellation decisions significantly impact on the optimal batch sizes, rendering it impossible to define the best possible set on beforehand.

Imagine a disruption duration during which m and n trains run in directions 0 and 1 respectively, with the order between trains in the same direction fixed. Deciding on the remaining orders between trains, results in $m \times n$ possibilities. Additionally, cancellation is possible for each of the $(m+n)$ trains, resulting in $mn(m+n)$ possible results. Evaluation of one scenario by a heuristic required around 0.1 s, or about 20 min for all 11,664 possibilities in case of a balanced timetable with $m = n = 18$. Restrictions such as "train x can have at most y min of delay" may reduce the number of possible solutions, but this very much depends on the timetable structure.

Table 5.1: Overview of parameters and measures considered to be of major impact on the operations during a disruption on the first (and following) AIP(s).

Input parameter	Variable
Length of segment D	Train re-timing
Presence of a stop (second AIP)	Train ordering
Number of tracks	Track assignment (third AIP)
Timetable	Stop-skipping (second AIP)
Minimum buffer times	Global rerouting
Running time over block sections	Train cancellation
(Relative) passenger numbers	
Disruption duration / scheduling horizon	
Allowed speeds (train and infra)	

For the second and third AIPs, additional measures come into play, again increasing the number of possibilities.

An optimization model gives more flexibility with regard to input parameters. Varying those in Table 5.1, comes down to using a preprocessing step to define the model's input. Additionally, the set of measures is represented by the decision variables and does not have to be predefined. The resulting measures will be optimal with regard to a certain objective function, and only one run is required.

On the other hand, incorporating a high level of detail results in complex models which are hard to solve to optimality within a reasonable computation time, as shown in Section 4.4. As a result, a mismatch may exist between the model and real-life operation due to assumptions made to decrease the problem size. Delays are not taken into account as a stochastic process, only as a fixed value about which perfect knowledge is assumed. Finally, optimization models operate as a black box and may result in unexpected observations, which could be hard to explain due to complex interactions.

5.3.2 Choice of solution approach

The goal of the thesis is twofold. One, the practical part consists of presenting a method that can be used by dispatchers to derive disruption timetables at runtime in case of a disruption on an AIP. Two, a theoretical analysis on the impact of different input parameters for a certain disruption on the optimal set of measures, is conducted.

For the practical part, the high level of detail of a simulation tool is not required as operations are still expected to deviate from schedule. Better is to provide a rough sketch of the best solution, after which dispatchers, or a TMS, can still adjust the result to the prevailing situation. Concerning the theoretical analysis, running a high number of scenarios with different input parameters is desirable. Additionally, the short time to react in case of a disruption, limits the time available to test multiple sets of measures. A mathematical model can quickly find the optimal or at least a good solution. Moreover, it provides more flexibility in measures, abolishing the need for a predefined set. Hence, the choice for a mathematical model can be justified.

Nevertheless, the simulation model by Van Thielen et al. [56] could be useful to evaluate both the conflict-freeness of the generated disruption timetable, and its effects on traffic outside of the corridor. However, different input data is used, which would hamper the interpretation of results. This approach is left for future research.

5.4 Model input, framework and output

A dispatcher has to provide the model with data about, amongst others, the disruption, timetable and scheduling horizon. Additionally, service requirements such as a maximum delay for certain trains, may have to be satisfied and several other input parameters are needed. These will be discussed when appropriate. Enabling the optimization model to generate a disruption timetable requires several pre-processing steps. Figure 5.3 displays the general flow of information in the framework.

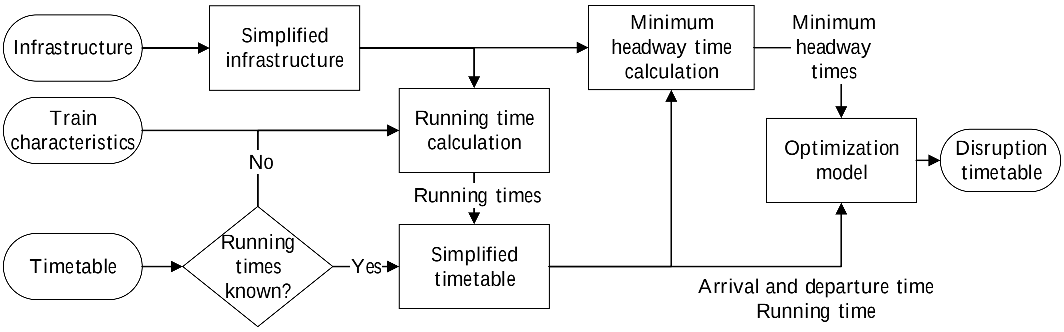


Figure 5.3: Overview of inputs and required pre-processing steps before building the optimization model, which results in a disruption timetable.

Based on the real-life infrastructure, a simplified version can be built, representing the situation. Ideally, timetable data contains information about the minimum running time over each block section, e.g. by defining the time a train passes each signal. Whereas Infrabel has this data at hand, publicly available information such as line folders [38] only mention the arrival and departure times in stations. Within this thesis, running times have to be calculated based on the simplified infrastructure, train type characteristics and a number of assumptions. A simplified timetable is built, characterizing trains by their entry time, the direction of movement and their running times. Next, blocking time stairways are constructed to derive minimum headway times. Finally, the optimization model computes the new disruption timetable. In real-life application by Infrabel, all required input information on timetable and infrastructure is at hand, abolishing the need for the above mentioned simplifications.

The output consists of a disruption timetable with a number of train cancellations and re-timings, visualized by means of time-distance diagrams. Appendix A explains how those shown in this thesis, should be interpret. Excel sheets are used to provide all necessary input concerning infrastructure and timetable, next to a large number of parameters. All pre-processing and post-processing steps are carried out in Matlab [35]. The YALMIP toolbox [31] passes the optimization model on to CPLEX [22].

5.4.1 Deriving input for the Tienen-Landen case study

For the Tienen-Landen case study, information on the infrastructure was obtained from so-called *Schematic Signalling Plans* (SSP), which contain detailed information amongst which signal location, aspects that can be shown, and direction it is used for. Figure 5.4 shows a potential lay-out of a double-track segment with two block sections. Although signals are present to show aspects in both directions, they are not located at exactly the same position. As a result, the length l_b^{dir} of the b^{th} block section may differ per direction $dir \in \{0, 1\}$. In most circumstances, differences in lengths l_b^1 and l_b^0 were smaller than 100 m. For running and blocking time calculations within this thesis, they are considered to be equal in length. Nevertheless, the developed model is still able to handle differences as these only affect (known) input information such as the timetable.

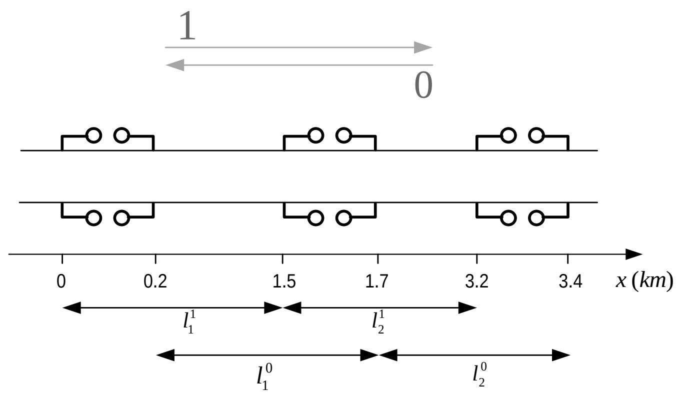


Figure 5.4: Real-life configuration of a railway line with signals in both directions located at distinct points. However, based on the limited distance between them, this configuration can be modeled as the general AIP presented in Figure 5.1.

Figure 5.5 shows the resulting infrastructure model for the Tienen-Landen case and presents information on speed restrictions. Segment A is not incorporated as trains leaving Tienen in direction 1 immediately meet the switch S_1. Segments D and B hold two and three block sections respectively, each with an average length of 1,600 m. The block section of switch S_1 is shorter than the one for S_2.

The disruption starts at 5:54[2], at the start of morning peak, and dispatchers want to reschedule traffic over a 3 h horizon. Unfortunately, detailed timetable data was not at hand and public sources had to be used. The specific trains running during this scheduling horizon were taken from the line folder of line 36 [38], resulting in 25 trains, respectively 10 and 15 for directions 1 and 0. Next to IC and L trains, several peak-hour trains towards Brussels were present, considered to have rolling stock

[2]This was the first departure time of a train on the corridor, closest to 6:00.

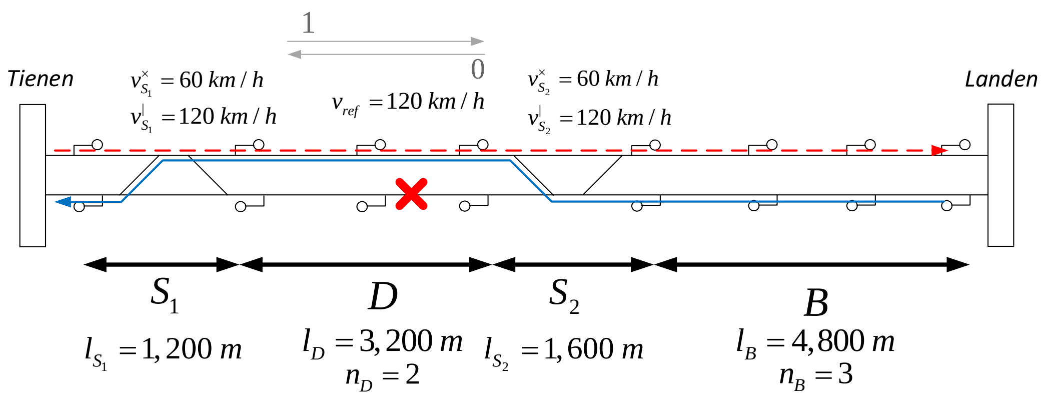


Figure 5.5: Schematic overview of infrastructure in the Tienen-Landen case study, based on simplifications of the SSPs for line 36 [25], providing information on allowed speed and visualizing train routes during a disruption on the lower track.

characteristics of L trains within the model. Departure times from stations Tienen and Landen were used as entry times for trains in directions 1 and 0 respectively. Appendix F presents a complete overview of the timetable data.

5.5 Optimization model description

This section provides a step-by-step description of the optimization model, its constraints and the objective function. First, Section 5.5.1 describes how the problem can be modelled as an extended single-machine scheduling problem, which defines the required input information. Therefore, Section 5.5.2 describes the last pre-processing steps to calculate running and headway times, and performance indicators to evaluate the solutions obtained. Next, the essential constraints and objective function for the core model are formulated in Section 5.5.3. Solving the Tienen-Landen case serves as an illustration here. Additionally, dispatchers may want to include various real-life constraints which are not essential to get a solution, but are expected to provide a better service. Section 5.5.5 elaborates on such possible *service constraints* and illustrates their effects on the Tienen-Landen case study.

5.5.1 As a single-machine scheduling problem

Figure 5.6 illustrates how the problem can be considered as a machine scheduling problem, with the upper track of the middle segment being out of operation. Assume that both segments A and B in Figure 5.1 consist of a number of block sections, allowing to buffer trains. No information on the stations shown in Figure 5.1 is incorporated in the model: they can hold any number of platform tracks. Hence, at each side of the disrupted area, infinite buffer capacity is assumed: trains waiting to enter the disrupted area can be stored either on the block sections in segments A or B, or at stations.

The segment D on which the disruption occurs constitutes together with both switches the disrupted area, which is considered as a single machine M^D and receives jobs, i.e. trains, from two directions. As M^D consists of multiple block sections, multiple trains running in the same direction may be processed at the same time.

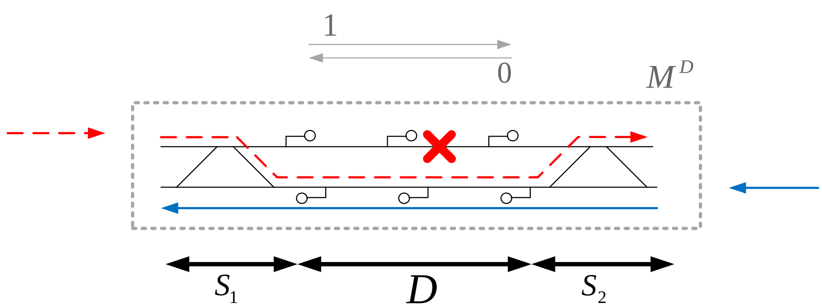


Figure 5.6: Interpretation of a disruption on the first AIP as a single-machine scheduling problem, with trains arriving at machine M^D from both directions.

Some similarities exist with the machine scheduling problem with batch operations. Hence, a group of trains with the same direction running after each other on D, is referred to as a *batch*. This possibility of having multiple trains of the same direction within D, has to be taken into account when determining the set-up times.

The complete set of trains T is divided in two subsets T_0 and T_1, depending on their direction $dir \in \{0, 1\}$. Transforming the problem to a single-machine scheduling problem, results in the following variables and parameters:

- r_t Release time of train t, its arrival time at switch S_1 ($dir = 1$) or S_2 ($dir = 0$).
- d_t Due date of train t, i.e. time at which the job should be finished the latest.
- t_t Time a which a train t is allowed onto the disrupted area, its starting time.
- p_t Process time of train t, corresponds to the train's minimum running time over the complete disrupted area.
- C_t Completion time of train t, i.e. departure time from the disrupted area.
- D_t Delay of train t upon leaving the disrupted area.
- s_{ij} Minimum headway time required after the exit of train i depending on which train j follows, resulting in a sequence-dependent set-up time.

Following the blocking constraints in the alternative graph approach of D'Ariano et al. [16] and Mascis and Pacciarelli [34], these set-up times s_{ij} are enforced between the completion time of train i and the starting time of train j.

Decision variables

Dispatchers can take several measures: train re-timing, reordering, cancellation and rerouting. Note that train re-timing is mostly the result of an interaction between all other measures and the release time: the order of train t with respect to other trains defines when it can enter the earliest, but it cannot before its release time r_t. Hence, three main decision variables come into play.

First of all, binary decision variables $q_{ij} \in Q$ indicate whether train $i \in T$ is scheduled before train $j \in T$ ($q_{ij} = 1$), or after it ($q_{ij} = 0$).[3] The size of Q can be reduced using symmetry, i.e. $q_{ij} = 1 - q_{ji}$, resulting in $|Q| = |T| \times (|T| - 1)$. Unless mentioned differently, the order of trains at the entrances of the corridor is assumed to be fixed. In those cases, the decision reduces to determining from which direction the next train can enter M^D.

If scheduling all trains is not possible, or not beneficial, train cancellation can be considered. In case a different route over the network is plausible, global rerouting can be considered, which may turn out better for passengers. Both decisions are represented by binary variables x_t and dev_t for each train $t \in T$. When a train t is either cancelled ($x_t = 1$) or rerouted ($dev_t = 1$), it does not appear on the sub-network considered in the problem: the job is rejected by machine M^D. These

[3] In line with the Manne concept [33], $q_{ij} = 1$ does not require train j to follow right after train i: other ones may be processed in-between.

trains are moved to the start of the timetable, i.e. their arrival and completion time on segment D are set to 0. In reality, a cancelled or deviated train results in passengers arriving at their destination with a delay. However, passengers attach a higher value of time to waiting, e.g. a factor 2. Hence, relative penalties should represent the nuisance as perceived by passengers.

Decision are taken only for trains within the scheduling horizon and assuming perfect knowledge on all parameters. However, in real-life operations, trains may enter the disrupted area with a delay. The model can use the current (prediction of) delays as release times. Nonetheless, estimates of both disruption duration and delays of incoming trains may get updated along time. Hence, it is advisable to re-run the model at fixed intervals using the updated information. Models developed in this thesis could be embedded in such a rolling horizon approach (see Section 4.4.5).

5.5.2 Determination of process and set-up times

Additional pre-processing steps are needed to derive the process and set-up times from the result in Section 5.4. In case running times are not provided as input, e.g. due to data unavailability, certain assumptions can be made to construct the speed profile of a train. Based on these running times, blocking time stairways are constructed to determine the minimum headway and set-up times.

Running time calculation

In the absence of block section-specific running times, these are determined by generating a speed profile along the complete corridor. Each train t has a maximum speed v_{tb} for each block section $b \in \beta$, which is the minimum of its own maximum allowable speed and the reference speed either during regular (v_{max}) or disrupted (v_D) operations. Note that trains crossing a switch, e.g. S_1, do this at a lower speed $v^{\times}_{S_1}$ than running straight over it $v^{|}_{S_1}$. After a train entered a block section with a higher v_{tb} than the previous one $v_{t(b-1)}$, it increases speed at its maximum acceleration rate a_t. On the other hand, when the next block section has a lower speed $v_{t(b+1)} \leq v_{tb}$, the train driver reduces speed upon entering block section b at a braking rate b_t.

Figure 5.7 shows the resulting speed profiles for trains following different routes over the disrupted area. The black lines indicate speed limits for each block section. In case a train has to cross the switches S_1 and S_2, it lowers its speed to the maximum allowable crossing speed $v^{\times}_{S_1}$. Both route and speed profile are represented by red dashed lines. Trains running straight over the switches (full blue lines) maintain a higher speed for a longer time. These speed profiles are used to calculate running times R_{tb} of train t for each block section b. Recall that two service types are distinguished: IC and L trains. Acceleration a_t and braking rates b_t for both train types are set to 0.6 and 0.375 m/s^2 respectively [8]. Maximum train speed v_t^{max} equals 140 and 120 km/h for IC and L trains respectively, unless stated otherwise.

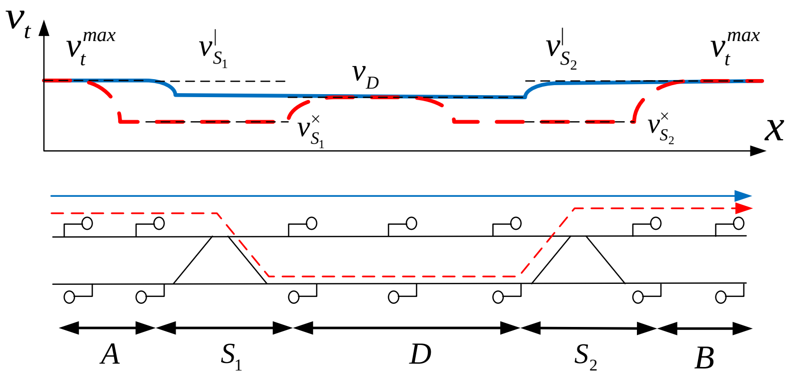


Figure 5.7: Speed profiles for trains following different routes over the disrupted area. Speed restrictions have to be respected as imposed by the signalling system (Section 2.2.1). Trains running straight over the switches (full blue lines) maintain a higher speed for a longer time than those crossing them (dashed red lines).

The resulting running times represent technical minimum running times. It is common practice to add a certain percentage to these values to account for small variations in driver behaviour, mostly a 5% supplement [43]. The resulting running times are rounded to seconds.

Based on the entry time of a train t and the calculated running times for each block section, the scheduled arrival and departure times for the block sections can be determined for both regular and disrupted operations. This results in the train paths as shown in a time-distance diagram, and allows the construction of blocking time stairways.

Set-up time calculation based on blocking time theory

Train separation is modelled between the exit of the first train i from the disrupted area (machine M^D) and the entry of its successor j. Set-up times are determined using blocking time theory (see Section 2.2.2):

$$s_{ij} = t^{clearing} + t^{release} + t^{clear\ signal} + t^{sight} + t_j^{approach} \qquad \forall i, j \in T \qquad (5.1)$$

Equation (5.1) indicates that train i occupies the last block section of M^D for a time $t^{clearing} + t^{release}$ after its exit. The block section has to be reserved $t_j^{approach}$ before train j physically enters, plus a margin $t^{clear\ signal} + t^{sight}$. Although clearing times depend on train speed and length, a fixed value of 12 s is used for $t^{clearing}$ and is assumed to be sufficient in any situation. Additionally, $t^{release}$ and $t^{clear\ signal}$ depend on technical features and are train-independent, requiring 9 and 6 s respectively. Sight and reaction time is assumed to be 9 s [51]. This method holds for trains running opposite directions.

Modelling the disrupted area as a single machine, complicates the determination of minimum set-up time between trains moving in the same direction. Denote the block sections of M^D with $b \in \beta_D$, and the start and end of train i's blocking time on b as respectively $\tau^s_{i,b}$ and $\tau^e_{i,b}$. Using the release times r_i and r_j, set-up times are calculated as:

$$s_{ij} = \max_{b \in \beta_D} \left\{ \tau^e_{i,b} - \tau^s_{j,b} \right\} + r_j - r_i - \sum_{b \in \beta_D} 1.05 R_{ib} \qquad \forall i, j \in T \tag{5.2}$$

Figure 5.8 illustrates the reasoning behind Equation (5.2). First, let both trains start at the same time, $r_i = r_j$ (Figure 5.8a). Next, move train j until no overlap exists for any of the block sections of machine M^D, resulting in a minimum headway time h_{ij} (Figure 5.8b). Subtracting the total running time of train i over M^D from this headway time h_{ij}, results in the set-up time s_{ij}.

The approach time for M^D $\left(t_j^{approach}\right)$ remains to be determined. In a 2-block-3-aspect signalling system, this time equals the running time of train j over the block section preceding the entry on the disrupted area, e.g. the last block section of segment A for trains in direction 1. However, in absence of segments A and/or B, it is assumed train j has to depart from standstill and enters the disrupted area when it reaches the maximum speed.

Performance indicators

Measures taken during the disruption affect passengers in various ways. Train cancellation forces them to take a later one and results in arriving too late at their final destination. Similar arguments go up for train rerouting: in general, the alternative route has a longer travel time. Next to the time-distance diagrams, results will report the number of cancelled and rerouted trains.

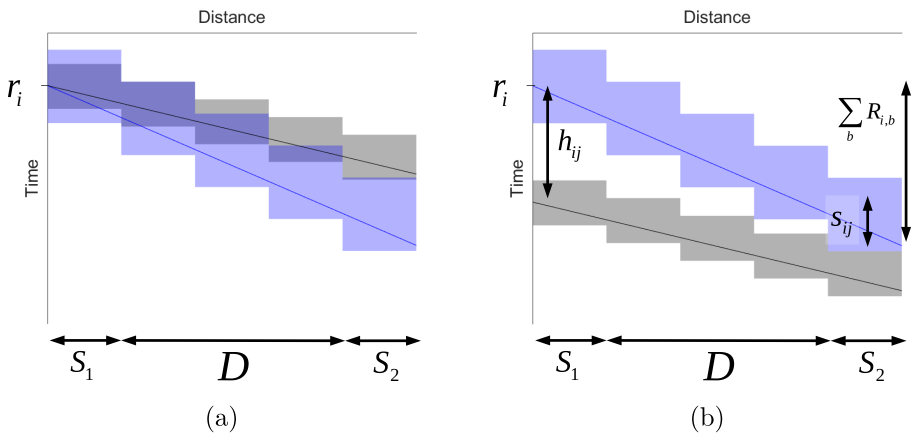


Figure 5.8: Calculation of the minimum set-up time when train j follows train i (s_{ij}) on the disrupted area. (a) Starting from overlapping release times, moving the blocking time stairways of train j until no overlap exists in (b), results in the minimum headway time h_{ij}. The set-up time is determined by subtracting the total running time of train i from this value.

Determining the order between all train pairs, fixes the trains' entry and exit times. They may experience a delay compared to the original timetable, because of entering the disrupted area later than their release time, and/or increased running times. Delay statistics are determined relative to the original departure time. Maximum, total and average delay may be mentioned, where appropriate also distinguishing between trains running in different directions.

For the calculation of the delay D_t within the model, the due date or departure time from the disrupted area d_t is needed. Question is which value to use: the original departure time (as for the statistics), or the earliest possible one under the disruption speed v_D. In the first case, D_t would depend on the original speed of the train, as v_D incurs a higher additional running time for faster trains. Hence, d_t represents the *adapted departure time*, taking into account the increased running times during the disruption.

5.5.3 Essential constraints and objective function

This section formulates the constraints and objective function for the single-machine scheduling problem, using the concepts introduced in the previous sections. Next, the model is applied to the Tienen-Landen case study (Section 5.5.4).

Release time and process times

In a machine scheduling problem, jobs cannot be started before their release time. Here, the starting time of train t on the disrupted area, cannot be earlier than its scheduled arrival time at either S_1 or S_2, representing its release time r_t. After this, processing of train t can start:

$$t_t \geq r_t(1 - x_t - dev_t) \qquad \forall t \in T \qquad (5.3)$$

$$C_t \geq t_t + p_t(1 - x_t - dev_t) \qquad \forall t \in T \qquad (5.4)$$

Constraints (5.3) ensure this minimum release time is respected, unless train t is cancelled ($x_t = 1$) or deviated ($dev_t = 1$). Constraints (5.4) link the starting time t_t of train train t, with its completion time C_t, which denotes its exit from the disrupted area. The required process time p_t equals the train's total minimum running time over the disrupted area (machine M^D).

The associated completion time C_t is used to calculate the exit delay D_t as follows:

$$D_t \geq C_t - d_t(1 - x_t - dev_t) \qquad \forall t \in T \qquad (5.5)$$

$$D_t \geq 0 \qquad \forall t \in T \qquad (5.6)$$

Constraints (5.5) calculate the delay, which cannot be negative (Constraints (5.6)). Following the meaning of d_t as the earliest departure time from the disrupted area, this last set of constraints is redundant for this model. However, they become important for the second and third AIP (Chapters 7 and 8).

Enforcing minimum set-up times

For safety reasons, a train cannot enter a block section before a minimum time has elapsed after the entry of its predecessor, i.e. the headway time. Section 5.5.2 described how the set-up time s_{ij} for train j following train i on machine M^D can be determined. Set-up time constraints are added between all pairs of trains as follows:

$$t_j \geq C_i + s_{ij} - M(1 - q_{ij} + x_i + x_j + dev_i + dev_j) \qquad \forall i, j \in T \tag{5.7}$$

$$t_i \geq C_j + s_{ji} - M(q_{ij} + x_i + x_j + dev_i + dev_j) \qquad \forall i, j \in T \tag{5.8}$$

Constraints (5.7) and (5.8) are disjunctive. Big-M M represents a "large" constant, set to twice the disruption duration in seconds[4]. If train j enters machine M^D later than train i, i.e. $q_{ij} = 1$, Constraint (5.7) reduces to $t_j \geq C_i + s_{ij}$ and Constraint (5.8) is relaxed. Both constraints are relaxed if at least one of both trains involved is cancelled or rerouted.

Objective function

Without an objective function, cancelling all trains is a viable option. Regarding passenger service, this is unacceptable. Therefore, the goal is to minimize the effects on the passengers, by balancing between cancelling and delaying trains, and global rerouting when allowed. As for the machine scheduling problem with rejection, a weighted objective function is adopted, penalizing possible measures:

$$\min \sum_{t \in T} w_t^{cancel} x_t + \sum_{t \in T} w_t^{deviation} dev_t + \sum_{t \in T} w_t^{delay} D_t \tag{5.9}$$

Objective function (5.9) balances between all possible measures, which may have train-dependent penalties. Cancellation is balanced against rerouting and delaying trains, the latter being a result of train orders rather than a decision variable in itself.

Determining weights w_t^{cancel}, $w_t^{deviation}$ and w_t^{delay} requires special attention, as these strongly affect the final solution. With this basic model, w_t^{cancel} has an intuitive interpretation: if cancelling train t reduces the total delay more than w_t^{cancel}, it is better to cancel train t. This reduction in total delay is referred to as train t's *induced delay*. The penalty for rerouting $\left(w_t^{deviation}\right)$ should be at least the additional journey time along the alternative route (see Section 5.5.5), and has a similar interpretation as w_t^{cancel}. Additionally, train rerouting is only beneficial if $w_t^{deviation} \leq w_t^{cancel}$. Finally, delays will be measured up to a 1 second basis and $w_t^{delay} = 1$.

In the remainder, penalties are assumed to be train independent and rerouting is not available, unless stated otherwise.

5.5.4 Application to the Tienen-Landen case study

The model consisting of Constraints (5.3 - 5.8) and objective function (5.9), was applied to Tienen-Landen case study with a train breakdown on the lower track

[4]Increasing M may lead to computation time increases. However, experiments have shown that smaller values may unwillingly affect the solution.

around 6:00, as indicated in Figure 5.5. On top of input data described in Section 5.4, the dispatcher imposed a disruption speed v_D and wanted to receive a disruption timetable over a scheduling horizon of 3 h. Without loss of generality, trains were expected to arrive on time. If this was not the case, the model would have collected the (predicted) arrival times as input.

Table 5.2 provides an overview of all scenarios discussed here. First, the specific parameters are mentioned, followed by statistics on delays and number of cancelled trains. Both the objective function value and computation time are provided. The last columns present the absolute objective function value obtained with a FIFO heuristic (presented in Section 5.6.1) and its relative increase compared to the model's result. Each scenario is discussed in detail below.

Scenario 1 imposed a disruption speed of $v_D = 80\ km/h$ and penalizes train cancellation with $w^{cancel} = 3,600$, resulting in a total delay of about 2 h (7,792 s). The time-distance diagram in Figure 5.9 shows trains in the same direction formed batches to reduce the total delay. As the decrease in speed over segment D and associated running time increases were rather limited, all trains could still be scheduled.

No clear conclusion on the optimal batch size could be drawn for two reasons. First of all, more trains run in direction 0 than in direction 1: using the same batch size would be disadvantageous for direction 0. Secondly, release times are not uniformly distributed over the scheduling horizon, especially for direction 0. Hence, the timetable structure is of significant importance and may result in the FIFO approach obtaining the same results.

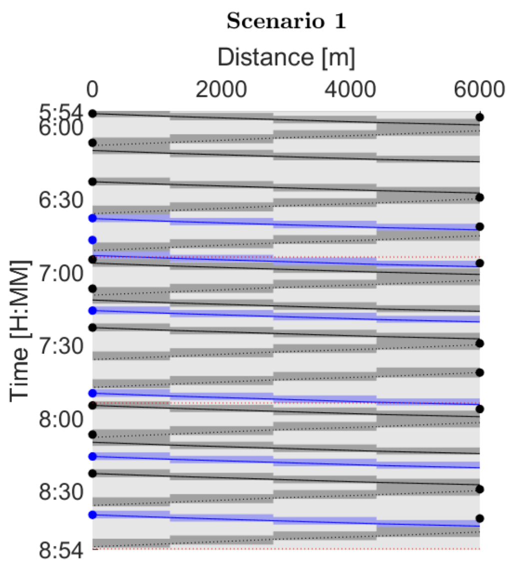


Figure 5.9: Time-distance diagram for scenario 1 of the Tienen-Landen case study.

Table 5.2: Description of each scenario for the Tienen-Landen case study, presenting information on the set-up, i.e. disruption speed v_D, cancellation penalty w^{cancel} and other parameters specific to the scenario. Solution statistics report the maximum, total and average delay values as described in Section 5.5.2, together with the number of cancelled trains, constituting the weighted objective function. Finally, absolute objective function values for results obtained by the FIFO approach (presented in Section 5.6.1) are given together with the relative increase to the model's objective function value.

#	Disruption characteristics			Max. delay (s)	Total delay (s)	Average delay (s)	Number cancelled	Objective function value	CPU time (s)	FIFO	
	v_D (km/h)	w^{cancel}	Other parameters							Absolute value	Relative increase
1	80	3,600		773	7,792	312	0	4,242	0.4	4,242	0%
2	40	3,600		2,041	18,225	792	2	17,419	6.6	38,295	120%
3	40	7,200		2,090	29,194	1,168	0	20,604	4.2	38,295	86%
4	40	3,600	$D_t^{max} = 1,800$	2,041	18,225	792	2	17,419	3.5	38,295	120%
5	40	7,200	$D_t^{max} = 1,800$	2,090	29,194	1,168	0	20,604	0.7	38,295	86%
6	40	3,600	$N_{batch}^{max} = 5$	1,485	14,950	680	3	18,208	23.8	-	-
7	40	3,600	$N_{batch}^{max} = 3$	2,257	19,669	894	3	23,013	120.1	-	-
8	40	3,600	$OB = 1$	2,041	18,494	804	2	17,774	5.9	38,295	115%
9	40	3,600	$N_{1,LOS}^{min} = 3$	2,041	18,494	804	2	17,774	12.7	38,295	115%
10	40	3,600	$\frac{pax_0}{pax_1} = 10$	2,161	12,202	642	6	33,822	0.9	233,496	590%
11	80	3,600	$v_L = 50\ km/h$ Fixed order	895	7,433	310	0	5,030	0.4	10,938	117%
12	80	3,600	$v_L = 50\ km/h$ relaxed order	895	7,334	306	0	4,931	0.4	10,938	122%
13	80	3,600	$v_L = 50\ km/h$, Relaxed order	895	7,334	306	0	4,931	0.4	10,938	122%
14	40	3,600	$\delta^{swap} = 1,800\ s$ $w^{deviation} = 2,700$ $N_{dev,1}^{max} = 1$ $N_{dev,0}^{max} = 0$	1,485	14,767	671	1	16,053	15.7	38,295	139%

To investigate the cancellation penalty impact, scenarios 2 and 3 imposed a lower disruption speed ($v_D = 40\ km/h$) with w^{cancel} set to 3,600 and 7,200 respectively. Table 5.2 reports significant increases in total delay compared to scenario 1: respectively 310% and 385% for scenarios 2 and 3. The former managed to mitigate this increase by cancelling two trains in the less crowded direction 1. As time-distance diagrams for scenarios 2 and 3 (Figure 5.10, left and right) show, the sole difference was the insertion of a cancelled train (full red arrow). For scenario 2, cancellation resulted in a small buffer within one of the subsequent batches (dotted orange lines), avoiding the transfer of delays which arose earlier in the timetable: $D_t = 0\ s$ for the first train after this buffer. Scenario 3 did not result in train cancellations, meaning their induced delay was in-between 3,600 and 7,200 s. Both problems require longer computation times, 6 and 4 s respectively, explained by cancellation of trains becoming a viable option, interacting with a large number of order variables.

Regardless of cancellations, both scenarios resulted in larger train batches, which, as expected, were larger in the more busy direction 0. Hence, a number of trains started queuing at either end of the disrupted area while processing trains in the other direction. For example, the batch in the orange dashed box in scenario 3 (Figure 5.10, right) avoided trains running in direction 1 after 6:15 for up to one hour. Despite the higher passenger number for direction 0, it is questioned whether dispatchers (and passengers) would accept this solution. Therefore, Section 5.5.5 presents several ways to influence the results before running the model.

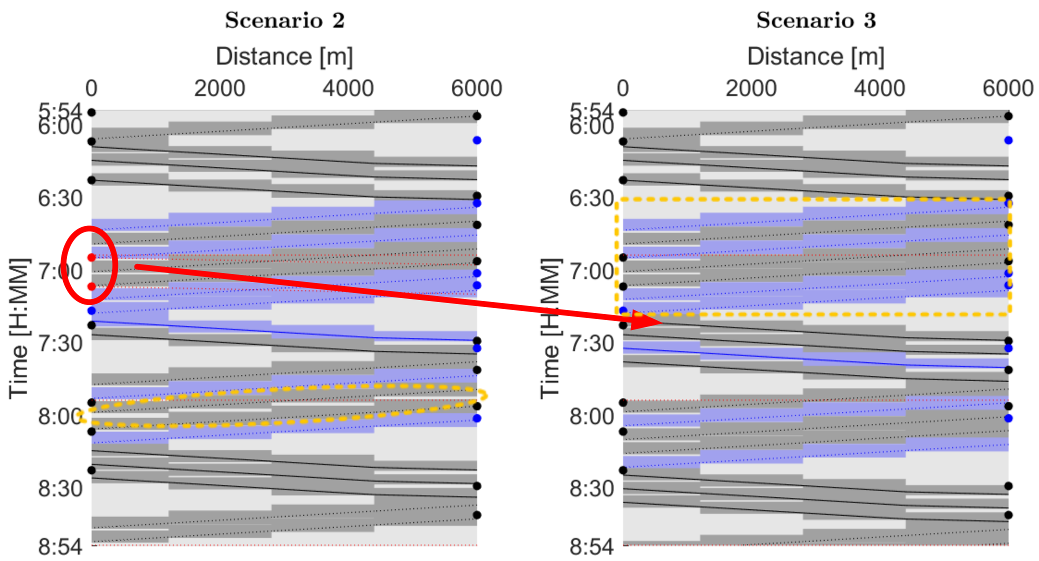


Figure 5.10: Time-distance diagrams for scenarios 2 (left) and 3 (right) of the Tienen-Landen case study.

Finally, average delay for both directions did not differ too much in scenario 3: 19 min 28 s and 21 min 26 s for directions 0 and 1 respectively. Each delay, regardless of the direction, contributed to an equal extent to the objective function value, explaining this observation. As it may be beneficial to reduce delays in the more crowded direction with higher passenger numbers, Section 5.5.5 describes a method to incorporate these into the objective function.

5.5.5 Service constraints

The model as presented in Section 5.5.3 contains all essential aspects to come up with a solution for the problem. However, this solution may not always be acceptable from an operations point-of-view. Without going into detail what is considered to be an "acceptable" solution, this section presents several ways to influence the final result.

Maximum delay

When using high cancellation penalties w_t^{cancel}, a single train may get a very long delay. Dispatchers may impose maximum delay constraints to guarantee, for example, passenger connections or prioritize specific trains. The benefit of imposing such constraints may be twofold. Primarily, it limits delays for a single train. To avoid cancellation of that train, the model can decide to redistribute the delay over all other trains by altering batch sizes. Additionally, the reduction in solution space may result in shorter computation times. However, computation times could possibly increase since it may hamper finding a feasible and/or optimal solution.

$$D_t \leq D_t^{max} \qquad \forall t \in T \tag{5.10}$$

Constraints (5.10) limit the delay of train t by D_t^{max}, representing the maximum acceptable delay. Imposing severe restrictions in the absence of train cancellation may however lead to infeasibility.

Following the maximum delay of scenario 1 reported in Table 5.2, imposing a maximum delay would affect the solution only for D_t^{max} at most around 10 min. As this is very restrictive, scenarios 2 and 3 are evaluated instead with $D_t^{max} = 1,800\ s$, resulting in scenarios 4 and 5 respectively. As maximum delay (Section 5.5.2) and D_t are determined relative to respectively original and adapted departure times, a "mismatch" may exist. Table 5.2 does not report changes compared to scenarios 2 and 3 as the adapted D_t was smaller than $D_t^{max} = 1,800\ s$. The sole effect is a strong decrease in computation time for both scenarios 4 and 5. However, it is not possible to predict on beforehand which values of D_t^{max} lead to such results.

Maximum train batch size

Optimally exploiting capacity, may lead to large batches as illustrated by scenarios 2 and 3 (Figure 5.10). Limiting the number of trains after each other in one direction, i.e. batch size, may be an acceptable constraint if objective function value increases remain limited.

Incorporating limits on batch size, N_{batch}^{max}, in the model, can be done by exploiting order variables q_{ij}. Recall that $q_{ij} = 1$ if train j is scheduled after train i. Appendix C describes the complete reasoning behind the following set of constraints:

$$q_{ij} \geq x_i + dev_i \qquad \forall i,j \in T, i > j \tag{5.11}$$

$$1 - q_{ij} \geq (x_j + dev_j) - (x_i + dev_i) \qquad \forall i,j \in T, i > j \tag{5.12}$$

$$\sum_{k \in T_1} q_{kj} - \sum_{k \in T_1} q_{ki} \leq \left(n_j - n_i - X_i^j\right) N_{batch}^{max} + M(x_i + dev_i + x_j + dev_j) \qquad \forall i,j \in T_0, n_i < n_j \tag{5.13}$$

$$\sum_{k \in T_1} q_{kj} - \sum_{k \in T_1} (x_k + dev_k) \leq \left(n_j - \sum_{n_k < n_j} (x_k + dev_k)\right) N_{batch}^{max} + M(x_j + dev_j) \qquad \forall j \in T_0 \tag{5.14}$$

Constraints (5.11) and (5.12) fix order variables q_{ij} if trains i and/or j are cancelled and moved to the start of the timetable. In Constraints (5.13) and (5.14), n_i and n_j represent the ordinal numbers of trains i and j in direction 0. Cancellation term $X_i^j = \sum_{k: n_i < n_k < n_j \in T_0} (x_k + dev_k)$ groups the cancellation variables of all trains running in-between trains i and j in the same direction. The left-hand side terms of Constraints (5.13) count the number of trains in the opposite direction operated before respectively trains j and i, regardless of whether they are consecutive or not. Hence, Constraints (5.13) and (5.14) impose strict limitations on batch size respectively in-between a pair of trains, and the first trains running in direction 0. These last two constraints can be generated for the other direction by reversing indices 0 and 1.

Scenarios 6 and 7 limit the batch size of scenario 2, i.e. $v_D = 40\ km/h$ and $w^{cancel} = 3,600$, to five and three trains respectively. Observing scenario 2's time-distance diagram (Figure 5.10, left), one expects only batches in direction 0 get affected for scenario 6, whereas scenario 7 also influences those in the other direction. Table 5.2 reports results for both scenarios 6 and 7, with respectively a 5 and 13% increase in objective function value. Following timetable structure, especially trains in direction 0 got larger delays and some were cancelled. As opposed to imposing a maximum delay, which reduced solution space and decreased computation times, limiting the batch size rendered the problem more difficult to solve because of these cancellation decisions. Figure 5.11 shows the resulting time-distance diagrams for scenarios 6 (left) and 7 (right), illustrating the effectiveness of Constraints (5.13) and (5.14) in limiting batch size.

Comparison between scenarios 2 (Figure 5.10, left) and 6, confirmed the expectation that only batches in direction 0 got affected. These batches had the maximum size, except for the first and last one, as not enough trains were present to “fill” them. Redistributing trains over the batches is not sufficient to avoid an additional cancellation. Following the cancellation of the first train in direction 0, the early

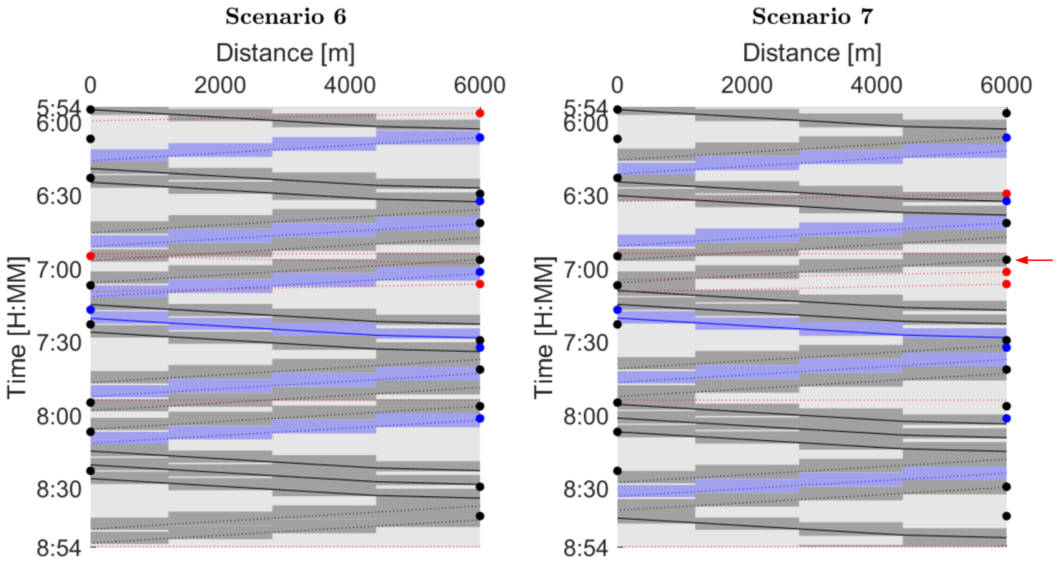


Figure 5.11: Time-distance diagrams for scenarios 6 (left) and 7 (right) of the Tienen-Landen case study.

batches in direction 1 got reassembled. Although the large waiting time between trains in that direction was the main argument, it did not decrease sufficiently under these service constraints. Regardless of these major changes, the objective function value increased with only 5%.

Results for scenario 7 were in line with these observations: as maximum batch size further decreased, trains in direction 0 got redistributed over the batches. Three trains in direction 0 got cancelled, which is not surprising as the redistribution may move subsequent trains to later batches, incurring higher delays. Cancellation of trains in direction 0 reduced the induced delay of the previously cancelled train in direction 1 below w^{cancel}. Batches in both directions had the maximum batch size if a sufficient number of trains were queuing at either end of the disrupted area. However, if a train arrives still before the end of the batch, it often was added to the batch as for the indicated train in Figure 5.11 (right).

For both scenarios 6 and 7, the maximum delayed train is one in direction 1 instead of the more busy direction 0. Although it may seem unexpected, it is caused by trains with high delays in direction 0 being cancelled instead of planned. Planning one of these would mean delaying multiple trains in direction 0 and reassigning trains to batches. For direction 1, the latter decision did not come into play, as it had less trains running.

To conclude, enforcing a N_{batch}^{max} is expected to limit the time between batches in one direction, but may lead to limited improvements at the expense of efficiency losses. Timetable structure is of major importance here, and predicting which trains should be cancelled is rather difficult. Hence, limiting the maximum delay of trains, and possibly avoiding their cancellation, seems to be a more effective way to tackle the issue of long waiting times.

Influencing the number of train cancellations

In case of a high number of trains in (at least) one direction, cancelling all trains from the other one, could be beneficial in terms of the objective function (5.9). Therefore, dispatchers may want to have an influence on train cancellations. For example, another way to manipulate the result of scenario 2 is forcing more trains to run in direction 1 between 6:15 and 7:15 by hard constraints.

Two approaches are presented. The first one balances the number of cancellations in each direction over intervals of 1 h, whereas a second approach enforces a *minimum level of service* in terms of number of trains running per hour per direction. Both have their pros and contras depending on the situation, especially timetable unbalances.

Balancing constraints Balancing the number of train cancellations, results in passengers being more evenly affected by the measures regardless of the direction of their journey. Louwerse and Huisman [32] use a similar concept to avoid rolling stock unbalances at both ends of the corridor, with the aim to avoid additional cancellations due to rolling stock shortages. Let OB represent the maximum off-balance in number of train cancellations between directions per hour. For each train in one of both directions, the number of cancellations in the following hour have to be balanced. This is considered by looking at trains with an original departure time, i.e. release times r_t, in the following 1 h-interval:

$$\left| \sum_{\tau \in T_{0,[r_t, r_t+H[}} x_\tau - \sum_{\tau \in T_{1,[r_t, r_t+H[}} x_\tau \right| \leq OB \qquad \forall t \in T \tag{5.15}$$

Constraints (5.15) express that the absolute difference between the number of cancellations in both directions has to be smaller than a maximum value OB. For each train t, all trains τ in both directions within the next hour, i.e. $r_\tau \in [r_t, r_t + H[$ with $H = 3,600\ s$, are identified, resulting in sets $T_{0,[r_t,r_t+H[}$ and $T_{1,[r_t,r_t+H[}$. Constraints (5.15) are linearised:

$$\sum_{\tau \in T_{0,[r_t, r_t+H[}} x_\tau - \sum_{\tau \in T_{1,[r_t, r_t+H[}} x_\tau \leq OB \qquad \forall t \in T \tag{5.16}$$

$$\sum_{\tau \in T_{0,[r_t, r_t+H[}} x_\tau - \sum_{\tau \in T_{1,[r_t, r_t+H[}} x_\tau \geq -OB \qquad \forall t \in T \tag{5.17}$$

Constraints (5.16) and (5.17) may not lead to desirable results for all scenarios, such as one with many more trains running in direction 0 compared to direction 1. The model may still cancel a sufficient number of trains in the busier direction to balance with all trains of the other direction being cancelled. Hence, in case of strong unidirectional traffic, adding balancing constraints may have an adverse effect.

Additionally, these constraints exploit release times rather than actual starting times of trains on the disrupted area. Delaying trains may still result in trains not operating during a period of 1 h. Better would be to define constraints based on the actual starting times t_t, which are unknown upon construction of the model. Hence, this would require additional variables, further complicating the model, and is not considered here.

Minimum level of service The minimum number of trains running in a direction during a certain hour can also be considered as a constraint, e.g. in case of strong unidirectional traffic. Let T^h_{dir} represent the set of trains t running in direction dir with release time within the h^{th} hour, i.e., $r_t \in [3,600\ (h-1), 3,600\ h[$. Ensuring a minimum number of trains $N^{min}_{dir,LOS}$ is modelled as follows:

$$\sum_{t \in T^h_{dir}} (1 - x_t) \geq N^{min}_{dir,LOS} \qquad \forall h \in \mathscr{H}, dir \in \{0,1\} \tag{5.18}$$

The left-hand side of Constraints (5.18) counts the number of trains still running. Similar to balancing constraints, these are based on release times of trains, and make abstraction of delays.

Scenario 8 supplemented scenario 2 with balancing constraints (5.16) and (5.17) using $OB = 1$. Several plausible results exist: (1) a train in direction 0 gets cancelled to restore the balance; (2) train cancellations in direction 1 are more spread out over the scheduling horizon; or (3) one less train cancellation in direction 1. Dispatchers would have difficulties choosing among these options without any decision support.

Comparison of time-distance diagrams for scenarios 2 and 8 (Figure 5.12), shows that the optimal solution was a combination of the first and third options: one less cancellation in direction 1 (full red circles) and an additional one in direction 0 (dashed orange). Whereas before cancellation was only considered for trains running at the start of a batch, this no longer held true for scenario 8. Although cancelling the first one (orange arrow) would also have satisfied Constraints (5.16) and (5.17), the newly added train in direction 0 (dashed orange circle) would have to wait until the train indicated in red had left the corridor. Whereas it is rather counter-intuitive, cancelling the latter was better in terms of total delay.

Table 5.2 reports an objective function value which is only 2% higher than for scenario 2 following a 169 s increase in total delay. Direction 1 got a larger share of this total delay increase, which is acceptable if an additional train can be run.

Imposing a minimum level of service only influences the result of scenario 2 (Figure 5.12, left), if $N^{min}_{1,LOS} \geq 3$. Adding Constraints (5.18) resulted in exactly the same time-distance diagrams and statistics (see Table 5.2) as for scenario 8.

To conclude, both balance and minimum level of service constraints can be used to influence the number of cancellations. In case of balanced timetables, e.g. for the line Gent-Antwerpen during off-peak, balance constraints may render acceptable results. Although not confirmed here, they may be less interesting for scenarios with a strong timetable unbalances, e.g. morning peak on a line towards Brussels. In those cases, direction- and interval-specific minimum level of service constraints might be more plausible.

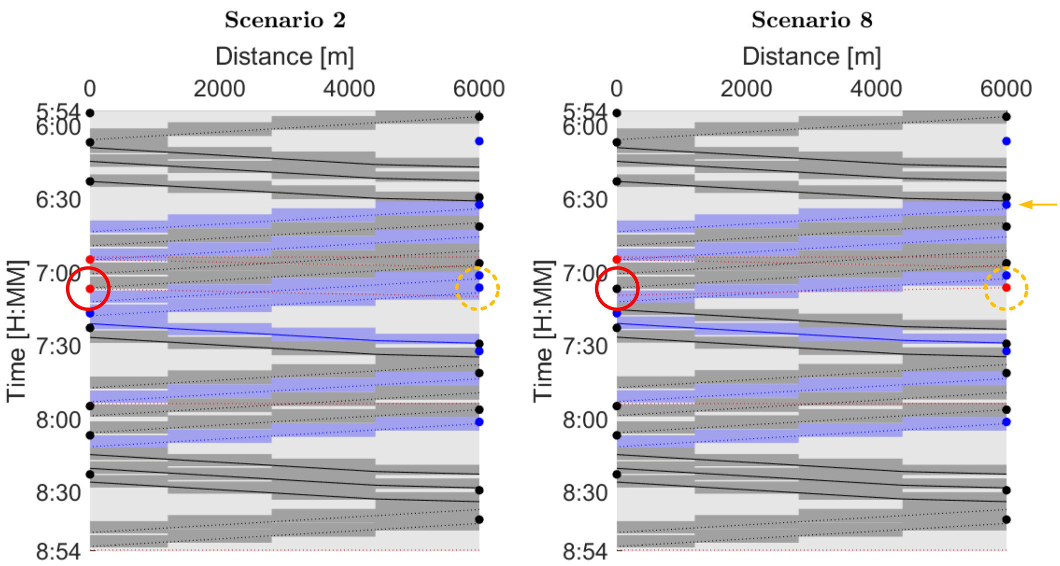


Figure 5.12: Time-distance diagrams for scenarios 2 (left) and 8 (right) of the Tienen-Landen case study.

Including passenger numbers in the objective function

Ideally, the objective function takes into account passenger numbers and penalizes cancellation of crowded trains more than ones with less passengers. Passenger demand is strongly influenced by the time of day as mentioned in Section 2.1, especially on lines connecting Brussels with other parts of the country, and peak hours trains are added to the schedule. Hence, for the Tienen-Landen case, morning peak had more trains running from Landen to Tienen (and Brussels), meaning that the model takes passenger numbers already implicitly into account. Despite the higher number of trains, the average number of passengers per train is still (much) higher towards Brussels during morning peak.

As absolute passenger numbers are unknown [52], the most plausible way to account for them is estimating the relative size of passenger numbers pax_0 and pax_1 in directions 0 and 1. Weights for the objective function (5.9) are adjusted accordingly:

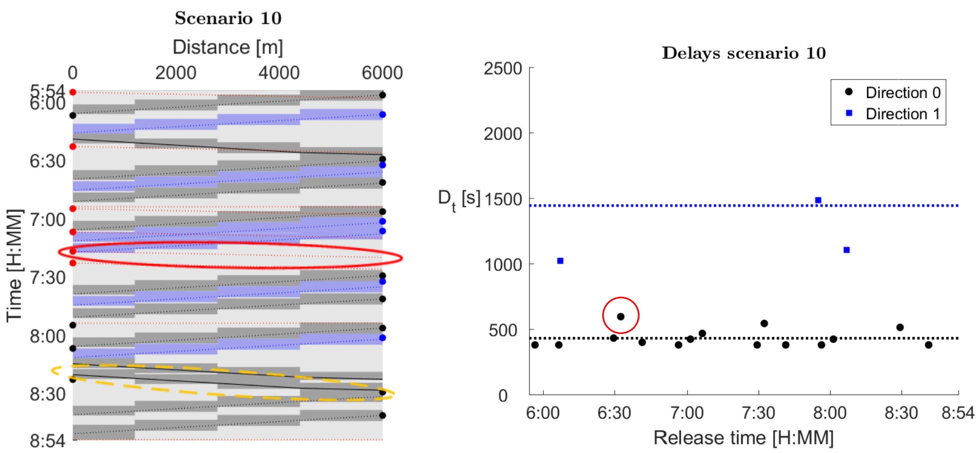
$$w_t^{cancel} \leftarrow \frac{pax_0}{pax_1} w_t^{cancel} \qquad \forall t \in T_0 \qquad (5.19)$$

Equation (5.19) updates the cancellation penalty for direction 0, the one for direction 1 is not altered and serves as a reference. For example, if the number of passengers is twice as large for direction 1 ($pax_1 = 2pax_0$), cancellation of a train in direction 1 gets a double weight compared to direction 0 trains. Penalties for delays $\left(w_t^{delay}\right)$ and rerouting $\left(w_t^{deviation}\right)$ are updated in a similar way.

To investigate the effects of higher passenger numbers, scenario 10 presents an alteration of scenario 2 with trains towards Brussels holding ten times the amount of passengers, i.e. $\frac{pax_0}{pax_1} = 10$ in Equations (5.19). Following the absence of train cancellation in direction 0 for scenario 2, this is not expected to happen for scenario 10 either. Results in Table 5.2 show that six trains were cancelled in the less crowded direction 1.

As the time-distance diagram in Figure 5.13a shows, trains in direction 1 were only added to the schedule if they did not induce (too much) delay for direction 0 trains. Often, dispatchers schedule trains in the most crowded direction and only add trains going in the other if they "fit in" [30]. For example, the train path indicated in red seems to fit in without inducing too much delay for direction 0 and a dispatcher may add it. However, its real infrastructure occupation, modelled by the blocking time stairways, would lead to conflicts and induce delays to all three trains following in direction 0. As dispatchers do not have an idea about the blocking times because of the lack in decision support, this scenario illustrates a strong benefit over ad hoc scheduling. Adding the train in orange (dotted line) induced a knock-on delay for only one train, which still seemed to be acceptable.

Figure 5.13b plots the delays of all scheduled trains, distinguishing them by direction. Interpreting w_t^{cancel} as before, a train t in direction 1 got cancelled if it induced an (unweighed) total delay increase of 360 s for trains in direction 0. The encircled train, the one appointed by the red arrow in Figure 5.13a, has a $D_t \geq 360$. Hence, one may expect the first train in direction 1 would have been cancelled to avoid this delays. As also the preceding train in direction 0 contributed to this delay,



(a)

(b)

Figure 5.13: (a) Time-distance diagrams for scenario 10 of the Tienen-Landen case study. (b) Delays of all scheduled trains in the disruption timetable of scenario 10, distinguishing those in direction 0 (black dots) from those in direction 1 (blue squares). Horizontal dashed lines represent average delays per direction.

is would not have been sufficient to cancel that one. Additionally, Figure 5.13b illustrates how delays in direction 1 increased dramatically, up to about 36 min, to save delays in direction 0.

Relaxing entrance orders

Up until now, the order of trains entering at both sides of the corridor was assumed to be fixed per direction. Nevertheless, changing the order between trains is a commonly adopted measure in several disruption management models (Section 4.4), e.g. swapping fast and slow trains. Enabling all binary q_{ij} variables to be either 0 or 1, may lead to large computation time increases. However, based on characteristics of the existing timetable, the solution space can be reduced by eliminating unlikely order swaps.

Two such restrictions are applied. First, swapping trains with the same journey time (and blocking time stairways) over the corridor is not beneficial and their order variable is fixed as before. Second, trains with release times far apart from each other, e.g. 7:00 and 9:30, are not switched. A threshold value δ^{swap} quantifies the maximum time duration between two trains i and j, in the same direction, in order to allow swapping. This can be formulated as:

$$q_{ij} \begin{cases} = 0 & \text{if } r_i > r_j + \delta^{swap} \\ = 1 & \text{if } r_j > r_i + \delta^{swap} \\ \in \{0, 1\} & \text{otherwise} \end{cases} \qquad \forall i, j \in T \tag{5.20}$$

Both for scenarios 1 and 2, train speeds were homogenized over the disrupted area, as the disruption speed v_D is lower than the maximum speed for IC as well as L trains, respectively 140 and 120 km/h. To illustrate possible effects, the speed of L trains is lowered to $v_L = 50\ km/h$, while $v_D = 80\ km/h$. Relaxing orders without a threshold δ^{swap} (scenario 12), left 63 orders for trains in the same direction to be determined. Table 5.2 reports a slight objective function value improvement (-1.3%) compared to scenario 11 with fixed orders, accomplished by altering only one order. Scenario 13, with $\delta^{swap} = 1,800\ s$, resulted in the same solution but only had 20 order variables to be determined.

The minor gains for both scenarios 12 and 13 over 11 are explained by the small differences in running times over the short disrupted area. In reality, train types may significantly differ in their characteristics, resulting in larger differences in running times over the complete corridor, i.e. including segments A and B. As a result, reordering of trains becomes more beneficial.

Route deviation

For some corridors, (global) rerouting is a plausible alternative to train cancellation, mitigating the effects on passengers. Physically routing a train over alternative corridors is not straightforward. First of all, capacity has to be available on these corridors, including for possible turning operations in stations to switch between corridors. Secondly, these trains are likely to run into conflicts with the originally scheduled train traffic, resulting in secondary delays for the other trains and increased uncertainty for passengers. Hence, it is difficult to estimate both the travel time over this alternative route, and to determine $w_t^{deviation}$.

It is assumed that dispatchers have an idea on how many trains can be operated on the alternative route per hour and per direction. Influencing factors consist of the number of trains running on these corridors and time of day, i.e. the timetable and infrastructure availability. As the model only considers its own corridor, these numbers are provided as input parameters, resulting in the following constraints:

$$\sum_{t \in T_{dir}^h} dev_t \leq N_{dev,dir}^{max} \qquad \forall h \in \mathscr{H} \tag{5.21}$$

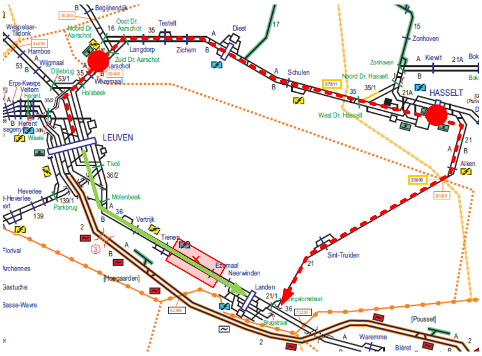
$$x_t + dev_t \leq 1 \qquad \forall t \in T \tag{5.22}$$

Constraints (5.21) limit the number of trains rerouted during hour h in direction dir by the available remaining capacity on the alternative corridors, expressed as a number of trains per hour $\left(N_{dev,dir}^{max}\right)$. Hence, this reasoning makes abstraction of the (unknown) timetable(s) on the other corridor(s). It is advisable to underestimate this number, and to (strongly) overestimate the penalty $w_t^{deviation}$. Constraints (5.22) state trains can either be cancelled or rerouted, but not both. Although these are actually abundant, as the model will never choose both $x_t = 1$ and $dev_t = 1$ (with a double penalty), they limit solution space.

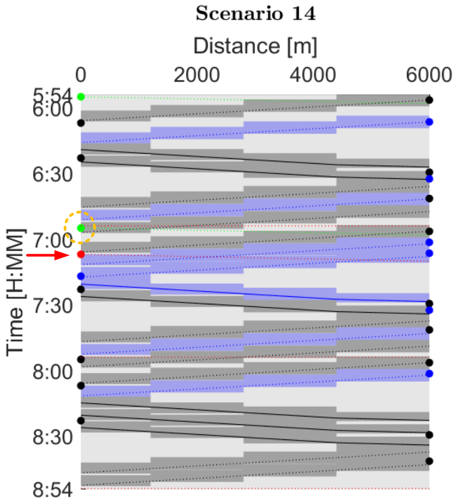
Figure 5.14a presents a plausible alternative route (dotted red line) for passengers travelling from Leuven to Landen on line 36 (full green line) [30]. Based on NMBS' route planner [40], travel time is expected to increase from 21 to 62 min[5]. However, passengers do not perceive the rerouting as only an increased travel time. Additional nuisance follows from, amongst others, crowding, uncertainty and changing travel plans. These factors are not taken into account in this example, but would justify a higher penalty than $w^{deviation} = 2,700\ s$ (45 min) used here. Due to traffic on the other route also being oriented towards Brussels, scenario 14 allowed at most one train in direction 1 to be rerouted per hour, none in direction 0, i.e. $N_{dev,1}^{max} = 1$ and $N_{dev,0}^{max} = 0$.

Table 5.2 reports one less train cancellation for scenario 14 compared to scenario 2. Two trains got rerouted, indicated by the green dots in the time-distance diagram of Figure 5.14b. As expected, one of these reroutings (orange dotted line) avoided a train cancellation. Following capacity restrictions $\left(N_{dev,1}^{max} = 1 \text{ train/h}\right)$, this was not possible for the second cancelled train (indicated by the red arrow). Figure 5.14b and Table 5.2 reveal another advantage of rerouting: it decreased total delay with 3,458 s compared to scenario 2. Hence, the induced delay of the first rerouted train was sufficiently high to justify rerouting it.

[5] Travel time for the alternative route consists of the travel times for an IC train from Leuven to Aarschot, from Aarschot to Hasselt and from Hasselt to Landen, assuming trains can immediately continue their journey, without waiting in the stations. These simplifications are justified due to the lack of knowledge of the timetables on the other corridors: a dispatcher would provide this value to the tool. However, it is expected many conflicts arise on these corridors.



(a)



(b)

Figure 5.14: (a) Trains originally scheduled between Leuven and Landen (green line) can be rerouted along Aarschot and Hasselt (red dots) resulting in the dashed red route. Adapted from [26]. (b) Time-distance diagram for scenario 14 of the Tienen-Landen case study.

Remarks on the practical applicably has to be made. In this scenario, the first train running got deviated. However, in real-life operations, the disruption timetable may still have to be communicated, and the solution may not be realisable. Second, following the discussion on $w^{deviation}$ and its simplifications, it is questionable whether rerouting this first train leads to better overall results. Its passengers may experience a strong increase in travel time, much higher than the expected value of 45 min.

5.6 FIFO and (max,wait) heuristics

The concept of AIPs has the advantage of being applicable to multiple parts of the network by changing a limited number of parameters. On the other hand, one does not have an idea about the effects on other parts of the network. Although the assumptions made here seem to be acceptable, conflicts are likely to arise outside of the corridor. Unfortunately, no tools were at hand to evaluate the implementation of solutions in a straightforward way. One possibility would have been to implement the measures in the simulation tool developed by Van Thielen et al. [56]. However, the area of the Tienen-Landen case study was not part of the input data, and a mismatch exists between the simulation tool and this thesis' assumptions.

While comparison with dispatchers' practice would have been desirable, it could not be conducted for data availability reasons mentioned before (Section 3.1). To give an idea about the model's performance, two heuristics are presented, based on expected dispatcher's behaviour in the current situation, i.e. without decision support tools. Section 5.6.1 presents an adapted version of the first-in-first-out (FIFO) approach to incorporate train cancellation. Based on problem-specific knowledge, Section 5.6.2 describes a heuristic which allows batching of trains.

5.6.1 An extended FIFO rescheduling approach

Currently, dispatchers have no quantitative evaluation tools at hand to evaluate the effects of their measures. A straightforward, but naive, approach is to schedule the longest waiting train, i.e. the one with the smallest release time r_t in the queue. The first train can operate as planned, followed by other trains in their arrival order, regardless of direction.

An exception is made for international trains[6]: they cannot be delayed, nor cancelled. Hence, the FIFO approach schedules trains until it would result in a conflict with an international train. Infrabel's TMS allows to predict conflicts up until 15 min before they occur [56], meaning that conflicts with international trains can be prevented by delaying the train causing it. After the international train has been processed, scheduling continues as before.

Scheduling a new train (Algorithm 1) is done by allowing it onto the corridor after the set-up time with the previous one has elapsed since its completion (line 4). However, if this would result in a delay exceeding a certain value D^{cancel}, which may differ from the model's cancellation penalty w^{cancel}, it is cancelled instead of scheduled (lines 7-8). In the remainder of the thesis, dispatchers are expected to cancel a train with more than 1 h of delay.

Algorithm 1 scheduleTrain procedure

1: **procedure** SCHEDULETRAIN($n,\ r_n,\ s_{pn}$)
2:     $time \leftarrow C_p$
3:     **if** $time + s_{pn} < D^{cancel} + d_n$ **then**
4:         Add the train to the schedule with arrival $= time + s_{pn}$
5:         $sameDir \leftarrow sameDir + 1$
6:     **else**
7:         Cancel this train
8:         $x_n = 1$
9:     **end if**
10: **end procedure**

This rather naive heuristic has been applied on (most of) the scenarios of the Tienen-Landen case study and resulting objective function values are reported in the last columns of Table 5.2. Only for the first scenario, the FIFO approach returned the optimal solution. In all others, the model strongly outperformed the FIFO approach.

Contrasting the time-distance diagrams for scenario 2 in Figure 5.15 solved by both the model (left) and the FIFO heuristic (right), illustrates why the latter performed much worse. First of all, the model cancelled two trains near the start of the disruption to save the induced total delay. The FIFO approach considered

[6] The original timetable for the Tienen-Landen case study did not hold any international trains, but the timetables for the practical case studies for the second and third AIP have at least one.

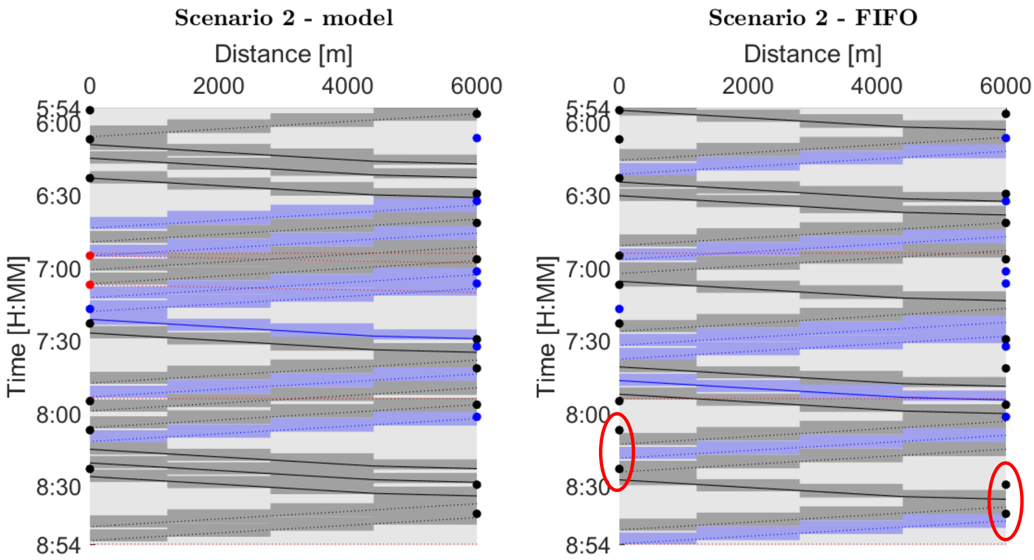


Figure 5.15: Time-distance diagrams for scenario 2 obtained by the model (left) and the FIFO approach (right) of the Tienen-Landen case study.

only the own train's delay to decide on cancellation. Secondly, the latter neglected differences in set-up times and did not batch trains in the same direction together. The model on the other hand, created large batches. As a side-effect, the trains indicated in red in Figure 5.15 (right) still had to be scheduled after the end of the scheduling horizon. Adding them interferes with traffic in the next hour.

Scenarios 6 and 7, which imposed a maximum batch size N_{batch}^{max}, have not been evaluated using this heuristic, as it does not take batch size into account. Therefore, based on observations, a second heuristic has been developed which does so.

5.6.2 (Max, wait) heuristic

A second heuristic uses more problem-specific knowledge, based on observations of the model's results for the Tienen-Landen case study as discussed in Section 5.5. For the scenarios with equal weighing of trains and without additional constraints, two main rules could be observed:

1. If a train of direction $dir \in \{0, 1\}$ is waiting at the start, or arrives within the operation of a batch in direction dir, it is added to this batch.

2. Often, it turned out beneficial to wait a certain time (denoted t^{wait}) after the ending of a batch to operate a train going in the same direction before starting the batch in the other direction. Scenario 7 (Figure 5.11, right) presented an example of this.

Both observations follow from the large increase in set-up times when a train is not followed by one in the same, but by one in the opposite direction.

Consider the following heuristic, presented by the pseudocode in Algorithm 2. The first train arriving after the start of the disruption is added to the disruption timetable and starts a batch in its own direction (lines 4-7), e.g. direction 0. The dispatcher, who knows the completion time C_p of the previous train p [7] checks whether a next train n arrives in the same direction, i.e. is released, before the exit time of the previous one ($r_n \leq C_p$, line 15). If not, this means the queue at the right-hand side of the corridor is empty, and there is a switch in the direction of batches. In case no trains are queuing at the left-hand side, the first train arriving at either end becomes the next train (lines 19-26).

Algorithm 2 (Max,wait) heuristic

1: **Given:** sets of trains T_{dir} with $dir \in \{0, 1\}$; $r_t \quad \forall t \in T$; $s_{ij} \quad \forall i, j \in T$
2: Parameters: N^{max}_{batch}, t^{wait}
3:
4: $T_{dir} \leftarrow$ Sort T_{dir} by release time
5: Schedule the first train
6: $n \leftarrow \underset{t \in T}{\text{argmin}} \{r_t\}$
7: timetable $\leftarrow$ scheduleTrain($n,\ r_n,\ 0$)
8: **while** $T_{dir} \neq \emptyset$ **do**
9:   **if** sameDir $\geq N_{batch}$ **then**
10:     Switch directions
11:     $sameDir = 0$
12:   **else**
13:     Try to schedule the next train in the same direction
14:     $n \leftarrow T_{currentDir}(1)$
15:     **if** $r_n \leq time + t^{wait}$ **then**
16:       $p \leftarrow n; \quad n \leftarrow t$
17:       timetable $\leftarrow$ scheduleTrain($n,\ r_n,\ s_{pn}$)
18:     **else**
19:       **if** $r_n \leq r_{T_{\neg currentDir}(1)}$ **then**
20:         Progress the time
21:         $time \leftarrow r_t$
22:       **else**
23:         Switch directions
24:         $sameDir = 0$
25:         $time \leftarrow r_{T_{\neg currentDir}(1)}$
26:       **end if**
27:     **end if**
28:   **end if**
29: **end while**
30: **Result:** timetable

However, a dispatcher may also look some time t^{wait} ahead. If a train t is released within this t^{wait}, i.e. $C_p < r_n \leq C_p + t^{wait}$, it is added to the batch. The dispatcher can imagine that this heuristic could lead to large batch sizes, hampering traffic in

[7] When a train is scheduled, the train path can be visualized on a screen. Hence, the completion time is known to the dispatcher when the train enters the disrupted area.

the other direction. Therefore, he or she only allows batches of up to N_{batch}^{max} trains (similar to Constraints (5.13) and (5.14)), and switches the direction of traffic upon reaching this limit (lines 9-11).

Trains are cancelled in the same way as for FIFO heuristic. Hence, if the delay of a train t is smaller than D^{cancel}, it is added to the schedule and the counter for the batch size is incremented (line 5 in Algorithm 1).

One expects this heuristic to benefit from the incorporation of problem-specific knowledge, for good combinations of parameters N_{batch}^{max} and t^{wait}. However, these have to be fine-tuned to ensure good performance, and the best set depends on the original timetable's structure. Therefore, the scenarios of the Tienen-Landen case study have not been evaluated with it. Section 6.3 presents the first application by fine-tuning N_{batch}^{max} and t^{wait} on two (artificial) timetables.

5.7 Conclusion

This chapter discussed all steps required to translate inputs concerning the disruption to an optimized disruption timetable in case of partial blockage of a double-track corridor without stops. The dispatchers should retrieve a timetable applying re-timing, reordering and cancellation of trains over a scheduling horizon with minimal effort.

After a discussion on their expected impact, a number of parameters were selected by balancing between general applicability of the model, and aiming at conflict-free timetables. These parameters included the timetable, length of the disrupted track and allowed speeds. For others, assumptions had to be made. Because of its flexibility, developing an optimization model was preferred over simulating a large number of different sets of measures.

The disruption was considered as a single-machine scheduling problem with release times and sequence-dependent set-up times. The former represent the expected arrival time at the disrupted area, the latter ensure minimum headway times as calculated by the blocking time theory. To model cancellation of trains, rejection variables were included for each train. Hence, the objective function balances between total delay and cancellation penalties, which seems to be a difficult decision as cancellation interacts with the delays for multiple other trains. Application of the model to a number of scenarios on a practical case study illustrated how it managed to take into account the delay a certain measure causes to all other trains. Hence, the model outperforms a regular FIFO heuristic. Based on observations, also a new heuristic has been developed.

Some assumptions have been made. First of all, stations and unaffected parts of the corridor are considered able to store all queuing trains, which should be valid in most circumstances as queues are not expected to exceed station capacity. The number of trains queuing remained limited in all scenarios. Secondly, the model assumes a fixed order for trains in the same direction. In addition to its practical benefits, relaxing train orders did not show significant improvements. Thirdly, the model exploits perfect information on the arrival times of incoming trains over the full scheduling horizon, which is a major assumption. To circumvent this, a rolling horizon approach could be used to re-run the model in case of (major) information updates. Other assumptions, such as on acceleration and braking rates, mainly served the purpose of running time calculation, which should for real-life application be derived from either the original timetable, or Infrabel's own calculation methods.

Dispatchers can use the model to generate a disruption timetable by providing a minimum of input data: the length of segment D, the disruption speed, the direction of switches, and the scheduling horizon. Timetable data, including delay predictions, should be extracted from existing information systems. These enable the model to quickly generate an optimal disruption timetable. Moreover, the dispatcher could specify additional constraints such as a maximum delay per train, e.g. to guarantee a transfer, to influence train cancellation, or to avoid high waiting times for a specific direction. Finally, passenger numbers could be taken into account as relative numbers.

To thoroughly evaluate the impact of the parameters mentioned here, Chapter 6 applies the model developed in this chapter on a theoretical case study for a number of scenarios. One could argument this AIP is not commonly encountered on the network because of its specific configuration without any stops. Therefore, Chapters 7 and 8 extend the presented AIP by including stops, and multiple tracks respectively.

Chapter 6

Partial Blockage of a Double-Track Corridor - Parameter Impact Assessment

Chapter 5 focussed on the development of the model and its framework for the first AIP by iteratively modelling a practical case study, i.e. the Tienen-Landen corridor. However, most of the parameters such as timetable frequency and length of segment D could not be changed. This chapter analyses the impact of differing values for such parameters on an artificial case study to provide an answer to the third subquestion (Section 3.2). Secondly, the structural analysis may allow to derive some general guidelines which can immediately lead to a good response during a certain disruption.

Section 6.1 introduces the artificial infrastructure and timetable(s), after which Section 6.2 assesses the interplay between frequency and the cancellation penalty w^{cancel}. Both heuristics presented in Section 5.6 are used to evaluate solution quality, which requires tailoring of the parameters for the (max,wait) heuristic (Section 6.3). Thereafter, Sections 6.4 and 6.5 evaluate the impact of the length of segment D and disruption speed v_D respectively. Accounting for passenger numbers may seriously affect the model's solution, Section 6.6 considers this parameters. Section 6.7 elaborates on two strategies to include buffer times in order to stabilize operations, and their impact on solution quality. To conclude, Section 6.8 summarizes the main findings and presents some "rules-of-thumb".

6.1 Case study description

Figure 6.1 presents the infrastructure and its speed limitations considered within this chapter: a corridor consisting of two double switches located in a block section with length 1,400 m. In-between both switches, segment D contains three block sections with a length of 1,600 m each. Without loss of generality, segments A and B have not been incorporated. A disruption occurs on the lower track, i.e. in direction 0, and forces the trains in direction 0 (full blue line) to be rerouted over the complete disrupted area.

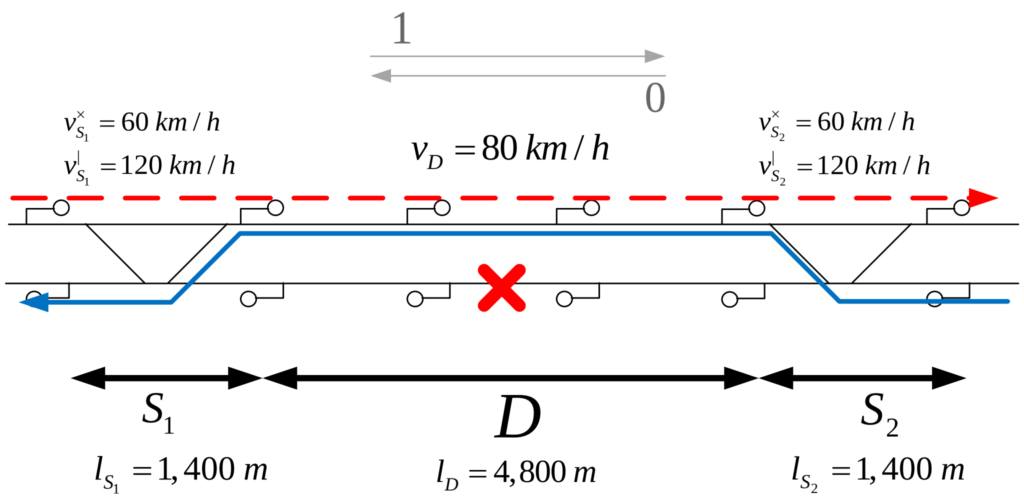


Figure 6.1: Representation of the infrastructure used for the artificial case study to assess the impact of parameters for the first AIP. Segment D holds three block sections in this example, and a disruption on the lower track forces trains in direction 0 (full blue line) to cross over segments S_1 and S_2.

The complete hourly timetable operates six trains per hour per direction and its time-distance diagram is shown in Figure 6.2. Four IC trains (in black) and two L trains (in blue) are operated with entry times uniformly distributed over the hour per train type. Table 6.1 reports their entry times as minutes after the hour, together with train type and direction. For example, the second one is an IC train always arriving at the disrupted area 15 min after the hour in direction 1, from left to right. Three different timetables are considered, Table 6.1 mentions the trains operating in each of them. The dispatcher estimates the disruption to last 3 h and schedules traffic over this horizon.

Solving some of the scenarios required excessive computation times. Appendix B presents a discussion, advising three computational settings for the analysis in this thesis, including the necessary motivation:

1. Basic settings with a computation time limit of 120 s.
2. Basic settings with a computation time limit of 900 s.
3. Focus on finding feasible solutions rather than proving optimality during 120 s.

Throughout this chapter, reported results are those obtained by the third settings, referred to as *MIPfocus*, because of the practical applicability unless specified otherwise. Where deemed interesting, a comparison with the other results is presented. Tables will report the computation times, for those reaching the 120 s time limit for MIPfocus settings, optimality could not be proven. An asterisk (*) is used to indicate scenarios for which results are the same as those obtained within 900 s and proven optimal.

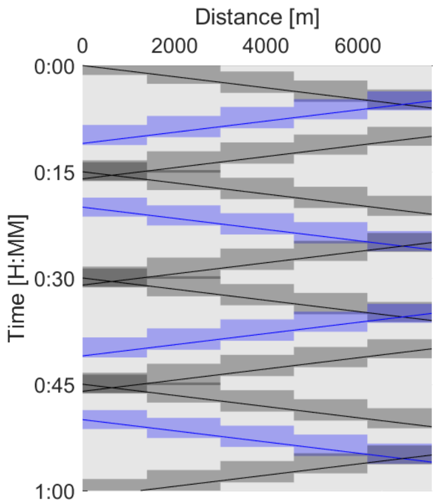


Figure 6.2: Time-distance diagram for one hour of the original high-frequency timetable of the artificial case study for the first AIP. Blocking time stairways already account for the lower disruption speed $v_D = 80\ km/h$, and show many overlaps.

Table 6.1: Timetable information of the artificial case study to assess the impact of parameters for the first AIP. Information concerns the train type, its entry time, direction and the timetable(s) they are operated in (x).

Train type	Entry time (min)	Direction	Timetable		
			Low	Mid	High
IC	0	1	x	x	x
IC	15	1		x	x
L	20	1			x
IC	30	1	x	x	x
IC	45	1		x	x
L	50	1			x
L	5	0			x
IC	10	0	x	x	x
IC	25	0		x	x
L	35	0			x
IC	40	0	x	x	x
IC	55	0		x	x

6.2 Frequency and cancellation penalty

The number of trains originally scheduled to run on the corridor may have a significant effect on the final result. Two effects can be expected with increasing frequency: (1) increasing batch sizes to reduce total delay; and (2) more train cancellations. To assess the impact of frequency on the final result, three different timetables are used. As presented in Table 6.1, the “low-”, “mid-” and “high-frequency” timetables consist of respectively two, four and six trains per hour per direction. The additional regional trains distort the regularity in the “high-frequency” timetable.

Additionally, the penalty for cancellation (w^{cancel}) is varied to assess its effect, which is expected to become more pronounced with increasing frequency. Running more trains means that a delayed train may induce delays to more other trains. Imagine ten trains running directly after a train t and $w^{cancel} = 3,600\ s$. If train t induces on average more than 6 min of delay for each following one, it gets cancelled. Increasing w^{cancel} to 7,200 s, increases this threshold to 12 min which is still an acceptable limit. Regardless of this intuitive interpretation, the choice for w^{cancel} should represent the nuisance as perceived by passengers, as discussed in Section 5.5.1.

Finally, also the FIFO approach (Section 5.6.1) is considered, expected to perform worse with increasing frequency. Moreover, the busier the timetable gets, the more trains will spill over the scheduling horizon as happened for the Tienen-Landen case study.

Table 6.2 presents information on all assessed scenarios, indicating both which timetable was used and whether the model (with a w^{cancel}) or the FIFO has been applied. Next, delay statistics and number of cancelled trains are reported, which together constitute the objective function value. The last column reports the required CPU time.

6.2.1 Low- and mid-frequency timetable results

Table 6.2 reports no cancellations for any of the scenarios with the low- and mid-frequency timetables (A1-A6). In case trains did not get cancelled with $w^{cancel} = 3,600\ s$, neither will they for increasing penalty. Nonetheless, a higher w^{cancel} had a positive effect by lowering the computation time as in balancing between cancelling a train, or scheduling it with its induced delay, the latter became more attractive.

For low-frequency timetables, the FIFO approach (scenario A1) resulted in the same solution as the model (scenarios A2 and A3). The combination of the low frequency and the uniform spread in time, resulted in train batches holding exactly one train. In such a situation, a FIFO approach should always find the optimal result: in case it would not be optimal, the model would have batched two trains together. Here, sufficient time between trains in opposite direction was present, leading to only small delays and not justifying train batching. As this timetable is considered less interesting, it will not be used in the remainder of the thesis.

Table 6.2: Description of each scenario when varying the timetable frequency and cancellation penalty w^{cancel}, presenting information on the set-up, possibly using the FIFO approach. Solution statistics report the maximum, total and average delay values, together with the number of cancellations, constituting the weighted objective function. Computation times for the MIPfocus settings are reported. An asterisk (*) is used to indicate scenarios for which results are the same as those obtained within 900 s and proven optimal.

#	Timetable (trains/h /direction)	Cancellation penalty (w^{cancel})	Max. delay (s)	Total delay (s)	Average delay (s)	Number cancelled	Objective function value	CPU time (s)
A1		FIFO	715	6,720	560	0	1,416	0.0
A2	Low (2)	3,600	715	6,720	560	0	1,416	0.3
A3		7,200	715	6,720	560	0	1,416	0.2
A4		FIFO	2,021	26,567	1,107	0	22,715	0.0
A5	Mid (4)	3,600	1,004	10,985	458	0	7,133	1.4
A6		7,200	1,004	10,985	458	0	7,133	0.6
A7		FIFO	3,557	70,799	2,082	2	72,623	0.1
A8	High (6)	3,600	1,369	21,204	606	1	19,227	120.1
A9		5,400	1,604	25,105	697	0	19,327	120.1*
A10		7,200	1,604	25,105	697	0	19,327	120.1*

Similar as for the low-frequency timetable, trains in the mid-frequency one arrived at the disrupted area from both directions in an alternating way. Hence, the FIFO approach resulted in train batches of exactly one train. Due to the increased timetable density, this strategy led to train traffic spilling over the end of the scheduling horizon (t_{end}) as shown in Figure 6.3 (left). Comparing this with the time-distance diagram obtained by the model (Figure 6.3, right), illustrates the value of waiting for an additional train to be processed by a batch. The train indicated in red was added to the batch, which increased batch size and an additional train started queuing at the right end. Hence, also the next batch in direction 0 had two trains, and a pattern got established. Comparing delay statistics for scenarios A4 and A5 in Table 6.2 illustrates the resulting improvements concerning delay statistics: average delay decreased from 1,107 s to 458 s.

During the first 30 min, batches in scenario A5 (orange dashed box in Figure 6.3, right) contained only one train, as no other trains were queuing yet. The result can be interpreted as follows: waiting for a second train in the same direction would reduce delay for that one, although incurring additional delays for multiple other trains. The latter increases outweighed the delay savings for that second train, thereby creating batches of one train.

6.2.2 High-frequency timetable results

Train cancellation became beneficial for those scenarios with a the high-frequency (A7-A10). First of all, the FIFO approach (scenario A7) resulted in adverse effects,

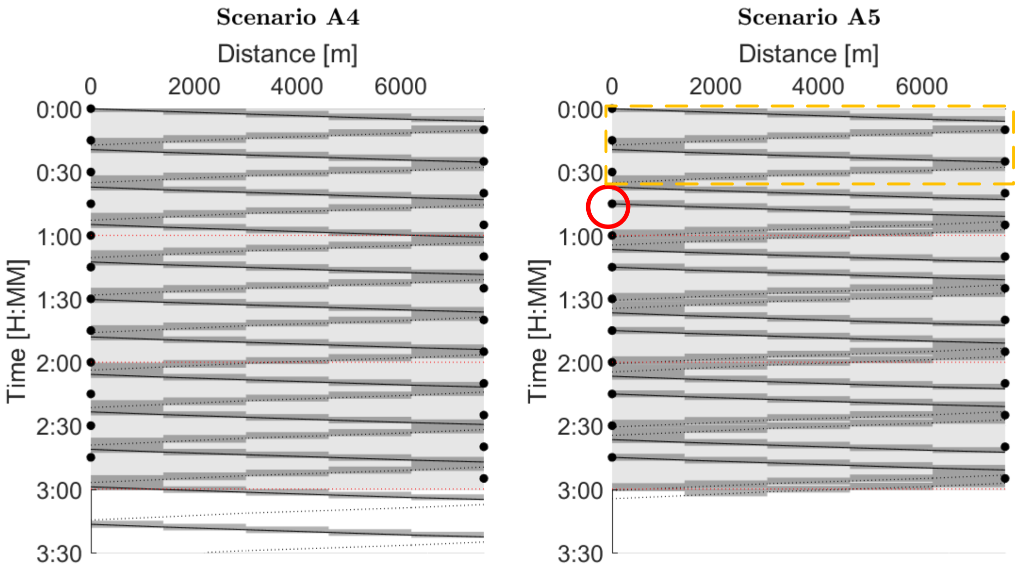


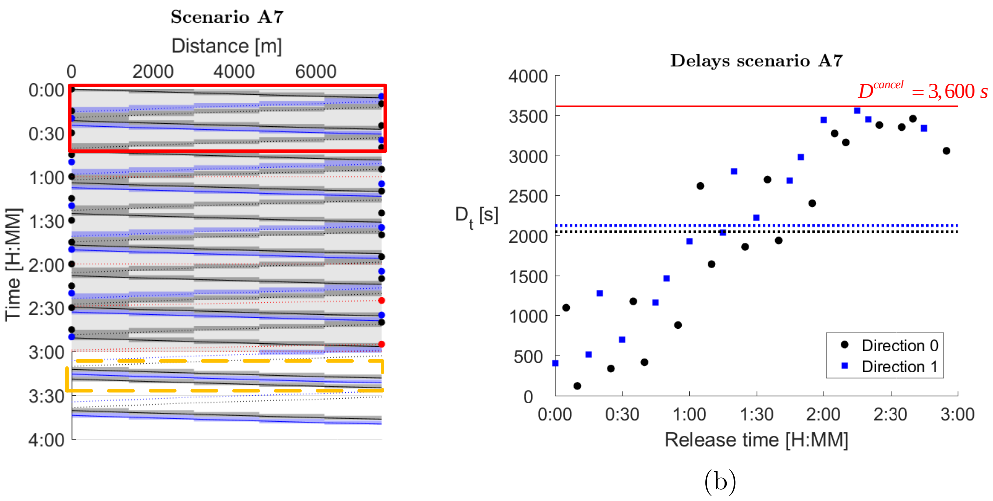
Figure 6.3: Time-distance diagrams for scenarios A4, solved by the FIFO approach (left) and A5 (right).

and a 376% increase in objective function value compared to the model. Following the introduction of irregularity in the original timetable, batches of two trains appeared in its time-distance diagram (Figure 6.4a). The FIFO approach only considered release times of trains when scheduling, regardless of the number of trains queuing. Following timetable periodicity, batch sizes exhibited a repetitive pattern as indicated by the red box in Figure 6.4a. However, delays for both directions strongly accumulated along time as shown in Figure 6.4b and by the maximum delays in Table 6.2. Near t_{end}, delays became too big and trains with $D_t > 3,600\ s$ were cancelled, disturbing the batch size regularity (orange dashed box in Figure 6.4a).

Scenario A8, with $w^{cancel} = 3,600\ s$, resulted in only one train cancellation. Although this may not seem to be a difficult decision, two problem characteristics complicate it and increase the required computation time. One, the homogenized speed resulted in highly similar trains. Two, a train cancellation may affect the optimal batch size for the remaining traffic and train ordering has to be reassessed. As a result, the solver could not prove optimality within 900 s for scenario A8. Nonetheless, results were considered to be plausible.

Figure 6.5 shows the time-distance diagrams associated with the solutions for scenarios A8 (left) and A10 (right). For the latter, the insertion of the cancelled train (red, full arrow) made it beneficial to add a train in the other direction to an earlier batch (orange dashed arrow). Hence, 44 out of 1,260 order variables changed values, and batches were rearranged. As observed in Section 5.5.4, cancelling trains was only conducted for the first one of a batch.

Increasing w^{cancel} to 7,200 s in scenario A10, no longer resulted in train cancellation, concluding that the cancelled train had an induced delay between 3,600



(a)

(b)

Figure 6.4: (a) Time-distance diagram for scenarios A7, solved by the FIFO approach. (b) Train delays for increasing release time in scenario A7, separated by direction. The horizontal dotted lines indicate average delay for each direction.

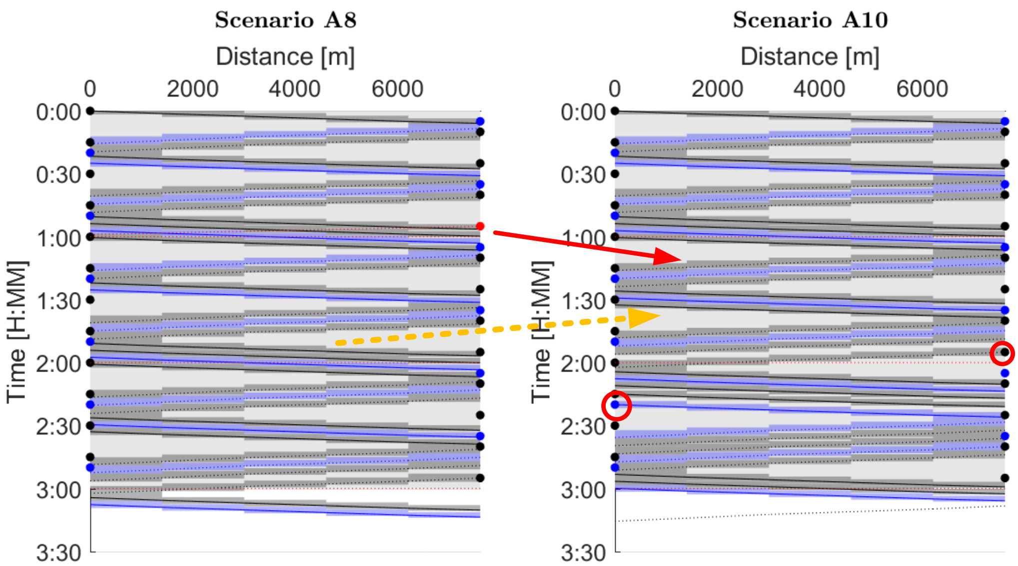


Figure 6.5: Time-distance diagrams for scenarios A8 (left) and A10 (right) on the first AIP.

and 7,200 s. Additionally, optimality could be proven within 210 s. Comparison of total delay for scenarios A8 and A10 in Table 6.2 learns that the cancellation in scenario A8 reduced total delay with about 3,900 s. Hence, assuming the solution for $w^{cancel} = 3,600\ s$ is the optimal one, this is the total delay induced by inserting that train. Optimality for scenario A9 with $w^{cancel} = 5,400\ s$ could only be proven after 647 s. It seems that the closer w^{cancel} gets to the highest induced delay , the more difficult it becomes to balance between delaying and cancellation.

Exploiting capacity of a single-track line as much as possible would lead to all trains in one direction being batched together, followed by the other. However, timetable structure affects such as solution, rendering it suboptimal. Results for the high-frequency timetable illustrate this behaviour. In case none got cancelled (scenario A10), Figure 6.5 (right) shows batch sizes in general increased along time. At some points, it was beneficial to wait for an additional train to join the batch, e.g. the red encircled ones, leading to increases in batch sizes for the remainder of the disruption (in line with scenario A5 in Figure 6.3, right).

Finally, in all scenarios, trains left within the next batch after joining the queue at either end of the disrupted area. In other words, a batch never ends if there is still a train waiting. Assume that the model would decide differently, i.e. let the last train i in the queue wait for the next batch. Hence, a four-train batch in the other direction could leave a time h^{same} earlier,[1] saving a delay of $4h^{same}$. However, train i would have to wait longer, as set-up time is larger between trains in opposite directions, increasing its delay with more than $4h^{same}$. In absence of train cancellation, the

[1]Let h^{same} denote the minimum headway time between trains running in the same direction.

model minimizes total delay, proving that this will never happen.

6.2.3 Conclusion

For low-frequency scenarios with sufficient time between trains, the FIFO approach seems to render satisfactory results. For more realistic scenarios with a higher frequency, the model outperforms the heuristic for two main reasons.

On the one hand, the model manages to improve results by batching trains efficiently. Two main rules could be identified. First of all, trains queuing at either end, join the current or the next batch of their own direction. Secondly, waiting for an additional train to join the batch could be beneficial in some cases, yet timetable structure seems to be crucial in this decision. Both rules strengthen the belief that the (max,wait) heuristic, presented in Section 5.6.2, could render good results. Therefore, this will be discussed further in the next section.

On the other hand, train cancellation is handled more efficiently by the model. As opposed to the FIFO heuristic which resulted in delay accumulation for the high-frequency timetable, the model seems to anticipate future delays. Results confirmed the intuitive choice of cancelling only early trains in a batch. Although this seems to be a straightforward rule, it may rearrange a number of batches, and could result in difficulties finding the optimal results.

6.3 Tailoring the (max,wait) heuristic

Several of the observations and points raised in Section 6.2.3 are actually in line with what the (max,wait) heuristic, introduced in Section 5.6.2, tries to accomplish. To determine the best possible settings for the (max,wait) heuristic, its results for varying parameter values are compared with results obtained by the model (scenarios A6 and A10 in Table 6.2) and the FIFO approach (scenarios A4 and A7).

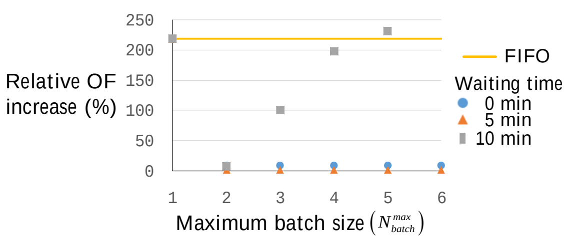
Two analyses are conducted, i.e. for the mid- and high-frequency timetables, with maximum batch size N_{batch}^{max} varying from 1 to 6 and waiting time t^{wait} either 0, 5 or 10 min. Tables 6.3 and 6.4 report for each combination of parameter settings the number of cancelled trains, total delay and objective function value (calculated using $w^{cancel} = 7,200$). The last two columns report the relative increases in total delay and objective function value compared to those obtained by the model. Figures 6.6a and 6.6b plot the relative objective function value increase for the mid- and high-frequency timetables respectively.

6.3.1 Mid-frequency timetable results

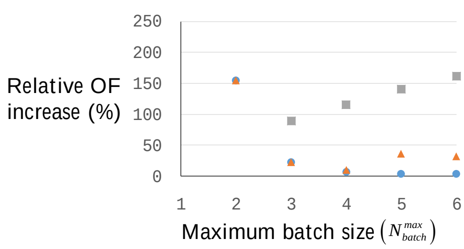
For the mid-frequency timetable, the heuristic's results got very close to those of the model for no or limited t^{wait} as reported by Table 6.3. Figure 6.6a shows solutions severely worsened for high t^{wait}, resulting from waiting for an additional train, which subsequently led to more trains queuing at the other end of the corridor. As a result, queues and batch sizes kept increasing along time.

Table 6.3: Total delay, number of cancellations and score on the objective function for results obtained by applying the (max,wait) heuristic with varying parameter values on the mid-frequency timetable. The last two columns report the relative increases compared to the model's results.

Waiting time t^{wait} (min)	Max. batch size N_{batch}^{max}	Number cancelled	Total delay (s)	Objective function value	Relative total delay increase	Relative objective function value increase
0	1	0	26,567	22,715	142%	218%
	2	0	11,649	7,797	6%	9%
	3	0	11,649	7,797	6%	9%
	4	0	11,649	7,797	6%	9%
	5	0	11,649	7,797	6%	9%
	6	0	11,649	7,797	6%	9%
5	1	0	26,567	22,715	142%	218%
	2	0	11,093	7,241	1%	2%
	3	0	11,093	7,241	1%	2%
	4	0	11,093	7,241	1%	2%
	5	0	11,093	7,241	1%	2%
	6	0	11,093	7,241	1%	2%
10	1	0	26,567	22,715	142%	218%
	2	0	11,471	7,619	4%	7%
	3	0	18,174	14,322	65%	101%
	4	0	25,132	21,280	129%	198%
	5	1	20,076	23,625	83%	231%
	6	2	20,350	31,300	85%	339%
FIFO		0	26,567	22,715	142%	218%
Model		0	10,985	7,133	-	-



(a) Mid-frequency



(b) High-frequency

Figure 6.6: Relative increases in objective function (OF) value of results obtained by the (max,wait) heuristic for varying N_{batch}^{max} (horizontal axis) and t_{wait} on the (a) mid- and (b) high-frequency timetables.

Choosing $t^{wait} = 5\ min$ resulted in slightly better performance than not waiting by allowing multiple batches of two trains as shown in the time-distance diagram in Figure 6.7 (right). However, comparing the result with the results of the model (Figure 6.7, left) shows it was not optimal to wait for the train indicated by the red arrow. It would have been better to wait for the orange dashed encircled one in Figure 6.7 (left). Hence, extending batch size by waiting for an additional train is rather subtle: it may improve results as well as worsen them.

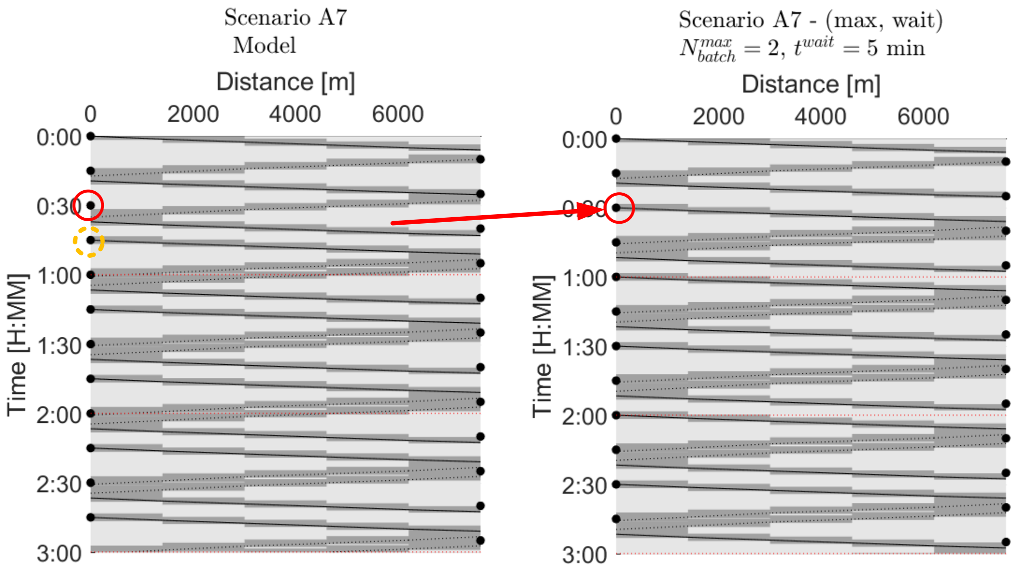


Figure 6.7: Time-distance diagrams for results of the mid-frequency timetable, obtained by the model (left, scenario A7) and the (max,wait) heuristic with $N_{batch}^{max} = 2$ and $t^{wait} = 5\ min$ (right).

6.3.2 High-frequency timetable results

Allowing at most one train per batch, performed worse than the FIFO approach as indicated in Table 6.4: the latter scheduled some batches with two trains as release times in the same direction were close in time. Additionally, a high number of trains had to be cancelled. Figure 6.6b shows that increasing N_{batch}^{max} without waiting, improved performance. A small t^{wait} resulted in different solutions only for larger values of N_{batch}^{max} for two reasons. First of all, for small values of N_{batch}^{max}, batches were already full even without waiting for additional trains. Secondly, no train may arrive within acceptable time. As for the mid-frequency timetable, waiting too long had adverse effects.

Choosing N_{batch}^{max} equal to the largest batch size for the model's result, i.e. five, and without waiting, results were still about 4% worse. Figure 6.8 displays time-distance diagrams of both the model's result for scenario A10 (left) and the (max,wait) heuristic with these settings (right), indicating the fundamental difference in red.

Table 6.4: Total delay, number of cancellations and score on the objective function for results obtained by applying the (max,wait) heuristic with varying parameter values on the high-frequency timetable. The last two columns report the relative increases compared to the model's results.

Waiting time t^{wait} (min)	Max. batch size N^{max}_{batch}	Number cancelled	Total delay (s)	Objective function value	Relative total delay increase	Relative objective function value increase
0	1	0	62,396	130,223	149%	574%
	2	0	55,026	49,248	119%	155%
	3	0	29,569	23,791	18%	23%
	4	0	26,473	20,695	5%	7%
	5	0	25,808	20,030	3%	4%
	6	0	25,808	20,030	3%	4%
5	1	0	62,396	130,223	149%	574%
	2	0	55,026	49,248	119%	155%
	3	0	29,569	23,791	18%	23%
	4	0	27,079	21,301	8%	10%
	5	0	32,035	26,257	28%	36%
	6	0	31,325	25,547	25%	32%
10	1	0	62,396	130,223	149%	574%
	2	0	69,056	70,679	175%	266%
	3	0	42,261	36,483	68%	89%
	4	0	47,324	41,546	89%	115%
	5	1	52,279	46,501	108%	141%
	6	2	56,337	50,559	124%	162%
FIFO		2	70,799	324,623	182%	1580%
Model		0	25,105	19,327	-	-

For the heuristic, the second batch in direction 1 got larger than for the model, as the latter decided to postpone a train arriving during the processing of this batch to the next batch. Due to a snowball effect of increasing batch sizes, this influenced the remainder of the timetable. The (max,wait) heuristic did not manage to balance between postponing the train, and scheduling it. The latter is always done, unless batch size would exceed N^{max}_{batch}.

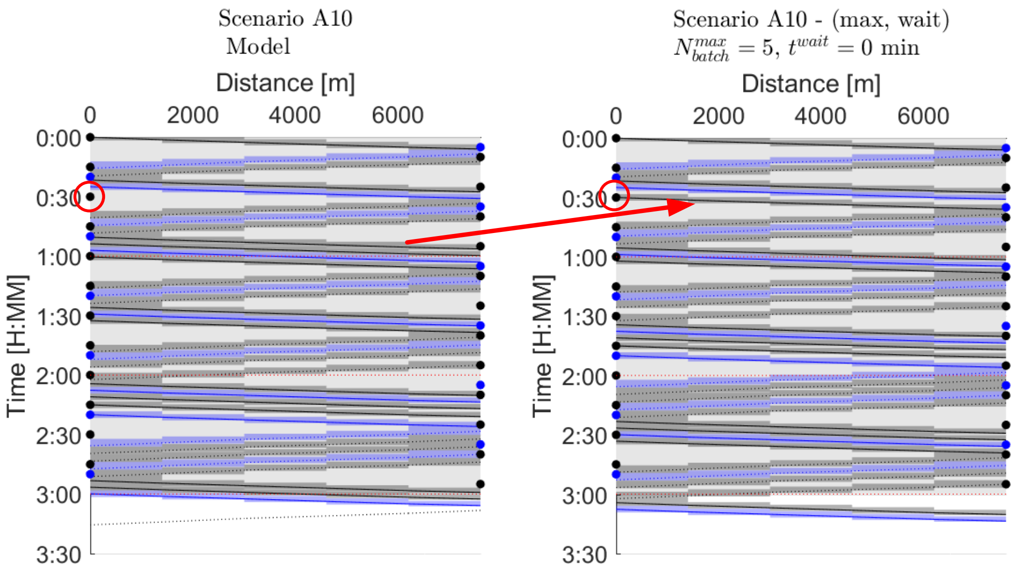


Figure 6.8: Time-distance diagrams for results of the high-frequency timetable, obtained by the model (left, scenario A10) and the (max,wait) heuristic with $N_{batch}^{max} = 5$ and $t^{wait} = 0\ min$ (right).

6.3.3 Conclusion

The better performance of the (max,wait) heuristic over the FIFO is out of question. Whereas the FIFO approach can be applied without any changes, the (max,wait) heuristic requires values as input for its two parameters. Despite the incorporation of more problem-specific knowledge, it still performed worse than the model. First of all, waiting for an additional train becomes a fixed decision rather than a balance between postponing and scheduling it. Although not evaluated here, results are expected to worsen when cancellation comes into play as the (max,wait) heuristic does not account for induced delays.

Analyses for both mid- and high-frequency timetables were based on knowledge about the optimal solution. Conducting these experiments for all possible scenarios does not make sense as dispatchers do not have the optimal solution at hand when choosing the parameters. For the mid-frequency timetable, the (max,wait) heuristic performed best with $N_{batch}^{max} = 2$ and $t^{wait} = 5\ min$. On the other hand, allowing $N_{batch}^{max} = 5$ without waiting rendered the best performance for the high-frequency timetable. Therefore, these parameter sets are selected to apply the (max,wait) heuristic on the respective timetables for the remainder of the thesis.

6.4 Length of segment D

Assuming a constant average speed over the disrupted area, running times increase (approximately) linearly with its length. As block section length is a fixed value on open-track corridors of the Belgian railway network, varying this length is equivalent with varying the number of block sections in segment D (n_D). In practice, the number of blocked block sections, and thus the length of segment D, will be determined by the availability of the required switches before and after the blocked area. Minimum headway times between trains in opposite directions increase with the length of the segment, which leads to larger batch sizes.

Following the analysis in Section 6.2 and the reasoning on nuisance caused to passengers, a cancellation penalty of 7,200 was considered here. All other input parameters were the same as those described in Section 6.1, except for n_D. Initially, eight different scenarios were assessed, using four different values for n_D in combination with two different timetables. The number of block sections n_D was equal to 1, 3, 6, or 20, resulting in a disrupted area with lengths 4.4, 7.6, 12.4 and 34.8 km respectively. In a dense railway network as the Belgian one, the latter value is highly unlikely to be found, but served as validation of the model.

Table 6.5 reports information of the timetable and number of block sections in segment D for each scenario, followed by delay and cancellation statistics. Objective function values are reported for the model's results, and those obtained with the FIFO and (max,wait) heuristics. The MIPfocus settings rendered the best results for each scenario, although optimality could not be proven for scenarios B7 and B8.

Table 6.5 shows a general trend of increasing average delay with increasing n_D. On the one hand, the increased distance to be covered resulted in a larger impact of the speed decrease. On the other hand, larger batch sizes led to more delays for other trains, but allowed to avoid cancellation for all scenarios, except the less realistic ones (B4 and B8).

Additionally, Table 6.5 illustrates that choosing parameter values for the (max,wait) heuristic is not as straightforward as fine-tuning them for a single case study: with increasing n_D, its performance deteriorated. Intuitively, one expects better results when increasing N_{batch}^{max} with increasing n_D. However, Section 6.3 showed that t^{wait} may also have significant impact.

6.4.1 Mid-frequency timetable results

For the mid-frequency, regular, timetable, scenario B1 with a single block section led to only very minor delays. Both heuristics rendered the exact same results, as trains do not hinder each other. With increasing n_D, the FIFO approach still scheduled trains in the same order, whereas the (max,wait) heuristic succeeded in forming batches of up to two trains ($N_{batch}^{max} = 2$) initially mitigating the increase in total delay. Performance got progressively worse for increasing n_D. As a result, maximum delays strongly increased with n_D for both heuristics, e.g. 2,347 and 3,510 s for respectively the FIFO and (max,wait) heuristic applied to scenario B3.

Table 6.5: Description of each scenario when varying the number of block sections in segment D. Solution statistics report the maximum, total and average delay values, together with the number of cancellations, constituting the weighted objective function. Computation times for the MIPfocus settings are reported. An asterisk (*) is used to indicate scenarios for which results are the same as those obtained within 900 s and proven optimal. Finally, the last columns report the objective function value obtained by respectively the FIFO and (max,wait) heuristics, and their relative increase compared to the model's results.

#	Timetable (trains/h /direction)	Length D n_D	Max. delay (s)	Total delay (s)	Average delay (s)	Number cancelled	Objective function value	CPU time (s)	FIFO		(Max, wait)	
									Absolute value	Relative increase	Absolute value	Relative increase
B1	Mid (4)	1	167	3,573	149	0	1,089	0.4	1,089	0%	1,089	0%
B2		3	1,004	10,985	458	0	7,133	0.6	22,715	218%	7,241	2%
B3		6	1,587	18,286	762	0	12,562	1.7	66,966	433%	23,600	88%
B4		20	3,915	41,136	1,959	3	50,043	46.6	92,365	85%	101,352	103%
B5	High (6)	1	1,067	11,769	327	0	8,043	9.3	13,286	65%	9,501	18%
B6		3	1,604	25,105	697	0	19,327	120.1*	324,623	1580%	20,030	4%
B7		6	2,510	43,976	1,222	0	35,390	120.1	117,230	231%	49,840	41%
B8		20	4,184	71,591	2,309	5	88,873	120.1	144,989	63%	134,691	52%

Time-distance diagrams confirmed the hypothesis of increasing batch size with increasing n_D. Scenario B2 was equivalent to scenario A5 for which the time-distance diagram was shown in Figure 6.3 (right). Increasing n_D to six (scenario B3) and eight block sections led to the time-distance diagrams in Figure 6.9.

Next to larger batch sizes, all three showed a tendency towards constant batch sizes when time increases. After a transition period ($t^{transition}$), sufficient number of trains were queuing to constitute these constant batches. Part of this is explained by the regularity of the timetable. Red dashed lines in the time-distance diagrams in Figure 6.9 split the timetable in a transition part (on top of it) and one with a constant batch size (beneath it). This illustrates that if batch sizes are the same for increasing n_D, the maximum batch size was reached earlier, i.e. $t^{transition}$ decreased.

To investigate whether both effects of increasing batch size and decreasing transition time are fundamental or merely a coincidence, scenarios with multiple intermediate values for n_D between six and twelve block sections, have been run. Table 6.6 reports the number of block sections n_D, size of each batch until a fixed value has been reached (indicated in bold) and the transition time $t^{transition}$. From ten block sections on, the resulting pattern seems to stabilize, possibly because increasing the batch size would require even more trains to be queuing. The increased running times would slightly delay the starting times of all batches, but not sufficiently to make it worthwhile waiting for the next train to arrive. In general, the reported values confirm the earlier observed trend.

6.4.2 High-frequency timetable results

It is expected that a higher frequency leads to larger batches, as more trains are queuing at the moment the next batch starts, or arrive during it[2]. The regular batch size observed for solutions to the mid-frequency scenarios, was attributed to timetable regularity. Question is whether disturbing that regularity with additional L trains in the high-frequency one, would result in different the conclusions.

[2]Section 6.2 already advocated why it is not beneficial to leave a train waiting if it arrives during the processing of a batch in its own direction.

Table 6.6: Optimal batch sizes and transition time for increasing number of block sections in segment D with the mid-frequency timetable. Bold entries in the second column indicate the constant batch sizes obtained after the transition period.

Number of block sections in D (n_D)	Batch sizes	Transition time $t^{transition}$ (min)
1	1 / 1 / **1** / ...	0
3	1 / 1 / 1 / 1 / **2** / ...	35
6	1 / 1 / 2 / **3** / ...	45
8	2 / **3** / ...	30
10	2 / 3 / 4 / **5** / ...	90
12	2 / 3 / 4 / **5** / ...	90
14	1 / 2 / 4 / **5** / ...	80
20	2 / 5 / **6** / ...	100

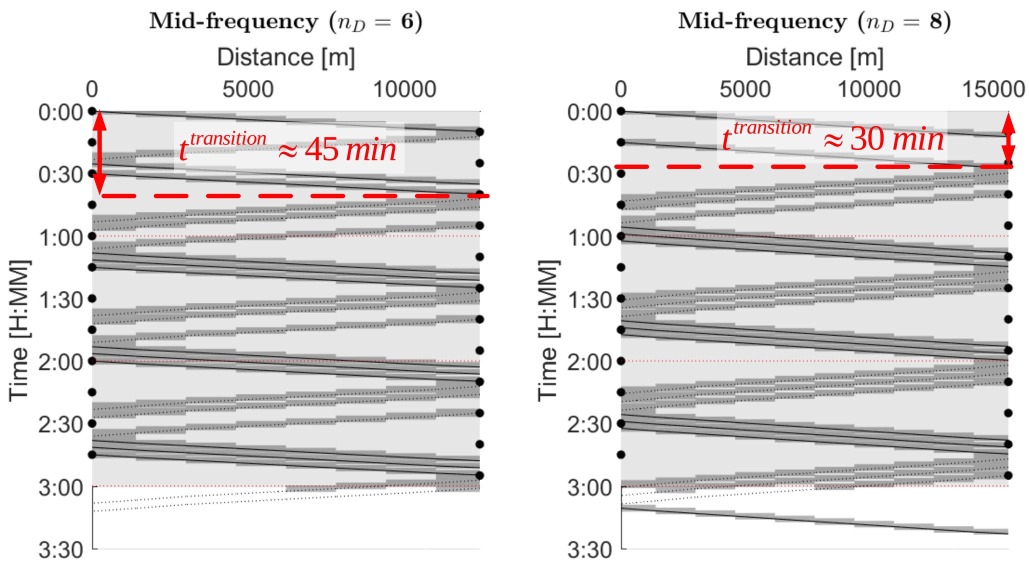


Figure 6.9: Time-distance diagrams for scenarios with six, i.e. scenario B2, (left) and eight (right) block sections in segment D, on the first AIP

Figure 6.5 (right) reported the time-distance diagram for scenario A10 which is equivalent to scenario B6. No clear conclusion on batch size could be drawn: a large batch with four trains (around 1:00) is followed by two smaller ones. There seems to be a tendency for larger batches along time, which may indicate that the previously identified $t^{transition}$ would only occur for longer scheduling horizons. The time-distance diagram for scenario B7 ($n_D = 6$) in Figure 6.10 (left) displays another kind of pattern: a steady increase in batch size. Differences with scenario B6 ($n_D = 3$) are attributed to the longer running times, resulting in later start times for the batches, and more trains queuing.

For scenario B7, optimality of the solution could not be proven within 900 s of computation time (shown in Figure 6.10, right). However, the MIPfocus settings rendered a slightly better solution than the one after 900 s: objective function value decreased with 2% (from 36,049 to 35,390) and average delay from 1,240 s to 1,222 s. Figure 6.10 shows strongly differing timetable structures explained by one fundamental change, indicated in red. First processing a train in direction 1 in the 900 s time limit-results, allowed a faster increase in batch size, but did not improve performance. Only 47 out of 1,260 order variables differed in value, illustrating how close solutions can be situated.

Note that the difference between the maximum and average delay value reported for scenarios B5 to B8 in Table 6.5 increased with increasing n_D, pointing at delays showing a higher variety in size. Batch sizes increased between scenarios B6 and B7, meaning early arriving trains got higher delays than the later ones of the same batch. Hence, the number of large delays increased, whereas the smaller ones remained

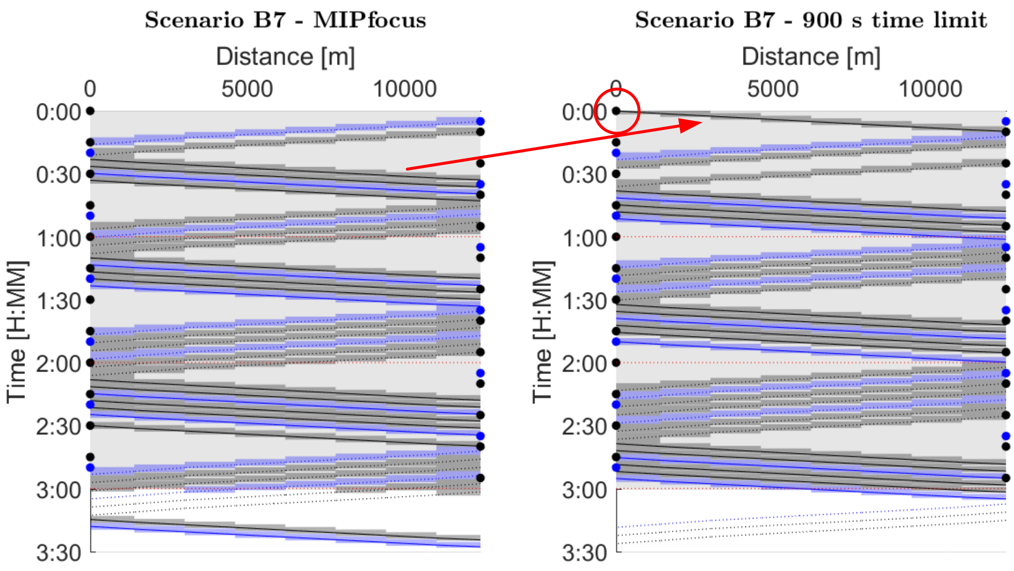


Figure 6.10: Time-distance diagrams for scenarios B7 on the first AIP for results obtained using MIPfocus settings with a time limit of 120 s (left), and regular search settings with a 900 s time limit.

similar in size.

Finally, a higher frequency was expected to lead to a higher number of cancellations. The increased computation times can be interpreted as a sign that train cancellation became more interesting. However, only when the number of block sections exceeded ten, cancellation was deemed necessary.

6.4.3 Conclusion

As increasing the length of the segment D results in increasing running times, also set-up times between trains in opposite directions increase. Although it is beneficial to form larger batches, this seems to arise only after some time to ensure sufficient trains are queuing. After this transition period, regular original timetables resulted in regular batch sizes. Increasing n_D had two possible effects: either the batch sizes increased, or the transition time decreased. Such results were found for multiple scenarios, supporting the conclusion that this is fundamental. Additionally, this hampers the selection of case-specific parameters for the (max,wait) heuristic as no indication is given on which of both parameters should be altered.

These observations could be explained by a snowball effect: increasing running times, leaves more trains waiting at the other side. Following the conclusions in Section 6.2.3, these all join the next batch, which again increased process times and the number of trains queuing.

Although the original high-frequency timetable also held some regularity, i.e. it is repeated every hour, fixed patterns were not found. Therefore, the conclusions for the regular timetables seem not to be valid anymore. Nonetheless, the snowball effect could still be observed, but batch sizes were not regular at all.

As no trains got cancelled for any of the realistic scenarios, its effects have not been evaluated. Therefore, one has to keep in mind that cancellation of trains may distort such patterns. A plausible expectation is that the patterns will be re-established some time after the cancellation.

6.5 Disruption speed v_D

Next to the number of block sections in segment D, also disruption speed v_D influences running times. After passing the exact location of the disruption, acceleration to a higher speed may be allowed [30]. Making abstraction of this exact location within segment D, the average speed over the whole segment is used as input to the model. As a speed decrease comes down to a running time increase, similar results as for increasing n_D are expected: increasing batch size and more train cancellations. Remember that the latter was not needed for most scenarios in Section 6.4.

Three different levels for v_D are assessed: 40 km/h, being the absolute maximum speed to be able to stop-on-sight, 80 and 100 km/h. None of the other parameters mentioned in Section 6.1 were altered. Both the mid- and high-frequency timetables have been assessed for all three speed limits. Table 6.7 reports delay statistics, number of cancelled trains, and associated objective function values for results obtained with MIPfocus settings, together with their computation time. With decreasing v_D, computation times increased as cancellation became more interesting. Solutions for scenarios C5 could not be proven to be optimal. Finally, objective function values for the results obtained by both heuristics, and their relative increases are reported.

6.5.1 Mid-frequency timetable results

Table 6.7 indicates train cancellations were not needed for the mid-frequency timetable, which led to strong increases in total delay (+300% for scenario C1 compared to scenario C3). The FIFO heuristic performed much worse for any of the speed restrictions. In scenario C1, delays quickly increased and five trains had to be cancelled. Whereas its parameters were fine-tuned for scenario C2, which is equivalent to scenario A3, the (max,wait) heuristic managed to find the optimal solution to scenario C3. On the other hand, it led to worse results for scenario C1, although still performing much better than the FIFO heuristic, e.g. only two trains had to be cancelled.

Timetable regularity led to the tendency to form larger batches, which were equal in size after a transition time for scenario in Section 6.4. For the most severe speed restriction (Figure 6.11, left), similar results appeared. After approximately 1 h, all batches had four trains running in the same direction after each other. The end of the scheduling horizon raised an exception, attributed to trains operating after t_{end} not being incorporated in the model's timetable. Such observations could also be made for scenario C2 (Figure 6.11, right). Similar to Table 6.6 in Section 6.4, batches were smaller and $t^{transition}$ decreased for decreasing running time, i.e. increasing v_D.

Table 6.7: Description of each scenario when varying disruption speed v_D. Solution statistics report the maximum, total and average delay values, together with the number of cancellations, constituting the weighted objective function. Computation times for the MIPfocus settings are reported. An asterisk (*) is used to indicate scenarios for which results are the same as those obtained within 900 s and proven optimal. Finally, the last columns report the objective function value obtained by respectively the FIFO and (max,wait) heuristics, and their relative increase compared to the model's results.

#	Timetable (trains/h /direction)	Disruption speed v_D (km/h))	Max. delay (s)	Total delay (s)	Average delay (s)	Number cancelled	Objective function value	CPU time (s)	FIFO		(Max, wait)	
									Absolute value	Relative increase	Absolute value	Relative increase
C1		40	2,815	33,776	1,407	0	23,168	49.9	76,726	231%	54,058	133%
C2	Mid (4)	80	1,004	10,985	458	0	7,133	0.7	22,715	218%	7,241	2%
C3		100	911	8,140	339	0	5,404	0.8	9,625	78%	5,404	0%
C4		40	3,817	49,601	1,600	5	71,936	120.1	145,592	102%	103,776	44%
C5	High (6)	80	1,604	25,105	697	0	19,327	120.1*	79,823	313%	20,030	4%
C6		100	1,176	16,096	447	0	11,992	33.7	52,704	339%	11,992	0%

For scenario C3, mixed results were obtained. Instead of a fixed size, batches in the same direction alternated between one and two trains. This can be explained by the off-set between the departures in both directions: trains in direction 1 leave 10 min after those in direction 0, not exactly in-between them. Hence, for the one batch, it was worthwhile waiting for a second train, or there was already one queuing, for the next one, it was not.

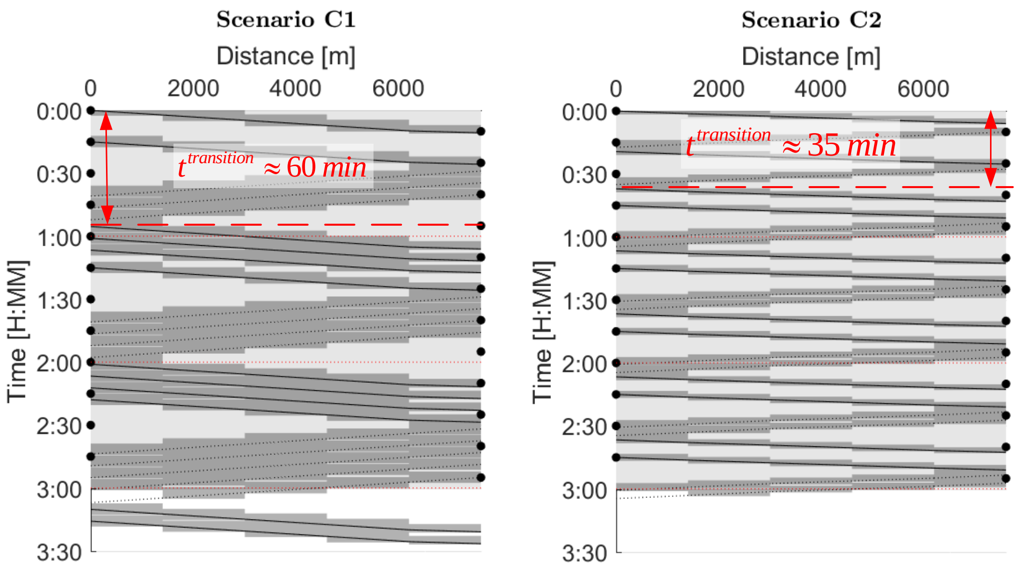


Figure 6.11: Time-distance diagrams for scenarios C1 (left) and C2 (right) on the first AIP.

6.5.2 High-frequency timetable results

As reported by Table 6.7, five trains had to be cancelled for scenario C4 with $v_D = 40\ km/h$. The time-distance diagram in Figure 6.12 (left) indicates the causes. First of all, trains running in both directions required a long running time, increasing the probability that a next train arrived and thus batch sizes increased. Already near the start of the disruption, a number of trains started queuing at the left end. For the first ones, delays increased too much, and they had to be cancelled, similar for the next batch in direction 1. Although batch sizes did not necessarily decrease, cancelling trains was no longer beneficial towards t_{end}. Less trains still had to be operated, reducing the induced delays. Following the large batch sizes, both directions experienced larger gaps in-between services. Hence, Constraints (5.13) and (5.14) may be used to limit the batch size and indirectly reduce the gaps. Considering cancellation of trains only near the end, both the FIFO and (max,wait) heuristic performed much worse and cancelled respectively eight and seven trains.

The best found solutions for scenario C4 differed for each computational setting.

Figure 6.12 shows the time-distance diagrams for the results obtained with MIPfocus settings (left) and with a 900 s time limit (right). Although the former performed only 0.4% worse in terms of objective function value, they differed in the distribution of train cancellations per direction. The coloured circles indicate the same trains for both timetables, from which one observes the decision to cancel or not changed for each "group". This resembles a strong difference in decision variables, e.g. 285 order variables (out of 1,260) were altered.

6.5.3 Conclusions

The impact of disruption speed v_D on the obtained solutions is very much in line with observations for increasing length of segment D, as both variations boils down to influencing running times over the disrupted area. Although not observed before, regularity in batch sizes can also reveal as alternating between two different sizes. To conclude, this is likely due to the ratio of headway times in the original timetable and running times over the disrupted area.

For high-frequency timetables, cancellation may become a beneficial measure in case of severe speed restrictions. As expected before (see Section 6.4.3), these cancellations might distort the snowball effect on batch sizes. Additionally, results (again) illustrated the main benefit of the model over both heuristics: cancellation of a train is balanced against its induced delay to all other trains rather than only its own delay.

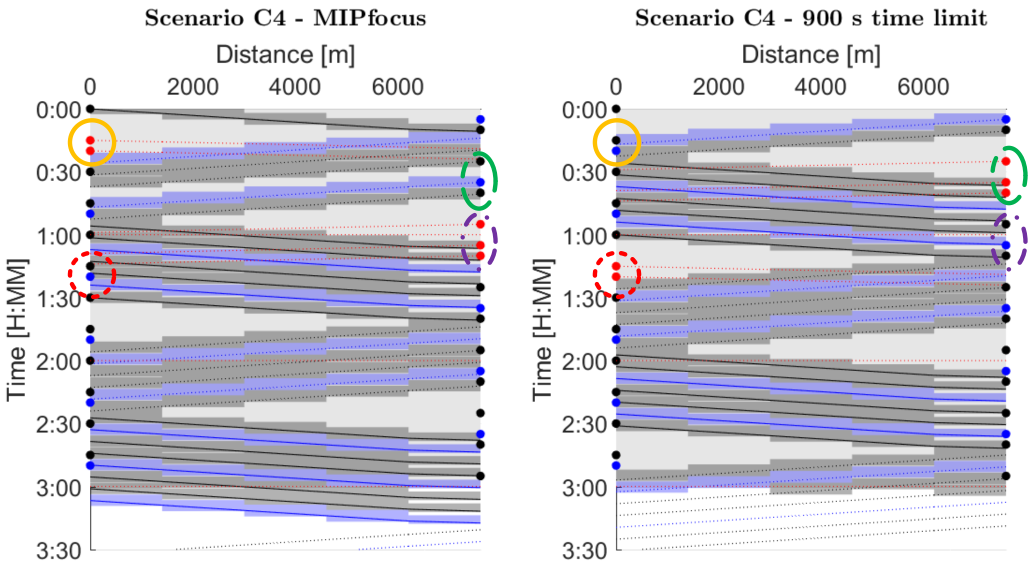


Figure 6.12: Time-distance diagrams for scenarios C4 on the first AIP for results obtained using MIPfocus settings with a time limit of 120 s (left), and regular search settings with a 900 s time limit.

Speed over segment D was assumed to be the same over its full length, which is not in line with actual practice, and may lead to conflicts following differing running and blocking times. However, this could be avoided if the dispatcher also provides information on the exact location, i.e. block section, of the broken train. One also has to keep in mind that these restrictions very much depend on the nature, e.g. a train breakdown, of the disruption itself [30].

6.6 Unbalances in passenger numbers

This section assesses the impact of unbalances in passenger numbers for both directions. For the model itself, this comes down to cancellation and delay penalties differing by direction, as expressed by Equation (5.19). Due to its formulation, absolute passenger numbers are not required. Four scenarios are assessed: a base one has the same number of passengers in both directions. Next, scenarios with twice and fivefold the number of passengers in direction 1 are considered, i.e. $\frac{pax_1}{pax_0} = 2$ or 5. Scenarios D4 and D8 are more extreme with $\frac{pax_1}{pax_0} = 10$. The latter may occur during peak hours on lines connected with Brussels.

Expectations are that train cancellations are only considered in direction 0. Secondly, one can expect higher batch sizes for direction 1: it is more beneficial to wait a bit longer for a train that is not queuing yet, so it can join the batch. Saving a double, five- or tenfold delay penalty, outweighs delays incurred to trains in the other direction. Table 6.8 shows the relevant statistics for the best found solutions. Note that in the results, the objective function value will decrease with increasing ratio $\frac{pax_1}{pax_0}$: penalties are no longer all integer, which can lead to fractional objective function values. As the heuristics did not account for passenger numbers, their disruption timetables did not change. The differences in objective function value score result solely from recalculations with the adjusted penalties.

One could adjust the FIFO approach to include relative passenger numbers by prioritizing the more crowded trains similar to international trains (see Section 5.6.1): cancelling them is not possible, and their delay is limited by a fraction of the cancellation threshold D^{cancel}. For example, if $\frac{pax_1}{pax_0} = 5$, D^{cancel} decreases to 20% of its original value. The (max,wait) heuristic could differ in N^{max}_{batch} between both directions: the maximum size increases for the more crowded direction, whereas it decreases for the other direction. These extensions have not been considered here.

A first observation is the increase in computation time with increasing unbalance in passenger numbers, notwithstanding the increased difference, i.e. penalties, between trains. MIPfocus settings rendered the best results for all scenarios. Those obtained after 900 s with regular settings either confirmed optimality (D5 and D6), or performed worse (D7 and D8).

Table 6.8: Description of each scenario when varying passenger numbers. Solution statistics report the maximum, total and average delay values, together with the number of cancellations, constituting the weighted objective function. Computation times for the MIPfocus settings are reported. An asterisk (*) is used to indicate scenarios for which results are the same as those obtained within 900 s and proven optimal. Finally, the last columns report the objective function value obtained by respectively the FIFO and (max,wait) heuristics, and their relative increase compared to the model's results. Although their associated disruption timetables were the same, objective function values were recalculated using the adjusted penalties.

#	Timetable (trains/h /direction)	Ratio $\frac{pax_1}{pax_0}$	Max. delay (s)	Total delay (s)	Average delay (s)	Number cancelled	Objective function value	CPU time (s)	FIFO		(Max, wait)	
									Absolute value	Relative increase	Absolute value	Relative increase
D1	Mid (4)	1	1,004	10,985	458	0	7,133	0.6	22,715.0	218%	7,241.0	2%
D2		2	1,004	10,985	458	0	4,704	0.6	16,769.5	257%	6,118.0	30%
D3		5	1,004	11,471	478	0	3,112	0.8	13,202.2	324%	5,444.2	75%
D4		10	1,904	13,271	553	0	2,519	1.8	12,013.1	377%	5,219.6	107%
D5	High (6)	1	1,604	25,105	697	0	19,327	120.1*	72,623.0	276%	20,030.0	4%
D6		2	1,950	25,801	717	0	14,096	120.1*	55,524.5	294%	15,505.0	10%
D7		5	2,204	28,700	797	0	9,808	120.1	45,265.4	362%	12,790.0	30%
D8		10	3,467	29,855	878	2	6,791	120.1	41,845.7	516%	11,885.0	75%

6.6.1 Mid-frequency timetable results

Solutions found for scenarios D1 and D2 were exactly the same, i.e. the difference in passenger numbers was not sufficient to justify changes in the orders of trains. The decrease in objective function value is attributed to changed weights. The resulting time-distance diagram is shown in Figure 6.13 (left). Expectations were that batch sizes in direction 1 would increase at the expense of additional delays in the other direction. However, one would have to wait a long time for adding a third train to a batch in direction 1, as indicated in red. The additional high delays for direction 0, outnumbered the possible gains of doing so.

Although increasing the ratio to 5 in scenario D3 (Figure 6.13, right) rendered it more beneficial, rather limited effects were observed. The trains indicated by the orange dashed arrows illustrates how a train in direction 1 got priority over the first one in direction 0. For the remainder of the timetable, the same argumentation as for scenario D2 goes up: it was not beneficial to wait for an additional train. Total delay increased compared to scenario D1, indicating that the increase in delay for direction 0 (+ 8 min 6 s) was larger than the absolute savings for direction 1 (- 4 min 49 s). Nonetheless, this measure was beneficial due to the differing penalties.

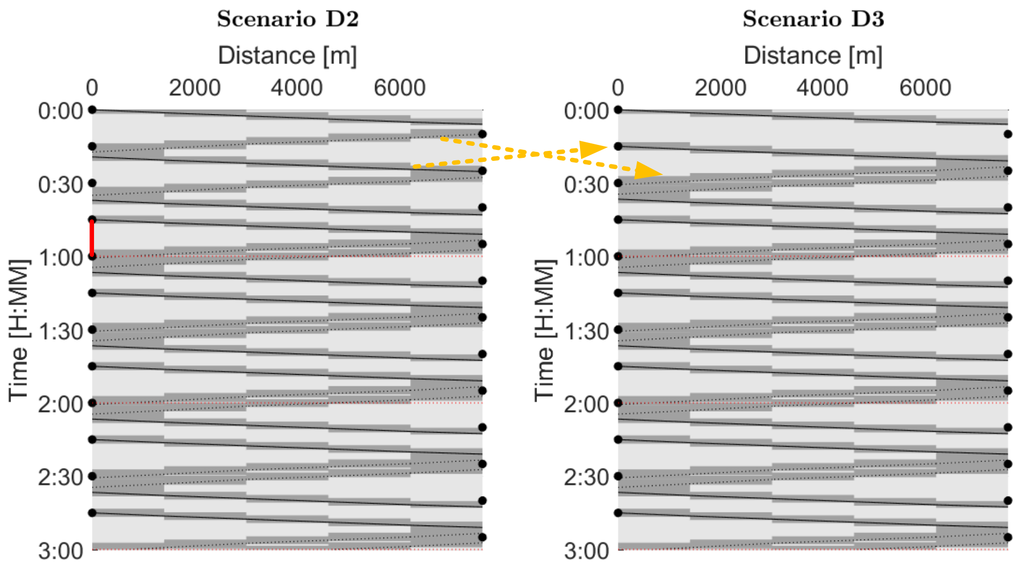


Figure 6.13: Time-distance diagrams for scenarios D2 (left) and D3 (right) on the first AIP.

6.6.2 High-frequency timetable results

Solving the high-frequency scenarios appeared to be more difficult, indicating that cancellation of a train in direction 0 became a plausible option to reduce delays for direction 1. However, it was not conducted for any of the scenarios, except the highly unbalanced one (scenario D8). Other than for the mid-frequency scenarios, differences between scenarios D5 (equivalent to A10) and D6 are expected: trains arrive at a higher frequency, making it worthwhile to wait for the next one in direction 1. Figure 6.5 (right) and Figure 6.14 (left) show the time-distance diagrams associated with scenarios D5 and D6 respectively. For the latter, the main differences are encircled in red: trains in direction 0 got assigned to the next batch despite the fact the first one started queuing during the processing of a batch in its own direction. These observations contradict conclusions from previous experiments, meaning those “rules” no longer amount when passenger numbers differ significantly. Scenario D7 led to similar observations, although changes occurred earlier in the timetable.

Train cancellation only came into play upon increasing the ratio to 10 (scenario D8). The time-distance diagram in Figure 6.14 (right) illustrate how delays in direction 1 remained very limited, whereas trains in direction 0 were only operated if they fit well in-between the batches in direction 1. Up to six trains queuing at the right end of the corridor (orange dashed circle), which all had to be processed after t_{end}. Hence, the assumption of infinite buffer capacity at both ends of the corridor may become violated. In such situations, trains should be stored further downstream of the disrupted area.

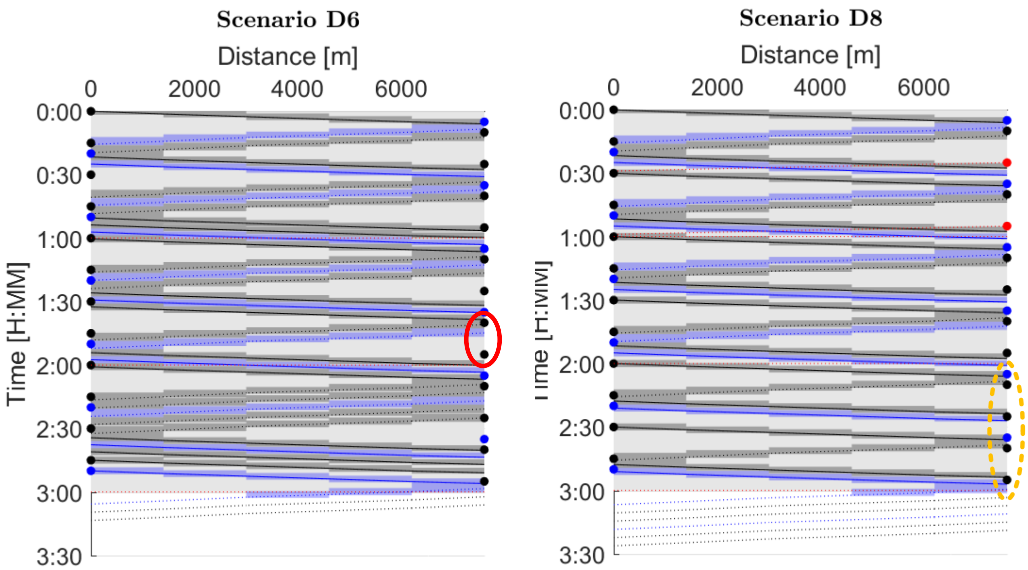


Figure 6.14: Time-distance diagrams for scenarios D6 (left) and D8 (right) on the first AIP.

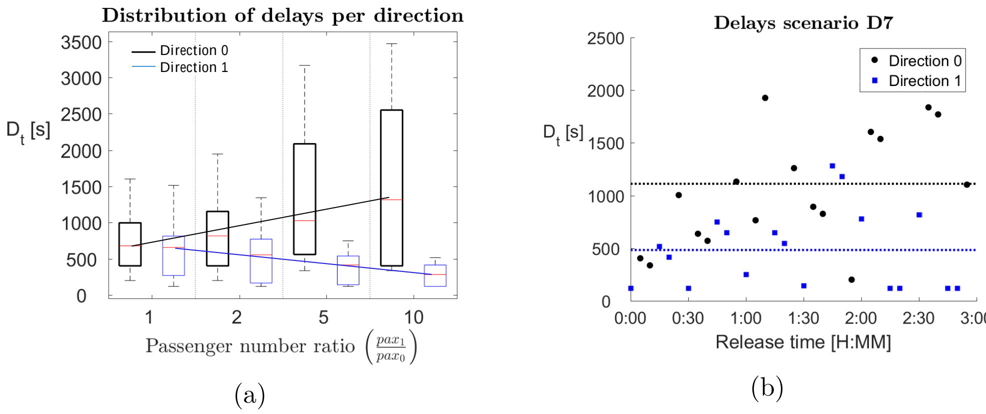


Figure 6.15: (a) Distributions of delays per direction for increasing number of passengers in direction 1 $\left(\frac{pax_1}{pax_0}\right)$ for the high-frequency scenarios (D5-D8). (b) Delays per train for scenario D7. Thick black lines (in (a)) or circles (in (b)) represent direction 0, similar to thin blue lines or squares for direction 1.

One could also investigate the delays incurred to passengers in both directions. Due to changed weights, higher absolute delays are expected for the less crowded direction 0. Figure 6.15a shows box plots of delay values per direction and scenario. Initially, both directions had about the same average, minimum and maximum delays (scenario D5). With increasing passenger numbers, the delays in direction 1 decreased at the expense of higher delays for direction 0. For the latter, the difference between the average and maximum delays increased for similar reasons as before (Section 6.4.2). The increase in delay statistics in Table 6.8 prove delay savings in direction 1 did not outnumber delay increases for direction 0.

Figure 6.15b differs between delays for both directions in scenario D7: those for the less crowded direction increased in size along time as priority is given to trains running direction 1. Hence, batches in direction 0 got interrupted to process a batch in direction 1 to mitigate its delay increases.

6.6.3 Conclusion

Accounting for passenger numbers boils down to applying differing penalties for the models developed in this thesis. As a result, some trains become prioritized over others, which may lead to the violation of "rules" identified in Section 6.2.3. Two effects were identified. First of all, waiting longer for an extra train to add it to the current batch, seemed to become more attractive with increasing passenger numbers. As a consequence, total (unweighed) delay may increase. Secondly, batches in the less crowded direction may get interrupted prematurely in case of strong unbalances. Trains in the less crowded direction are only operated if they could fit without incurring delays. Such measures are in line with current practice at Infrabel [30].

Although relative passenger numbers have been defined per direction, they could also differ over time for the same direction, e.g. when the scheduling horizon starts within peak hours, but ends after it. One expects similar observations, e.g. breaking batches when a (very) crowded train has to pass. Possibly, also changing orders in the same direction becomes beneficial.

6.7 Buffer times to stabilize operations

Before, disruption timetables included no or little buffer between the blocking time stairways of two subsequent trains. Two major requirements have to be satisfied to put such a schedule in practice. First of all, trains have to be queuing upon their scheduled starting time, i.e. they can have only limited delays when arriving at either end of the corridor. Secondly, trains have to strictly adhere to their assigned path and variations in process times are not allowed. For example, if a train driver at one of both ends reacts too slow to a signal clearing[3], he enters the corridor later. In the absence of buffer time, this may induce a knock-on delay to its successor(s).

To alleviate these assumptions and to make the schedule more robust to deviations, buffer times can be added at the cost of operating less trains, or increasing the (scheduled) delays D_t. Here, two strategies to add buffer times between trains are assessed. Set-up times s_{ij} obtained with Equations (5.1) and (5.2) can be adjusted as follows:

$$s_{ij} \leftarrow \begin{cases} s_{ij} + B_{same} & \text{if } dir_i = dir_j \\ s_{ij} + B_{diff} & \text{if } dir_i \neq dir_j \end{cases} \tag{6.1}$$

A first strategy adds some buffer time (B_{same} and B_{diff}) between any pair of two trains i and j, regardless of their directions dir_i and dir_j, i.e. $B_{same} = B_{diff}$. Another approach is to schedule buffer time only between trains in opposite directions ($B_{diff} > 0$ and $B_{same} = 0$), allowing to some extent the knock-on effect between trains within the same batch. Hence, delays arising are transferred to subsequent trains within the batch, but are less likely to affect the next batches.

Table 6.9 reports the statistics for five scenarios per timetable: the basic one without buffers, two scenarios with either a short or long buffer time for the first strategy and two scenarios for the second strategy. After reporting delay and solution statistics, a next column reports the *robustness cost* (c_{rob}), which is interpreted as the relative increase in objective function value compared to the scenarios without buffer. Finally, objective function values for the FIFO and (max,wait) heuristics are reported.

Note that the order of trains did not change with the FIFO heuristic, which resulted in large delay increases and up to four and ten cancelled trains for scenarios E5 and E10 respectively. On the other hand, the (max,wait) heuristic considered the set-up times between trains to assign them to batches, meaning that additional trains might have joined with increasing buffer times. As such, it is able to deal with small buffer times relatively well, except for scenarios E3 and E5 as batches of three trains were not allowed for the mid-frequency timetable ($N_{batch}^{max} = 2$).

[3] Current dispatching at Infrabel assumes "perfect" driver behaviour [30].

Table 6.9: Description of each scenario for two possible strategies to stabilize operations by means of varying buffer times B_{same} and B_{diff}. Solution statistics report the maximum, total and average delay values, together with the number of cancellations, constituting the weighted objective function. Computation times for the MIPfocus settings are reported. An asterisk (*) is used to indicate scenarios for which results are the same as those obtained within 900 s and proven optimal. Finally, the last columns report the objective function value obtained by respectively the FIFO and (max,wait) heuristics, and their relative increase compared to the model's results.

#	Timetable (trains/h /direction)	Buffer time (min) same / diff	Max. delay (s)	Total delay (s)	Average delay (s)	Number cancelled	Objective function value	CPU time (s)	c_{rob} (%)	FIFO		(Max, wait)	
										Absolute value	Relative increase	Absolute value	Relative increase
E1		0 / 0	1,004	10,985	458	0	7,133	0.6	-	22,715	218%	7,241	2%
E2		1 / 1	1,064	12,965	540	0	9,113	0.7	28%	39,275	331%	10,661	17%
E3	Mid (4)	2 / 2	1,401	16,489	687	0	12,637	6.9	77%	57,206	353%	26,241	108%
E4		0 / 3	1,461	15,598	650	0	11,746	1.7	65%	66,372	465%	17,661	50%
E5		0 / 5	1,581	18,358	765	0	14,506	3.1	103%	74,877	416%	29,899	106%
E6		0 / 0	1,604	25,105	697	0	19,327	120.1*	-	79,823	313%	20,030	4%
E7		1 / 1	2,514	37,600	1,074	1	39,223	120.1	103%	113,240	189%	45,806	17%
E8	High (6)	2 / 2	2,951	42,267	1,281	3	58,530	120.1	203%	132,597	127%	87,339	49%
E9		0 / 3	2,384	37,802	1,050	0	32,024	120.1	66%	127,581	298%	46,264	44%
E10		0 / 5	2,531	44,111	1,225	0	38,333	120.1	98%	133,303	248%	55,996	46%

6.7.1 Mid-frequency timetable results

Table 6.9 reports a 28% increase in objective function value when adding a small buffer of only 1 min. The time-distance diagram of scenario E1 (equivalent to A5) contained already some buffer time both within and between batches (Figure 6.3, right). Hence, results for scenario E2 did not show alterations in batch size, but rather a redistribution of train entry along time.

Increasing these buffer times to 3 min (scenario E3), had a c_{rob} of 77% and resulted in batches with three trains as the time-distance diagram in Figure 6.16 (left) shows. Similar to observations in Sections 6.4 and 6.5, the regularity of the original mid-frequency timetable resulted in train batch sizes with the same number of trains after 40 min. Adding buffer time for any pair of trains in scenarios E2 and E3 boiled down to increasing the set-up times between them and had similar effects as increasing the running time over the disrupted area.

Running times for trains over this short stretch varied between 358 and 439 s. Hence, the buffer times in scenarios E2 and E3 represented average set-up time increases of about 15 and 30% respectively. The former is acceptable, the latter seems to be exaggerated.

Due to the small batch sizes, adding buffer time between trains in opposite directions should favour the formation of larger batches: an additional train may arrive during the increased set-up time. Both scenarios E4 and E5 confirmed these expectations. In the former scenario, one train in direction 0 got assigned to an earlier batch, and a constant batch size got established sooner than in scenario E1. Batch sizes did not change from scenario E4 to E5, and the additional 2 min of buffer

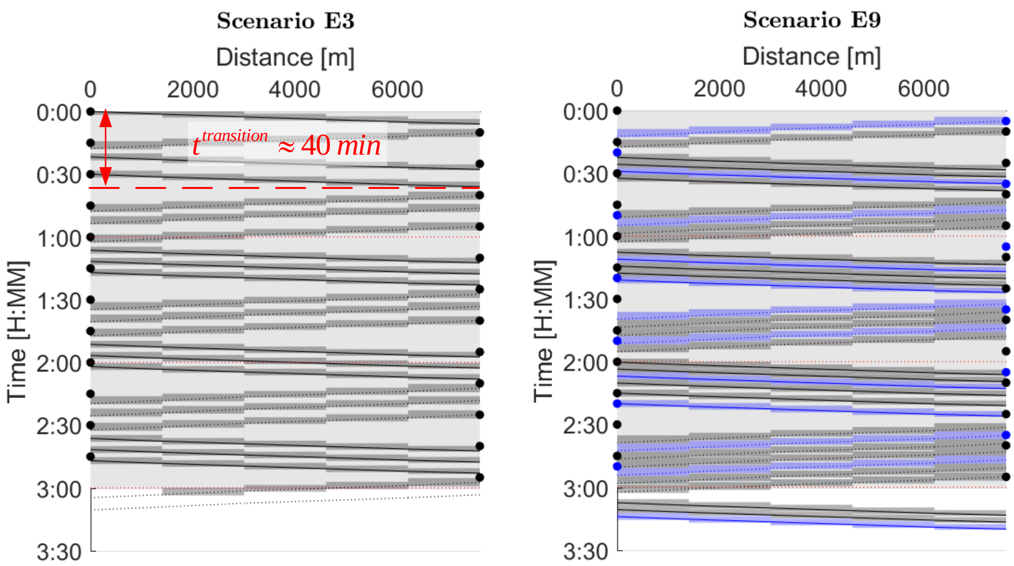


Figure 6.16: Time-distance diagrams for scenarios E3 (left) and E9 (right) on the first AIP.

time (BUF_{diff}) added just less than 2 min to the average delay. In light of the large headways between trains in the original timetable, small buffer times should be sufficient to absorb process variations and (partially) avoid knock-on delays.

6.7.2 High-frequency timetable results

With increasing frequency, the effect of adding buffers to stabilize operations becomes more important: trains follow each other faster and delays may propagate more easily.

In contrast to scenario E2, batch sizes increased from scenario E6 to E7. Opposite from observations in scenario E2 and E3, this was not to establish a fixed batch size. The next batch in the other direction started later which resulted in train cancellation. Additionally, this decision resulted in maximum batch size to increase from five to eight, in scenarios E6 and E7 respectively. With increasing buffer time in scenario E8, both effects were amplified: two additional train cancellations were needed and more batches with eight trains existed. Note that the cost of robustness almost doubled. Hence, the timetable irregularity and the snowball effect had a similar impact as before (Section 6.4.2).

Adding a buffer only after the end of a batch in scenarios E9 (displayed in Figure 6.16, right) and E10 led to higher total delays. Although batch sizes increased to similar extents as for scenarios E7 and E8, train cancellation was not required. In scenarios E7 and E8, the first train of a batch contributed to additional delays for all following trains. On top of that, it had already the largest delay within the batch and cancelling it saved both contributions. For scenarios E9 and E10, trains within a batch were allowed to follow each other directly, and only the last train induced delays to subsequent batches as only this one required the additional buffer time.

Moreover, the average headway decreased, as the buffer times of 3 and 5 min in scenarios E9 and E10 could be spread out over all trains within the previous batch rather than having a fixed value of 1 or 3 min for each train in scenarios E7 and E8 respectively. On the other hand, an increasing batch size also means that less time per train is available to absorb delays, which may reduce stability. The average remained about 12.5 ($B_{diff} = 3\ min$) or 37.5 s ($B_{diff} = 5\ min$) per train, or about 8 to 10% of total running time for the latter. Nevertheless, it deemed better than excessive times of 3 min per train as in scenario E8.

6.7.3 Conclusion

The first strategy, which added buffer times between any pair of trains, is highly similar, if not equivalent, to increasing running times over the sections. Again, for regular timetables, this may establish constant batch sizes. On the other hand, adding buffer times only after a batch (strategy 2) seems to reward the formation of larger batches by reducing average headway times. Moreover, such a strategy does not increase the induced delays of trains earlier in a batch.

From the above discussion, the first strategy seems to be preferable for mid-frequency timetables or cases in which the dispatcher expects small batches. On the other hand, for higher frequencies, the second strategy might be more interesting as disruption timetables mostly contain larger batches. The added buffer times could be shared over a larger number of trains, thereby reducing average minimum headway time.

Although the robustness cost seems to be rather high, it has to be kept in mind that the scenarios evaluated timetables without external delays. In case these would arise after generating the disruption timetable, one may expect better performance by those disruption timetables which hold buffer times, as these buffers mitigate the transfer of delays from one train to the other. The exact effect depends of course on the magnitude of the introduced delays. Such evaluations were out of the scope of this thesis.

6.8 Conclusion

This chapter assessed the impact of several parameters of the model for the first AIP, on the generated disruption timetables. An artificial case study allowed to draw clear conclusions, and compare scenarios across different sets. Moreover, it allowed to assess parameter impact for differing timetables, a regular and an irregular one. Timetable structure seems to be the most important factor. Hence, one could advocate it is not possible to derive general rules. However, conclusions drawn for perfectly regular timetables may be expected to hold also for less perfect ones, i.e. with small deviations compared to regular headways (e.g. trains at 0:00 and 0:35 instead 0:30). For these regular timetables, fixed patterns arose concerning batch size. Results support the conclusion that this could be a fundamental property.

Initial results showed that two main rules could be identified in case trains have equal penalties. One, queues of trains at either end should be emptied during the next batch in this direction. Two, it may be beneficial to wait a certain time on the next arriving train before ending a batch. However, deciding on whether to wait or not, is not straightforward and one expects dispatchers also not to take this into account. For strong unbalances in penalties among trains, e.g. differing passenger numbers, these rules may become violated, and prioritizing crowded trains if one does not have to wait too long for it, is considered good practice. For extreme cases, the model resembled current dispatcher's practice. Nonetheless, it is expected to perform better by also allowing minor delays for the prioritized trains.

Because it incorporates similar rules, the (max,wait) heuristic outperformed the FIFO one. For certain sets of parameters, it got close to the models' performance. However, the waiting aspect is considered to be its main deficiency. Achieving good performance without fine-tuning would again require a lot of experience from the dispatcher. Moreover, the heuristic fails to account for cancellation.

Parameters such as the length of segment D and disruption speed v_D, directly influence running times and are regarded as being the main influencers next to timetable structure. Increasing running times seem to favour larger batch sizes, which is rather intuitive.

Finally, buffer times could be an instrument to stabilize real-life operations. For lower frequency timetables, small values should be added after each train. Its effects may be similar to those of increasing running times. Timetables with a higher frequency seem to require a distinct approach as they are expected to result in larger batch sizes. Hence, (larger) buffer times should only be added after each batch. In a practical set-up, dispatchers could be allowed to select a certain strategy, based on these rules-of-thumb, before running the model.

Chapter 7

Partial Blockage of a Double-Track Corridor with Stops

Chapter 5 developed a model to generate disruption timetables for the first AIP, which exhibits a strong flexibility to model a wide range of situations with a small number of parameters. Most occurrences of this first AIP do however have stops along them, as shown on a technical map of the network [26]. In theory, the model could also be applied directly without considering stop-skipping. The latter simplification might hamper serious level of service improvements. Therefore, this chapter extends the model by explicitly incorporating stops and stop-skipping.

First, Section 7.1 describe the general configuration of the second AIP, thereby introducing the practical case study. Based on this description, several required extensions to the model framework could be identified. Section 7.2 discusses them, and adapts the original optimization model formulation, which becomes a single-machine scheduling problem with rejection and controllable process times. The flexibility of the extended model is illustrated by application to the Oostkamp-Aalter case study in Section 7.3. Similar as Chapter 6 did for the first AIP, Section 7.4 assesses the impact of the newly introduced parameters on the generated disruption timetables by means of an artificial case study. Finally, Section 7.5 presents the main conclusions.

7.1 Archetypical infrastructure piece description

On the corridor between two larger stations, also one or multiple open-track stop(s) could be present. The works by Törnquist and Persson [54] and Zhan et al. [60] consider stop-skipping as a measure in case of disruptions. Abril et al. [1] point out that introducing commercial stops decreases the number of trains that can be operated on a single-track corridor. Conducting a stop includes braking before the stop, a minimum dwell time allowing passengers to board and alight and re-acceleration, thereby increasing the running and blocking times.

Figure 7.1 represents the starting situation considered for the second AIP, the sole difference with the first AIP being the presence of an open-track stop o_1 along the segment D. In the remainder, O represents the set of (open-track) stops along the disrupted area. Skipping planned stops outside the disrupted area is difficult to justify towards passengers boarding or alighting at those stops: although their destination is not located within the disrupted area, they would be affected. Therefore, trains may only skip stops located within the disrupted area.

Investigating the technical map of the network [26], a similar but fundamentally different AIP including stops is commonly encountered: corridors with stops “within the switches”. Figure 7.2 illustrates the idea: in-between the signals delineating segment S_1, a stop o_2 is present. At the end of the platforms and in both directions, additional signals are present, indicated in bold. These can only show aspects for trains travelling in one direction, and may split segment S_1 in two block sections, which could allow a train with a stop to enter at an earlier time, as reserving the second half of the segment S_1 is not necessary yet. Moreover, this second half of the segment is only reserved some time $t_{stop}^{approach}$ before accelerating again, set to 30 s at Infrabel [30].

Assume a blockage within segment D. Whereas for the first AIP and the one in Figure 7.1 the exact track being blocked, was only of minor influence, this no longer holds. Figure 7.2 shows routes for trains in both directions during blockages of the track in directions 1 (top) and 0 (bottom) for a specific configuration of the switches. Extension of the argumentation below towards cases with a stop on segment S_2 and mirrored switches is straightforward.

A blockage of direction 1 results in only one platform at stop o_2 being available during the disruption for through-running trains, and trains in both directions have to stop at the same platform. Hence, this could be considered to be identical to the case in Figure 7.2 with S_1 completely included in the disrupted area. On the other hand, in case the bottom track is blocked, both platforms can be used. Trains in direction 1 can already perform their stop while a train in direction 0 is running over segment D. As a result, segment S_1 can be considered as two separate block sections:

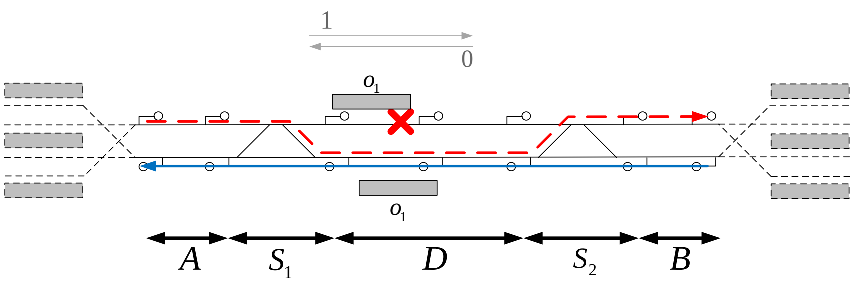


Figure 7.1: Sketch of the second AIP, which extends the first one in Figure 5.1 by including stops such as o_1 along the corridor. In case a disruption occurs on the upper track, trains running in direction 1 (red dashed line) have to be rerouted over the lower track. Those in direction 0 (full blue line) keep to their route.

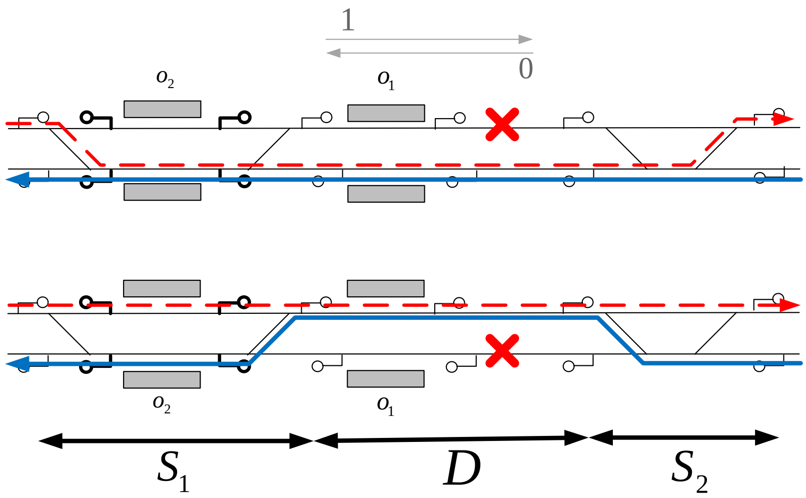


Figure 7.2: Stops on the second AIP may also be located within segments S_1 (as o_2) and S_2. Additional signals (in bold) are present at both ends of the platforms, showing aspects in only one direction, and resulting in differing impact for disruptions on either track. For closures of the upper one (top), trains in both directions have to share the platform at o_2, as opposed to disruptions on the lower track (bottom).

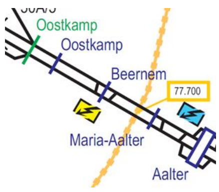
the first half, which includes the stop o_2, is located outside of the disrupted area, the second one lies within it[1]. Due to its prevalence on the network, it is worthwhile to take this extension of the basic AIP in Figure 7.1 into account. Effects on train traffic may depend on the interplay between the blocked direction, and the configuration of the switches.

7.1.1 Case study Oostkamp - Aalter

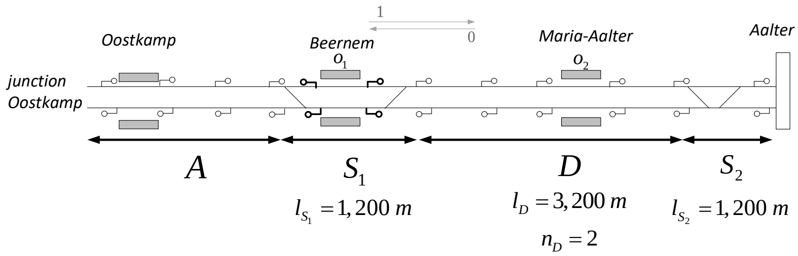
To develop and test the model, the open-track corridor between the junction of Oostkamp and the station of Aalter is considered, shown in Figure 7.3a. Three stops are present along the corridor: Oostkamp, Beernem and Maria-Aalter, located in segments A, S_1 and D respectively. Because of the stop in Beernem, this case study is similar to the situation illustrated in Figure 7.2.

Input data on both the infrastructure and timetable were obtained from the simulation tool developed by Van Thielen et al. [56]. Infrastructure data included information on signal locations for both directions and location of the stops. Figure 7.3b sketches the infrastructure, with segments A and D respectively spanning three and two block sections with an average length of 1,600 m. Segments S_1 and S_2 both have a length of 1,200 m. The stop in Oostkamp does not lie within the disrupted area, cannot be skipped, and is therefore not added to the set O. Stops at Beernem and Maria-Aalter are referred to as o_1 and o_2 respectively. Section 7.3 presents more of their characteristics.

[1]Hence, for a blockage in direction 1, it is possible to skip stop o_2. During disruptions on the other track, it is located outside of the disrupted area, not allowing skipping it any more.



(a)



(b)

Figure 7.3: (a) Technical representation of the Oostkamp-Aalter case study on line 50A, running from Oostende (top left) to Brussels (bottom right), for the second AIP. The considered corridor stretches from the junction of Oostkamp to the station of Aalter. Adapted from [26]. (b) Schematic representation as it is used for the model, including the three stops along the corridor.

Data concerned the 2014 timetable for weekdays from 6:00 until 9:00 in the morning, i.e. during morning peak, and contained for each train data on, amongst others, train type, speed, event timings and scheduled stops. Event timings were only defined for major timetable points such as the junction Oostkamp and the station of Aalter, which were used as entry times for trains in directions 1 and 0 respectively. For stops in-between these locations, scheduled dwell times of 0 s were mentioned, interpreted as trains being allowed to depart from the stops as soon as boarding has finished.

Most trains were assigned to two categories: IC and L trains, with the latter group also incorporating the P trains. Distributions of train speeds showed that IC and L trains had respective maximum speeds of at least 160 and 130 km/h. Braking and acceleration rates were set to respectively 0.375 and 0.6 m/s^2 [8] for both train types. Line speed is assumed to be the limiting factor, allowing trains not to run faster than 120 km/h. Additionally, one Thalys train running in direction 1 was operated between Oostende and Brussels, and had absolute priority, i.e. it cannot be cancelled, nor delayed. International trains were assigned the same characteristics as IC trains for two reasons. One, they run within mixed traffic and their operation has to fit within the timetable. Two, they operate on a non-high speed line with a limited reference speed. As a result, international trains do not have to reserve block sections earlier than IC trains do, and approach times can be modelled as before, but they do have absolute priority.

Direction 1, towards Brussels, has more trains than the other direction (18 against 10), in line with the discussion on demand pattern in Section 2.1. Appendix G reports the timetable data necessary for the model and presents the original time-distance diagrams for the complete corridor. To improve readability, time-distance diagrams in this chapter only show the more interesting segments S_1, D and S_2.

7.2 Model adaptations and extensions

The model framework for the first AIP presented in Figure 5.3 of Section 5.4, can be used for the second AIP with a few extensions. These are indicated in bold within the adapted framework presented in Figure 7.4. The sole new component is the stopping time calculation, which requires input on infrastructure and train characteristics. Following this extension, several other components require adaptations, among which the incorporation of stop-skipping in the optimization model.

The remainder of this section describes the adjustments to the model developed in Chapter 5 in order to cover both the cases with and without stops on segments S_1 and S_2. Several assumptions are made:

1. A train skips all of its stops on the disrupted area or none of them, easing communication towards all actors and passengers. Moreover, it is in line with current practice at Infrabel: a train is either classified as IC or L [30].
2. Only trains t for which a stop at o_s was originally scheduled can either perform a stop or not. Hence, switching stopping patterns between trains (as in [60]) is not considered as a measure.
3. Performing a stop at $o_s \in O$ requires a minimum dwell time $t^{min}_{dwell,s}$. Minimizing total delay means that no train stop lasts longer than this value.
4. A minimum number of trains stopping at a stop o_s each hour may be required in order to ensure its passengers can still be served. (Section 7.2.3)
5. Regardless of the previous restriction, stop-skipping incurs a penalty as it represents a decrease in level of service towards (a share of) the passengers. (Section 7.2.2)

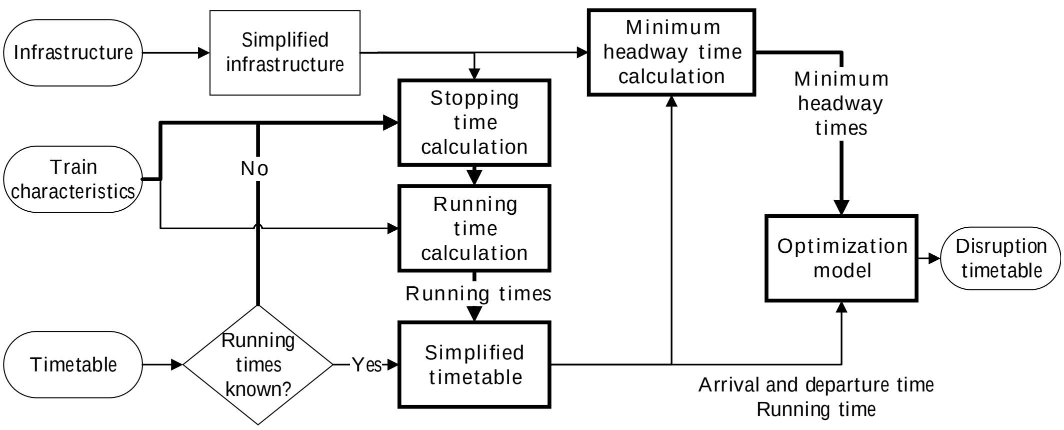


Figure 7.4: Overview of inputs and required pre-processing steps before building the optimization model for the second AIP. Extensions and adaptations compared to the one for the first AIP (Figure 5.3) are indicated in bold.

Each of these elements influences the model presented in Section 5.5, among which running and set-up time calculations. The latter two serve as input to the model, and Infrabel either has them at hand, or uses its own calculation methods. Therefore, Section 7.2.1 only presents the most relevant aspects for the process and set-up time constraints. For a complete discussion on running time, set-up time and delay calculation within this thesis, the reader is referred to Appendix D.

7.2.1 Adjusting process and set-up time constraints

To perform the commercial stop, trains have to brake, dwell, and re-accelerate, resulting in time supplements on top of only running through the segment. Trains either perform all of their stops or none, resulting in two possible process times on machine M^D for those trains with scheduled stops. Let binary decision variables σ_t represent the execution of all commercial stops by train t within the disrupted area. Stop-skipping reduces the running time over block section b from the running time including a stop ($R_{tb}^{stopping}$) to the standard value R_{tb} as determined by the method in Section 5.5.2. Let S represent the set of stopping operations (t, b) with train t performing a stop at stop o_s located in block section b. Process time constraints can be adapted as follows:

$$C_t \geq t_t + p_t(1 - x_t - dev_t) + \sum_{b:(t,b)\in S} \left(\Delta r_{tb}^{stop}\right) \sigma_t \qquad \forall t \in T \qquad (7.1)$$

$$\sigma_t \leq x_t + dev_t \qquad \forall t \in T \qquad (7.2)$$

Constraints (7.1) express that performing the stops ($\sigma_t = 1$) increases the process time without stops (p_t) with a *stopping time supplement* $\Delta r_{tb}^{stop} = R_{tb}^{stopping} - R_{tb}$ per stop location. Note that including the cancellation factor in the last term of Equation (7.1) would lead to non-linear constraints. Hence, Constraints (7.2) state that cancelling or rerouting a train, results in the stop being skipped.

Performing a stop also fundamentally changes the blocking time stairways of the involved train. Each blocking time still consists of all components listed in Section 2.2.2, except for the block section right after the stop: a stopping train only reserves the next block section a time $t_{stop}^{approach}$ before its dwell ends. Using the constraints and methods presented in Section 5.5.2 to model the separation of a pair of trains, may result in a conflict if one of both trains skips its stops. Set-up time constraints have to be adjusted accordingly.

Assume only train i has a scheduled stop. Two levels for set-up times with its non-stopping successor j arise: $s_{ij}^{stopping}$ and $s_{ij}^{disrupted}$ for performing and skipping the stops respectively. Let $\Delta s_{ij} = s_{ij}^{stopping} - s_{ij}^{disrupted}$ represent the (negative) increase in set-up times. Constraints (5.7) and (5.8) are replaced with:

$$t_j \geq C_i + s_{ij}^{disrupted} + \Delta s_{ij}\ \sigma_i - M(1 - q_{ij} + x_i + x_j + dev_i + dev_j) \qquad \forall i, j \in T \qquad (7.3)$$

$$t_i \geq C_j + s_{ij}^{disrupted} + \Delta s_{ij}\ \sigma_i - M(q_{ij} + x_i + x_j + dev_i + dev_j) \qquad \forall i, j \in T \qquad (7.4)$$

Constraints (7.3) and (7.4) ensure that set-up times between trains are adjusted depending on whether train i stops or not. They can be employed for cases in which j is the stopping one by replacing σ_i with σ_j.

Difficulties arise only part of segment S_1 is located within the disrupted area. For the latter, trains in opposite direction cannot have a conflict in the first half of segment S_1, as they operate on different tracks, as opposed to for trains running in the same direction. Additionally, also the case in which two trains following each other have scheduled stops, has to be considered. The reader is referred to Section D.2 of Appendix D for more details.

Finally, the question how the delay D_t of a stopping train has to be calculated, remains. For a non-stopping train, d_t in Constraints (5.5) referred to the adapted departure time, taking into account disruption speed v_D. For a train with a stop, d_t represents the earliest departure including all stops. Constraints (5.6) ensure stop-skipping does not result in $D_t < 0$.

7.2.2 Objective function extension

Train t skipping its stop o_s at block section b severely affects the passengers who had to board on or alight from train t at this location. Benefits may be twofold. One, the delay of other passengers of the same train decreases. For trains running on time, this argument does not hold, but if this is the first train of a batch, delays for other trains within the same batch decrease too, justifying the stop-skipping. Hence, the penalty for stop-skipping $w_{t,b}^{stop-skip}$ ideally represents the nuisance caused to passengers with destination o_s, who have to take another train. Objective function (5.9) can be extended as follows:

$$\min \sum_{t\in T} w_t^{cancel} x_t + \sum_{t\in T} w_t^{deviation} dev_t + \sum_{t\in T} w_t^{delay} D_t + \sum_{t\in T} \left(\sum_{b:(t,b)\in S} w_{t,b}^{stop-skip} \right) (1-\sigma_t) \tag{7.5}$$

The last term of objective function (7.5) adds a penalty $w_{t,b}^{stop-skip}$ for each stop of a stop-skipping train t ($\sigma_t = 0$), which ideally differs between trains and stops. Note that following Constraints (7.2), this penalty is also added to the objective function value in case of train cancellation or rerouting. Hence, cancelling an L train with stops may be worse than a non-stopping IC train if w_t^{cancel} is a fixed value. Therefore, train-specific cancellation penalties should be used with w_t^{cancel} not including the inconvenience caused to passengers boarding or alighting within the disrupted area.

Ideally, exact passengers numbers are known and used to determine $w_{t,b}^{stop-skip}$, but in the absence of this data an estimation can be made as follows. Let $\delta_{b,dir}^{stop}$ represent the estimated share of passengers boarding or alighting at the stop located in block section b for trains running in direction dir and $\omega_{b,stop}^{dir}$ a direction-specific multiplier for this fraction. This multiplier could represent the importance of passengers travelling in direction dir, or the time in-between trains with a stop in block section b relative to the scheduled train headway. For example, if trains in direction 0 run every 20 min and only one of them has a scheduled stop at b, $\omega^{dir}_{b,stop} = 3$. Using these values, $w^{stop-skip}_{t,b}$ is calculated as:

$$w^{stop-skip}_{t,b} = \delta^{stop}_{b,dir_t}\omega^{dir_t}_{b,stop}w^{cancel}_t \qquad \forall t \in T \qquad (7.6)$$

$$w^{cancel}_t \leftarrow \left(1 - \sum_{b:(t,b)\in S}\left(\delta^{stop}_{b,dir_t}\right)\right)w^{cancel}_t \qquad \forall t \in T \qquad (7.7)$$

Equation (7.6) determines the stop-skipping penalty as a fraction of the one for cancellation, considering the fraction of passengers and the additional time relative to that for train cancellation. Cancelled trains automatically skip their stop and the update of w^{cancel}_t using Equation (7.7) avoids double-counting of these passengers.

7.2.3 Additional service constraint: enforcing a minimum number of stops

To ensure passengers remain able to reach their destination by train during the disruption, a minimum number of trains may still have to stop at o_s in block section b. Hence, dispatchers could set a minimum number of stops per direction per hour, which can only be fulfilled by those trains with a commercial stop planned in the original timetable. Let $N^{min,dir}_{stop,o_s}$ denote the minimum number of trains in direction $dir \in \{0,1\}$ required to stop at block section b of stop o_s per hour, and recall that T^h_{dir} is the set of trains running in direction dir during hour h. The requirement is formulated as follows:

$$\sum_{t\in T^h_{dir}:(t,b)\in S} \sigma_t \geq N^{min,dir}_{stop,o_s} \qquad \forall o_s \in S, \forall h \in \mathscr{H}, dir \in \{0,1\} \qquad (7.8)$$

Constraints (7.8) ensure at least $N^{min,dir}_{stop,o_s}$ trains perform a stop at o_s.

7.3 Application on the Oostkamp-Aalter case study

Extending the model to include stops, introduced a range of new input parameters for the model. Before solving scenarios for the Oostkamp-Aalter case study, these are specified in Table 7.1. The stop at Oostkamp is located in segment A and cannot be skipped. Skipping the stop in Beernem within segment S_1, is allowed depending on the closed track, i.e. whether it is part of the disrupted area or not. The original timetable as visualized in Figure G.1 of Appendix G contains more train stops at Beernem than at other locations, interpreted as a higher demand: the minimum dwell time, and fraction of passengers are higher. Here, it is assumed that passengers boarding and alighting at Beernem and Maria-Aalter represent 10 and 5% of total passenger numbers for all trains respectively. On the other hand, $\omega^{dir_t}_{b,stop}$ is lower due to higher frequency of service.

Several scenarios have been run to assess the model's performance in generating disruption timetables for the upcoming 3 h after the disruption's start. Table 7.2

Table 7.1: Characteristics of all three stops located along the corridor for the Oostkamp-Aalter case study. They are located within a block section identified by the segment and the ordinal number of the block section, e.g. stop MA lies within the third block section of segment D. Additionally, the fraction of passengers boarding and alighting and the penalty multiplier $\omega^{dir}_{b,stop}$ are assumed direction-independent. The last column mentions the resulting stop-skipping penalty $w^{stop-skip}_{t,b}$ as a fraction of the penalty for cancellation.

Stop (o_s)	Location of stop o_s	Minimum dwell time (s)	Fraction passengers ($\delta^{stop}_{b,dir}$)	Can be skipped	Multiplier $\omega^{dir}_{b,stop}$	Stop-skipping penalty $w^{stop-skip}_{t,b}$
Oostkamp	A1	30	5% / 5%	No	-	-
Beernem (B)	S1	45	10% / 10%	Yes / No	2	0.20 w^{cancel}
Maria-Aalter (MA)	D3	30	10% / 10%	Yes	3	0.15 w^{cancel}

provides an overview of additional input parameter values for each scenario. Choosing $w^{cancel} = 7,20$ did not render results which are considered interesting for illustration purposes. Therefore, w^{cancel} was set to 3,600. Next, Table 7.2 reports solution statistics for the MIPfocus settings, which led to optimal results for all scenarios. Currently, dispatchers do not consider stop-skipping unless platforms cannot be reached. Therefore, the FIFO heuristic presented in Section 5.6 does not consider it.[2]

[2] Similar arguments hold for the (max,wait) heuristic, however, this was not evaluated because of the need for fine-tuning.

Table 7.2: Description of each scenario for the case study Oostkamp-Aalter, presenting information on the set-up, i.e. direction of the closed track, disruption speed v_D and additional parameters. Solution statistics report the maximum, total and average delay values, together with the number of cancelled trains and skipped stops, making up the weighted objective function. Finally, the last two columns report the objective function values for solutions obtained by the FIFO approach, mentioning both the absolute value and relative increase compared to the model's result.

#	Disruption characteristics			Max. delay (s)	Total delay (s)	Average delay (s)	Number cancelled	Number stops skipped	Objective function value	CPU time (s)	FIFO	
	direction	v_D (km/h)	others								Absolute value	Relative increase
1	0	80		1,758	15,123	540	0	5	14,660.0	6.4	43,403	296%
2	0	80	$N^{min,dir}_{stop,MA} = 1$	1,758	13,558	521	2	1	18,441.0	12.7	43,403	235%
3	0	80	Delay Thalys	1,436	14,494	518	0	2	12,411.0	2.7	49,683	400%
4	1	80		1,968	15,054	602	1	16	19,628.0	20.8	75,831	386%
5	1	80	$N^{min,dir}_{stop,MA} = 1$ $N^{min,dir}_{stop,B} = 1$	2,790	21,961	845	1	10	23,482.0	15.5	75,831	323%
6	0	80	Relax orders $N^{min,dir}_{stop,MA} = 1$	1,758	14,683	544	1	2	16,353.0	47.5	43,403	265%
7	0	80	Relax orders $N^{min,dir}_{stop,MA} = 1$ Prioritize IC	1,883	15,280	566	1	1	18,278.7	88.4	-	-
8	0	80	Relax orders	1,758	15,028	537	0	3	13,485.0	62.9	43,403	322%
9	0	80	Relax orders Prioritize IC	1,883	15,625	558	0	2	15,439.5	92.7	-	-

7.3.1 Disruption on the lower track

Scenario 1 resembled the blockage of the track in direction 0, for which Figure 7.5 (left) shows the resulting time-distance diagram. Five trains skipped their stops, especially due to the low number of passengers boarding or alighting at Maria-Aalter ($\delta_{b,dir}^{stop} = 0.1$). For example, performing the stop indicated by the red circle would not have resulted in significant delay increases, as only its successor would get a knock-on delay. Nonetheless, it had already a considerable delay, which together with the knock-on delay justified stop-skipping.

For the first stop-skipping train (indicated by the first orange dashed circle), this decision avoided cancellation of that train. Scheduling it would have delayed the succeeding train, which would get in conflict with the prioritized Thalys train. Hence, stop-skipping is unavoidable if one wants to run the train.

Scenario 2 avoided this stop-skipping by requiring one stop per hour at Maria-Aalter in direction 0 ($N_{stop,M}^{min,0} = 1$) and resulted in the cancellation of the red appointed train in Figure 7.5 (left). However, also a number of other changes were identified, among which the cancellation of the train indicated by the orange dashed arrow resulting in less stop-skippings. Despite this $N_{stop,M}^{min,0} = 1$ was meant to influence only the first stop-skipping, the restriction had a much larger impact. As performing the stop required more capacity, it delayed possibly moving them to another (earlier) batch or cancelling them. Total delay decreased with 1,565 s compared to scenario 1 (-10%), but objective function value increased with 3,781 s (+26%).

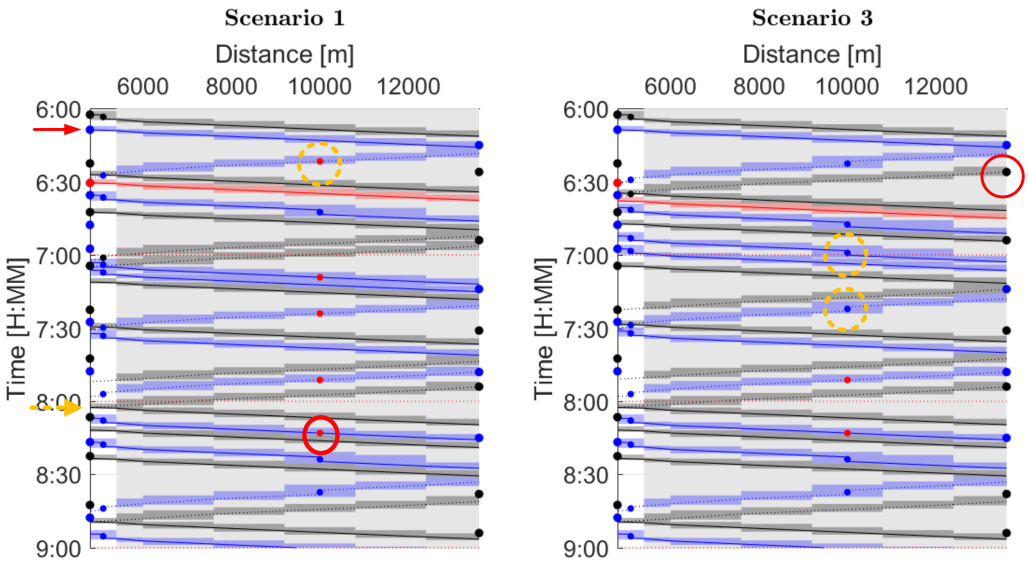


Figure 7.5: Time-distance diagrams scenarios 1 (left) and 3 (right) of the Oostkamp-Aalter case study.

Allowing the Thalys train to be delayed (scenario 3) resulted in significant improvements compared to scenario 1: objective function value decreased with 2,249 (-15%) and three stop-skippings could be avoided. The latter may seem contradicting with the fact that only one train before the Thalys had to skip its stop. However, the time distance diagram in Figure 7.5 (right) indicates many batches were rearranged. Including the train encircled in red in the first direction 0-batch, allowed a larger batch in direction 1 thereafter and avoided the need for skipping the stops indicated by orange dotted circles. Note that the Thalys' own delay was only 4 min 52 s (292 s), about half of the average delay reported in Table 7.2. Hence, allowing slight delays for international trains might be good practice.

7.3.2 Disruption on the upper track

For closures of the upper track in Figure 7.3b, trains in both directions have to stop at the same platform in Beernem. Hence, results are expected to be worse than for scenario 1, and stop-skipping trains also skip Beernem. Scenario 4 modelled such a situation and as reported in Table 7.2, 16 stops were skipped by ten trains (as the time-distance diagram in Figure 7.6 (left) points out). Optimal exploitation of capacity would result in no stops on segment D, but this is counteracted by the penalties. Three trains still performed their stop, of which the first two (indicated by the red arrows) did not cause knock-on delays by doing so.

Requiring a minimum of one stopping train per hour for each direction and at both stations, as in scenario 5, resulted in a strong total delay increase of 45% (average

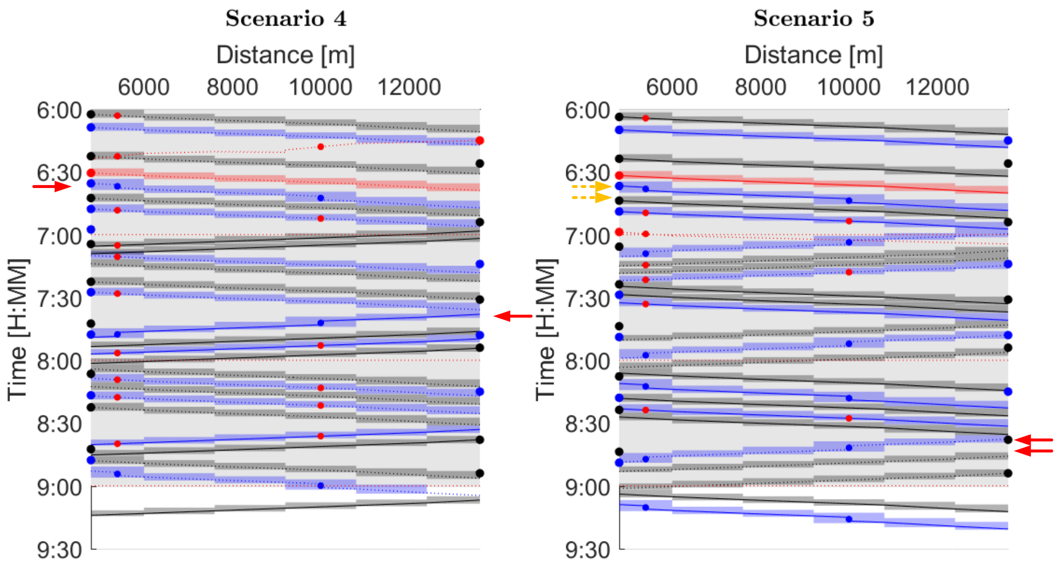


Figure 7.6: Time-distance diagrams scenarios 4 (left) and 5 (right) of the Oostkamp-Aalter case study.

+40%). However, due to the lower number of skipped stops, objective function value increased to lesser extents (+20%). No clear pattern to decide on stop-skipping could be identified from the time-distance diagram of scenario 5 (Figure 7.6, right). Several reasons may justify stop-skipping. One, if the first of two subsequent trains in the same direction has to perform a stop outside of the disrupted area, skipping the stop may not result in reduced headways. Hence, blocking time stairways cannot be moved closer together, which occurred for the pair of trains indicated by red arrows. Two, timetable structure is an important factor: trains within a batch may still perform their stops if no train was following right after (orange dotted arrows). Three, the blocked track has a significant impact on results. Compared to results for a disruption on the other track with a minimum number of stops (scenario 2), total delay and objective function value increased with 8,403 s (+62%) and 5,041 (+ 27%) respectively. The latter increase was smaller as one less train got cancelled. Additionally, measures strongly differed. Hence, stop-skipping can be seen as a very subtle measure, especially in combination with minimum level of service requirements.

7.3.3 Relaxing the order of trains

A fast train which gets behind a stopping train may either have an increased delay, or force the stopping train to skip its stops in order to reduce the delay. Swapping them may prevent both, and one could expect improvements when relaxing the order of trains entering the corridor from both directions. Scenario 6 incorporated the minimum level of service constraints of scenario 2, but relaxed all orders without

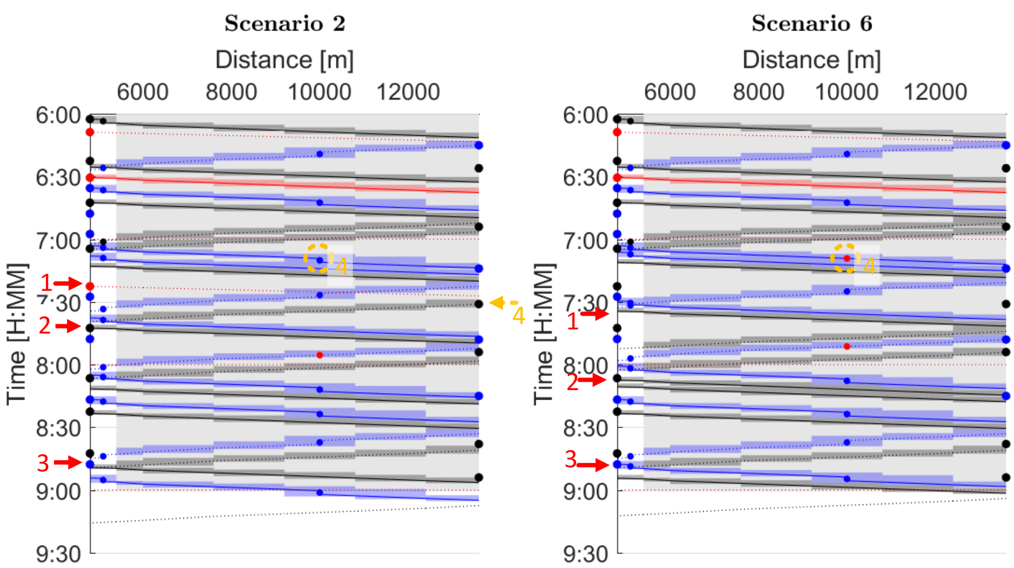


Figure 7.7: Time-distance diagrams scenarios 2 (left) and 6 (right) of the Oostkamp-Aalter case study.

a threshold δ^{swap} (see Section 5.5.5). Comparing time-distance diagrams for both scenarios 2 and 6 in Figure 7.7 (left) and (right), leads to identification of three order swaps, indicated by the red arrows. A fourth main observation is the addition of a stop and change in batch size (indicated in orange dotted). Various reasons could be identified (from top to down):

1. Swap 1 scheduled an IC train later than an L train and avoided the cancellation of the former. As a result, the next IC train behind this L train in scenario 2, moved to another batch and made a second swap over there.

2. Also swap 2 moved an IC train surprisingly behind the L train. The explanation lies in the homogenization of succeeding trains: the following one was also an IC train and their blocking time stairways could be moved closer together. Additionally, the L train performed a stop at Oostkamp (not shown in Figure 7.7) and scheduling it behind the IC train, as in scenario 2, led to a high exit delay for the L train. Hence, the summed delay of all three involved trains decreased and the swap was beneficial.

3. The third swap had a cause similar to the second explanation of swap 2: the L train performed a stop at Oostkamp. Swapping trains decreased the headway of the L train towards its predecessor, which was the previous train in direction 0 in Figure 7.7 (right). Without performing the swap, also the stop at Oostkamp, outside the disrupted area, would contribute to the headway with the IC train.

4. The first swap only became beneficial after a stop-skipping much earlier in the timetable, moving a train to a next batch. Hence, the L train of the first swap could enter the corridor much earlier. Note that these measures did not influence the third swap as there is some buffer in the disruption timetable after the trains of the first swap.

The overall effect of relaxing the orders was 10% decrease in objective function value: 16,353 instead of 18,224. Whereas for the first AIP (see Section 5.5.5), relaxing the orders resulted in limited improvements at the expense of strong computation time increases, this problem got solved within 48 s and showed significant improvements. Both follow from the increased distinction between trains: it becomes easier to decide which ones to cancel or swap as the effects of the latter are more prevalent.

Table 7.2 shows that increasing the priority of IC trains in scenario 7 by increasing their penalties with 30%, led to one less train cancellation compared to scenario 2: the IC train involved in the first swap in Figure 7.7. Nonetheless, timetable structure was altered and additional swaps occurred later on. Similar effects have been observed for relaxing the minimum level of service constraints in scenarios 8 and 9: compared to scenario 1, a few train swaps reduced both the number of skipped stops and/or total delay. These measures resulted in an 8% decrease in objective function value for scenario 8. Note that scenario 9 had a higher objective function value because of the increased penalties.

7.3.4 Conclusion

Stop-skipping appears to be a subtle measure which interacts with buffer times in the original schedule. Therefore, the FIFO heuristic performed much worse than it did for scenario 1 of the Tienen-Landen case study (Table 5.2). Although this heuristic may not be representative for dispatchers' decision-making, their way of considering stop-skipping might have to be revised based on observations presented here. Additionally, partly abandoning the absolute priority of international trains by allowing (small) delays, could significantly improve the performance of the system.

Comparison of scenarios for disruptions on different tracks, illustrated the benefit of extending the basic AIP in Figure 7.1 with stops on segments S_1 and S_2 (Figure 7.2). Improved levels of service are expected to arise from the reduction in disrupted area, when two platforms remain in operation at such stops.

One could conclude that relaxing the order of trains in the same direction, may improve solution quality in two ways. One, by creating more buffer time for stopping trains. Two, increased headways because of stops outside of the disrupted area might be avoided following the homogenization of journey times, e.g. non-stopping trains follow each other. To avoid cancellation of IC trains, dispatchers should increase their cancellation penalties.

7.4 Parameter impact assessment

Section 7.2 extended the model for the first AIP by introducing stops along all possible segments of the corridor. Here, the impact of several of the new parameters on the model's results is assessed by an artificial case study. Figure 7.8a represents the basic infrastructure, which extends the one of the first AIP (Chapter 6) with a stop on the second block section of segment D. Segments S_1 and S_2 are not split in two block sections because of the absence of a stop, and its associated signals. Hence, the disrupted area consists of segments S_1, D and S_2, resulting in a total of five block sections with a length of 7,600 m.

All local trains of the high-frequency timetable for the first AIP, presented in Table 6.1, have a scheduled stop at this location, requiring a minimum dwell time $t^{min}_{dwell,1} = 45\ s$. The resulting time-distance diagram in Figure 7.8b contains two stops per hour and therefore $\omega^{dir}_{b,stop} = 3$: through-going passengers can use six trains per hour, whereas passengers boarding or alighting at the stop can only use two of these. Unless stated differently, fraction of passengers with origin or destination at o_1 was $\delta^{stop}_{3,dir} = 0.075$ (7.5%), and no minimum number of stops (Constraints (7.8)) was enforced. Cancellation penalty is set to $w^{cancel} = 7,200$, the scheduling horizon was 3 h.

First, Section 7.4.1 evaluates the impact of the fraction of passengers boarding or alighting $\delta^{stop}_{3,dir}$, which results in varying values for the stop-skipping penalty. Next, the impact of the number of stops and stopping pattern is assessed in Section 7.4.2. Both analyses are conducted without minimum level of service requirements: Section 7.4.3 evaluates its effect. Other parameters could include the minimum dwell time,

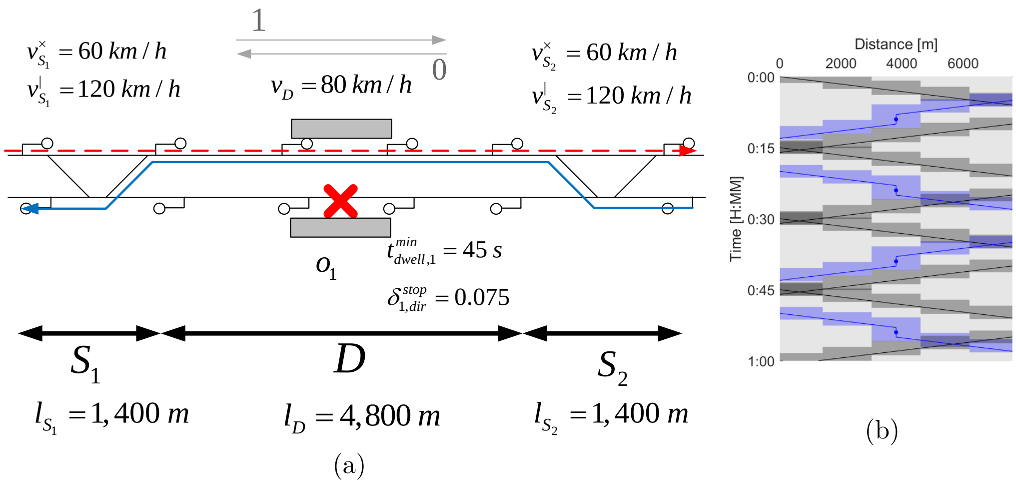


Figure 7.8: (a) Representation of the infrastructure used for the artificial case study to assess the impacts of parameters for the second AIP, with a stop o_1 at the middle block section. All information relevant to a blockage in direction 0, is mentioned in the sketch. (b) Original hourly time-distance diagram, including a stop for all local trains at o_1.

braking and acceleration rate. Scenarios have evaluated their impact, but due to their minor effects on running and set-up times, they did not result in significant differences and are not reported in this thesis. Moreover, during disruptions dwell times should be reduced as much as possible as it only consumes more capacity. Parameters varied in Chapter 6 mainly impacted on the batch size and distribution of trains over batches, on which stop-skipping is not expected to have significant impact. Hence, they are not discussed here.

All results reported are those obtained with the MIPfocus settings, as discussed in Appendix B. Comparison with both the FIFO and (max,wait) heuristics, developed in Section 5.6, is done by omitting stop-skipping as a measure. Settings for the (max,wait) heuristic are those found to be best-performing for the high-frequency timetable in Section 6.3: $N^{max}_{batch} = 5$ and $t^{wait} = 0\ min$.

7.4.1 Number of passengers

As Equation (7.6) expresses, the penalty for stop-skipping proportionally increases with the fraction of passengers boarding or alighting at the stop. Six scenarios have been constructed and solved with both small values for $\delta^{stop}_{3,dir}$ (5, 7.5 and 10%) and larger ones (20, 30, 50%). Note that the latter group presents less realistic scenarios as stops along segment D are not the major origins and destinations. As reported in Table 7.3, scenario A7 had differing $\delta^{stop}_{3,dir}$ values per direction. It is expected that with increasing fraction of passengers, less trains skip their stop.

Table 7.3 presents the delay statistics, i.e. maximum, total and average, and number of skipped stops for all scenarios. In line with results for the first AIP

Table 7.3: Description of each scenario when varying the fraction of passengers boarding and alighting at the stop, presenting information on the set-up, i.e. the exact fraction and resulting stop-skipping penalty ($w^{stop-skip}$). Solution statistics report the maximum, total and average delay values, together with the number of skipped stops, constituting the weighted objective function. No trains had to be cancelled for any of the scenarios. Finally, the last columns report the objective function values for solutions obtained by the FIFO and (max, wait) heuristics, mentioning both the absolute values and relative increases compared to the model's result. An asterisk (*) is used to indicate scenarios for which results are the same as those obtained within 900 s and proven optimal.

#	Fraction passengers $\delta_{3,dir}^{stop}$ (%)	Stop-skipping penalty $w^{stop-skip}$	Max. delay (s)	Total delay (s)	Average delay (s)	Number stops skipped	Objective function value	FIFO		(max, wait)	
								Absolute value	Relative increase	Absolute value	Relative increase
A1	5	1,080	2,260	32,958	916	6	34,020	89,246	162%	39,280	15%
A2	7.5	1,620	2,310	34,190	950	5	36,872*	89,516	143%	39,280	7%
A3	10	2,160	2,383	37,868	1,052	1	34,610*	89,786	159%	39,280	13%
A4	20	4,320	2,462	40,970	1,138	0	35,552*	90,866	156%	39,280	10%
A5	30	6,480	2,462	40,970	1,138	0	35,552*	91,946	159%	39,280	10%
A6	50	10,800	2,486	42,611	1,184	0	37,193	94,106	153%	39,280	6%
A7	*dir* 0 : 5	1,080	2,340	36,525	1,015	2	33,267	89,246	168%	39,280	18%
	dir 1: 10	2,160									

(Section 6.2), no trains were cancelled for $w^{cancel} = 7,200$. The last columns report the objective function values of the model's results and those obtained by both heuristics, together with the relative increase for the latter two. The FIFO heuristic resulted in seven train cancellations of which one stopping train. As a result, objective function value increased with increasing fraction of passengers, while the disruption timetable itself did not change. The (max,wait) heuristic did not cancel any train.

MIPfocus settings rendered the best solutions for six out of seven scenarios and is of most practical use. In addition, optimality could be confirmed within the 900 s time limit for five out of seven scenarios. The sudden objective function value increase in scenario A6 compared to scenarios A4 and A5, which did not skip any stop, proves the suboptimality of results for scenario A6.

As expected, Table 7.3 reports a decrease in number of skipped stops with increasing fraction of passengers, attributed to the higher penalty. Less trains skipping their stops caused additional delays for all other trains, as the increased running times of the local trains resulted in knock-on delays for all following ones. Both heuristics showed improved performance with increasing fraction of passengers, which is in line with expectations: less stop-skipping reduces the gap between the model's freedom and that of the heuristics.

Figure 7.9 (right) shows the time-distance diagram for scenario A1, the one of scenario A9 of the first AIP (Section 6.2) is shown for comparison. In total, six trains skipped their stop, all within the first 1.5 h. Comparison with the scenario without any stops provides an explanation for these decisions. The orange dashed arrow indicates a train t which was moved to an earlier batch, resulting in more trains to be queuing in the other direction. Hence, the next batches kept on growing in size, which is beneficial from a capacity point-of-view. The growing batch sizes allowed trains to enter the disrupted area earlier and resulted in some buffer after 1.5 h. Hence, the train encircled in orange (dashed) used this buffer to perform its stop without inducing too much knock-on delays. Buffer time was not sufficient to also perform the next stop. All trains within the red circle still performed their stops: the smaller number of queuing trains led to increased buffer times. Additionally, less trains still had to be scheduled, resulting in smaller induced delays.

Increasing $\delta_{3,dir}^{stop}$ to 7.5% in scenario A2 did not result in differing train batch sizes. The only effect was one less train skipping its stop. The stop between the encircled ones in Figure 7.9 (right) is performed because of an interaction between the increased penalty, and buffer time later in the schedule (before the red appointed trains): part of the additional running time is absorbed by this buffer. Although the remaining part induced additional delays, it was not sufficient to justify stop-skipping.

A fraction of 10% of passengers boarding and alighting at the stops (scenario A3) resulted in stop-skipping no longer being of interest for all but the first train. Performing the additional stops led to increased processing time for the batches, more trains queuing and hence larger batch sizes. Although Table 7.3 reports a total delay increase of 15% compared to scenario A1, cancellation did not come into play.

A further increase in $\delta_{3,dir}^{stop}$ for scenarios A4 did not result in stop-skippings anymore, as reported by Table 7.3. Hence, scenario A5 led to exactly the same result

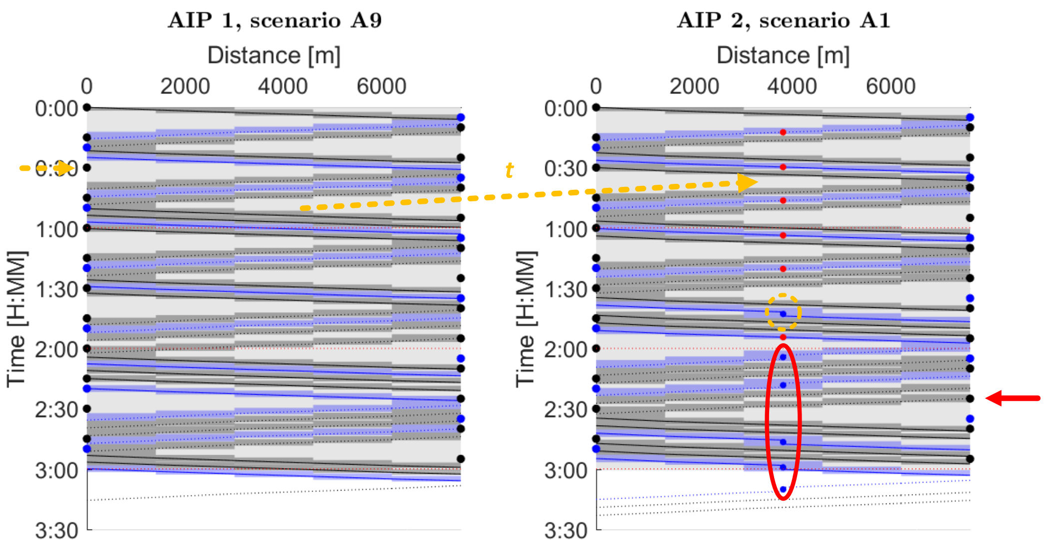


Figure 7.9: Time-distance diagrams for scenario A9 on the first AIP (left) and scenario A1 on the second AIP (right).

as scenario A4, as it only differed in stop-skipping penalty, a measure not considered anymore.

Only for scenario A6, the MIPfocus settings performed worse than the 900 s computation time limit. As expected, allowing more computation time led to the exact same results as for scenarios A4 and A5. Comparison of time-distance diagrams attributed the 5% objective function value increase to a single decision to wait for an additional train. As a result, a large number of trains started queuing at the other end of the corridor, leading to a snowball effect for batch sizes. Eventually, 40 out of 1,260 order variables differed in value, illustrating how strong variables are interlinked with each other.

Finally, scenario A7 assessed the effect of differing fraction of passengers per direction, which translated to differing stop-skipping penalties. Two trains skipped their stop, which is somewhere midway between the results for scenarios A1 and A3 (see Table 7.3). Figure 7.10 (right) presents the resulting time-distance diagram. Before, the distribution of stop-skipping trains over both directions 0 and 1 was more evenly, e.g. three versus two for scenario A2 (Figure 7.10, left). In scenario A7, both stop-skipping trains ran in the direction with a lower fraction of passengers, while both scenarios A2 and A7 had on average the same number of passengers boarding and alighting. Unexpectedly, stopping trains in direction 0 also differed in the decision on stop-skipping compared to scenario A2. The time-distance diagram in Figure 7.10 (right) shows that to mitigate the total delay of scenario A7, batch sizes were larger. Performing the stop encircled in red became possible due to the increased buffer time. For the one encircled in orange (dashed line), buffer time was not sufficient. Average delays increased compared to scenario A2 with 36 and 95 s for

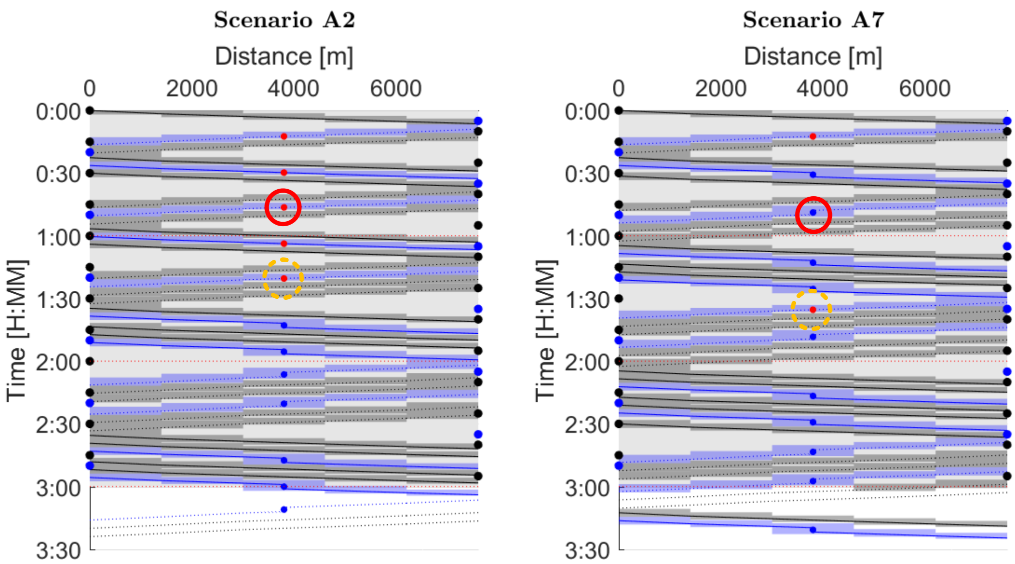


Figure 7.10: Time-distance diagrams for scenarios A2 (left) and A7 (right) on the second AIP.

trains in direction 0 and 1 respectively. The latter is larger following the additional stops in direction 1.

Conclusion

Decisions on performing a stop tend to interact with buffer times in the schedule, which could either follow from timetable structure, i.e. release times, or might be created by the model, i.e. re-assigning trains to batches. The additional delays caused by the stopping operation could then be (partly) absorbed by this buffer, resulting in a remaining additional delay which does not seem to justify stop-skipping anymore.

With increasing fraction of passengers boarding or alighting at segment D, the number of trains skipping their stops decreases as expected. When a certain threshold for $w^{stop-skip}$ or $\delta^{stop}_{b,dir}$ is reached, stop-skipping is no longer beneficial.

Often, stop-skipping is conducted at the start of the scheduling horizon. Nonetheless, both differing passenger numbers by direction as well as buffer times, may violate such a “rule”, illustrating the subtleness of the decision.

7.4.2 Number of stops and stopping pattern

The number of stops of all trains together and how these are distributed over the specific trains, i.e. stopping pattern, influence both running and set-up times. As stop-skipping is decided on a train-to-train basis, an increased number of stops for a train t increases the penalty $w^{stop-skip}$ and less trains are expected to skip their stop.

Intuitively, trains with less stops are expected to be more prone to stop-skipping than those with a higher number of stops.

Therefore, this section evaluates three scenarios on the infrastructure shown in Figure 7.11. First, all trains had one stop located in the first block section of segment D. Scenario B2 evaluated the effect of different stopping patterns with the first L trains in both directions stopping at both the first and third block section, whereas the second ones only stop on the first block section (see Table 7.4). Finally, all local trains had a second stop at o_2 in scenario B3. Figures 7.12a, 7.12b and 7.12c show one hour of the original time-distance diagrams for scenarios B1, B2 and B3 respectively. Following observations of Section 7.4.1, $\delta_{b,dir}^{stop}$ is set to 7.5%, considered to be a realistic fraction.

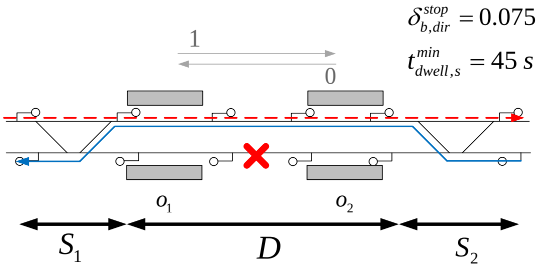
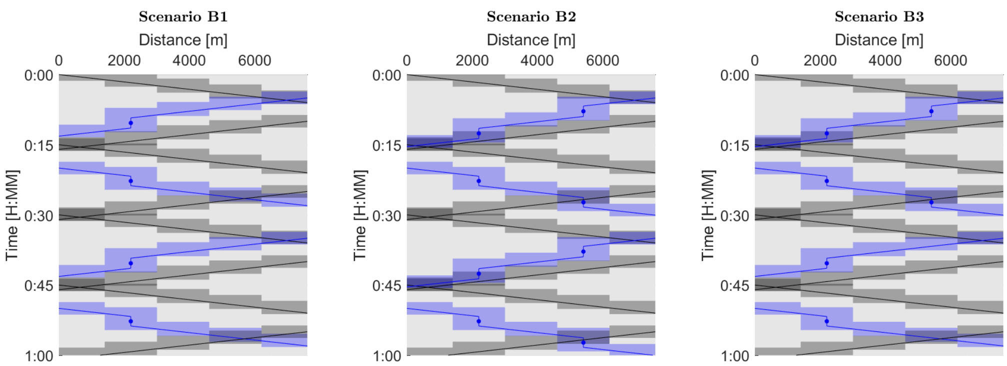


Figure 7.11: Configuration of the infrastructure for scenarios B1 to B3 on the second AIP, alongside with information on stop locations, fraction of passengers and dwell time.

Table 7.4: Number of hourly train stops at stops o_1 and o_2 per direction for each of the scenarios considered.

#	Stops at o_1 (trains/h)		Stops at o_2 (trains/h)	
	dir 0	dir 1	dir 0	dir 1
B1	2	2	0	0
B2	2	2	1	1
B3	2	2	2	2



(a) (b) (c)

Figure 7.12: Time-distance diagrams for one hour in scenarios (a) B1, (b) B2 and (c) B3 on the second AIP, indicating which of the local trains stop and where they perform stops.

Table 7.5 reports the delay statistics and number of skipped stops, the latter counts the trains with two stops twice. Adding additional stops strongly increased the objective function value between subsequent scenarios. MIPfocus settings rendered the best results for all scenarios, although none could be proven to be optimal. First of all, an increase in total delay could be expected: assuming that all stops were performed, the insertion of a new one induces small additional delays for each subsequent train with which there is no buffer time remaining. Hence, stop-skipping became more interesting to reduce total delay, but incurred a high additional penalty. Note that differences between scenario B1 and scenario A1 (also $\delta_{b,dir}^{stop} = 0.075$) in Section 7.4.1 arose from the differing location of the stop o_1. Previously, it was located in the third block section (see Figure 7.8a), whereas it now lies within the second on. Trains running in direction 0 had to cross the switches, resulting in increased running times in the block section of the stop as it had to brake earlier. Therefore, performing the stop relatively added less additional time to the running time and stop-skipping reduced the process time to a lesser extent.

Table 7.5: Description of each scenario when varying the stopping pattern, presenting information on the set-up, i.e. the number of stops per train (see Table 7.4) and associated penalties $w^{stop-skip}$. Solution statistics report the maximum, total and average delay values, together with the number of skipped stops, constituting the weighted objective function. No trains had to be cancelled for any of the scenarios. Finally, the last columns report the objective function values for solutions obtained by the FIFO and (max, wait) heuristics, mentioning both the absolute values and relative increases compared to the model's result.

#	Number of stops	Stop-skipping penalty $w^{stop-skip}$	Max. delay (s)	Total delay (s)	Average delay (s)	Number stops skipped	Objective function value	FIFO		(max, wait)	
								Absolute value	Relative increase	Absolute value	Relative increase
B1	1	1,620	2,357	37,664	1,046	1	33,806	88,142	161%	37,883	12%
B2	1 or 2	1,620 or 3,240	2,598	38,983	1,114	5	42,442	91,332	115%	54,002	27%
B3	2	3,240	2,770	40,028	1,144	8	49,184	95,405	94%	62,964	28%

When additional stops are added for each train (scenarios B2 and B3), stopping patterns come into play. First of all, more trains skipped their stops for scenario B3 (Figure 7.13, right): four trains, i.e. eight stops. Their running times increased twice over the full disrupted area and buffer times were not sufficient to allow performing both stops. Otherwise, larger delays would have been incurred to the subsequent trains. This contrasts with Section 7.4.1, were increased fraction of passengers, i.e. the penalty, led to less stop-skippings. Here, the increased number of trains skipping their stop avoided accumulation of delays as Table 7.5 reports a limited total delay increase of 6% from scenario B1 to B3.

One would expect trains with only one stop to be more prone to stop-skipping because of the lower penalty. However, Figure 7.13 (left) shows mainly trains with two stops skipped theirs in scenario B2. Nonetheless, following the explanations for the result of scenario B3, it seems to be a plausible solution: trains with only one stop would reduce total delay to a lesser extent. Finally, notice that train batch sizes were exactly the same for scenarios B2 and B3 (Figure 7.13), which suggests that for these scenarios, the originally scheduled buffer times were more decisive than the artificially created ones.

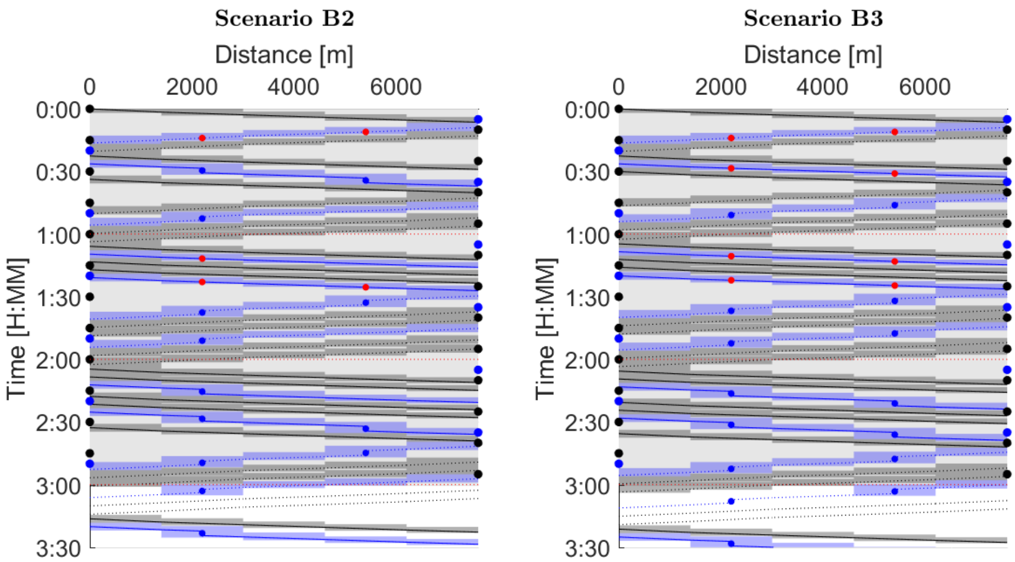


Figure 7.13: Time-distance diagrams for scenarios B2 (left) and B3 (right) on the second AIP.

Conclusion

The higher the number of stops of a train, the more important the interplay between stop-skipping penalty and induced delay seems to be. Skipping the stops would result in a higher penalty as more passengers are affected, but may also save more (induced) delays. This may lead to counter-intuitive results: trains with a higher number of stops, skipped theirs more often.

Extending the conclusion of Section 7.4.1, one could advocate the scheduled buffer times are more important than those created by scheduling larger batches. Perhaps, this is due to the specific scenario set-up.

7.4.3 Minimum number of stops

Service constraints such as a maximum batch size, or minimum level of service may considerably affect solution quality as was illustrated for the first AIP in Section 5.5.5. Here, the impact of a minimum number of stops $\left(N_{stop,1}^{min,dir}\right)$ is assessed by varying it from 0 to 2 for the basic scenario described by Figure 7.8a and Table 7.4. As Table 7.6 reports, scenarios included both imposing the same $N_{stop,1}^{min,dir}$ for each direction (scenarios C3 and C5) and only for one of them (scenarios C2 and C4). Note that a $N_{stop,1}^{min,dir} = 2$ for both directions $dir \in \{0, 1\}$ in scenario C5 boils down to scheduling all stops. Scenario C1 is equivalent to scenario A2 in Section 7.4.1. Table 7.6 reports delay statistics, number of cancellations and skipped stops for the MIPfocus solutions, together with the objective function value difference with the heuristics' solutions. The latter had absolute values of 89,516 and 39,280 for the FIFO and (max,wait) heuristics respectively.

Imposing Constraints (7.8) can only result in equal or worse objective function values. However, Table 7.6 reports worse values for scenario C1 compared to scenario C3, indicating that the more restricted solution space eased up the search for better solutions. Additionally, scenarios C3 itself was more restrictive than scenario C2 but led to better results. Both skipped the stop of one train in direction 0, but results for scenario C3 did it earlier on, decreasing objective function value with only 2% compared to scenario C2. These observations confirm suboptimality for (some of) the solutions found.

To draw conclusions regarding the $N_{stop,1}^{min,dir}$ parameter, scenarios were repeated for a scheduling horizon of 2 h. For all scenarios the optimal solution was found within 20 s of computation time, statistics on the resulting timetables are presented in Table 7.7. Observation of all time-distance diagrams showed that no unexpected or unexpected measures were taken anymore.

Table 7.6: Description of each scenario when increasing the minimum number of stops during a 3 h-scheduling horizon, presenting information on the requirements, i.e. minimum number of stops per direction per hour. Solution statistics report the maximum, total and average delay values, together with the number of skipped stops, constituting the weighted objective function. No trains had to be cancelled for any of the scenarios. Finally, the last columns report the objective function values for solutions obtained by the FIFO and (max, wait) heuristics, mentioning both the absolute values and relative increases compared to the model's result. An asterisk (*) is used to indicate scenarios for which results are the same as those obtained within 900 s and proven optimal.

#	Minimum number of stops $N_{stop,1}^{min,dir}$		Max. delay (s)	Total delay (s)	Average delay (s)	Number stops skipped	Objective function value	FIFO		(max, wait)	
	dir 0	dir 1						Absolute value	Relative increase	Absolute value	Relative increase
C1	0	0	2,310	34,190	950	5	36,872	89,516	143%	39,280	7%
C2	1	0	2,395	38,360	1,066	1	34,562	89,516	159%	39,280	14%
C3	1	1	2,383	37,868	1,052	1	34,070	89,516	163%	39,280	15%
C4	2	0	2,340	38,286	1,064	1	34,488	89,516	160%	39,280	14%
C5	2	2	2,462	40,970	1,138	0	35,552*	89,516	152%	39,280	10%

Table 7.7: Description of each scenario when increasing the minimum number of stops during a 2 h-scheduling horizon, presenting information on the requirements, i.e. minimum number of stops per direction per hour. Solution statistics report the maximum, total and average delay values, together with the number of skipped stops, constituting the weighted objective function. No trains had to be cancelled for any of the scenarios. Finally, the last columns report the objective function values for solutions obtained by the FIFO and (max, wait) heuristics, mentioning both the absolute values and relative increases compared to the model's result. All results were proven to be optimal.

#	Minimum number of stops $N_{stop,1}^{min,dir}$		Max. delay (s)	Total delay (s)	Average delay (s)	Number stops skipped	Objective function value	FIFO		(max, wait)	
	dir 0	dir 1						Absolute value	Relative increase	Absolute value	Relative increase
D1	0	0	2,310	21,874	911	1	19,882	49,432	199%	21,143	6%
D2	1	0	2,310	21,874	911	1	19,882	49,432	199%	21,143	6%
D3	1	1	2,310	21,874	911	1	19,882	49,432	199%	21,143	6%
D4	2	0	2,310	22,385	933	1	20,393	49,432	191%	21,143	4%
D5	2	2	2,389	24,028	1,001	0	20,416	49,432	191%	21,143	4%

Based on the time-distance diagram for scenario D1 in Figure 7.14, the single skipped stop (encircled in red) is explained by the large delay savings later on. All following trains got a smaller delay, which created some buffer time later on in the schedule (between the trains indicated by red arrows). Hence, delays which arose during the first part of the scheduling horizon, were no longer transferred to trains in the second part. Additionally, for each direction at least one stop per hour was performed, which turned out in the same solutions for scenarios D1, D2 and D3.

Finally, the time-distance diagram illustrates that batch size kept increasing with longer duration, also following from the increased process time of stopping trains. Only near the end of the scheduling horizon, batch size exceeded $N_{batch}^{max} = 5$ of the (max, wait) heuristic, indicated by the green box in Figure 7.14 (left). As a result, the (max,wait) heuristic performed remarkably well compared to the model. Nonetheless, the latter succeeded to improve objective function value (-6%) by skipping this stop.

Only within the first hour, the solution to scenario D1 (Figure 7.14, left) did not satisfy the minimum level of service constraint for direction 0 in scenario D4. Table 7.7 reports a 3% increase in objective function value for scenario D4, with the same number of skipped stops. The time-distance diagram in Figure 7.14 (right) shows the first stopping train in direction 1 no longer performed its stop (encircled in dashed orange), thereby creating some buffer to perform the stop indicated in red. For the remainder, timetable structure was not altered compared to the solution obtained for scenario D1 (Figure 7.14, left). Hence, one could conclude the induced delay of performing the skipped stop was 511 s smaller than for the skipped stop in scenario D1, equal to the difference in objective function values between scenarios D1 and D4 (Table 7.7).

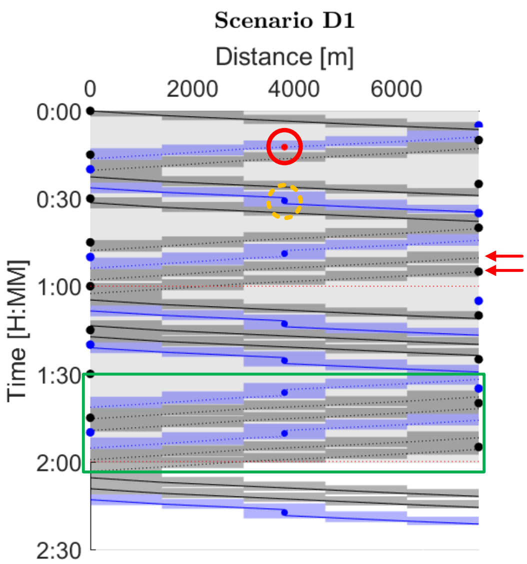


Figure 7.14: Time-distance diagrams for scenarios D1 (left) and D4 (right) on the second AIP.

Requiring the same level of service for direction 1 (scenario D5), did no longer allow the stop-skipping encircled in red and dashed orange in Figure 7.14 (left) and (right) respectively. It resulted in a total delay increase of 2,154 s compared to scenario D1 (+10%), which is higher than the stop-skipping penalty $w^{stop-skip} = 1,620$. Batch sizes were not altered compared to other scenarios and the buffer time between the appointed trains in Figure 7.14 (left) disappeared. As a result, the second train of this pair entered with a small delay, transferring it to all following trains.

Conclusion

Although a minimum level of service might be satisfied by the basic solutions, i.e. without Constraints 7.8, such situations cannot be predicted on beforehand. Results showed that manually adding a stop without skipping another one, could render the solution suboptimal compared to the one which would have been obtained upon setting requirements right from the start (see scenario D4). Therefore, dispatchers should be advised to incorporate such constraints right from the start, rather than changing the solution manually afterwards. Additionally, this may also help in finding better solutions by restricting solution space. However, such a conclusions seems to be difficult to generalize.

On the other hand, care has to be taken when defining $N^{min,dir}_{stop,o_s}$ to avoid imposing constraints which are too strict to obtain good results. For example, during morning peak hours, only a $N^{min,dir}_{stop,o_s}$ for the most crowded direction may be sufficient to safeguard solution quality. Imposing it for both directions, may lead to strong efficiency decreases.

7.5 Conclusion

This chapter extended the model developed in Chapter 5 for the first AIP, by explicitly including stops along the corridor. Although the latter could already be dealt with by the first model, it allowed to include stop-skipping as a measure. Complications may arise when a stop is located in segment S_1 or S_2, presenting a second extension. Scenarios for both the practical and artificial case study, illustrated the added value of both extensions.

Including stop-skipping required adjusting the single-machine scheduling problem with rejection, to one with rejection and (discretely) controllable process times. However, due to the specifics of railway operations, also the sequence-dependent set-up times became dependent on the execution of a stop. The penalty for reducing the process times includes aspects of both passenger numbers and timetable structure.

The most striking observation is the interaction between the stop-skipping decision and buffer times. Additionally, the latter could be created artificially by altering batch sizes. It is likely dispatchers do not take such interactions into account, whereas the comparison with the heuristics' results illustrate the added value of considering these measures. Nonetheless, within a sequence of batches which did not have any buffer times, mostly the first trains skipped their stop(s). This observation is in line with those for cancellations on the first AIP.

In addition, the subtleness of stop-skipping is illustrated by some observations of the parameter impact analysis. One, despite the increased penalties, skipping the stops of trains with two stops seemed to be better - and optimal - for some scenarios. Two, including a minimum number of stops tends not to change the resulting disruption timetable for some cases. Nonetheless, it may result in a more difficult problem to solve.

To employ the model in practice, dispatchers should provide some additional input. In addition to the length of segment D, the disruption speed, the direction of switches, and the scheduling horizon, as for the first AIP, the model requires information on the location of the stop(s), the (estimated) passenger fractions, and possibly a minimum number of stops. Other parameters such as the multiplier for the stop-skipping penalty, and minimum dwell time, could be derived automatically from the timetable's structure.

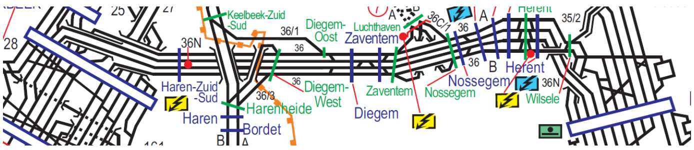
Chapter 8 continues by considering corridors with multiple tracks. In light of the added value of incorporating stop-skipping as a measure, it should also be considered for this third AIP.

Chapter 8

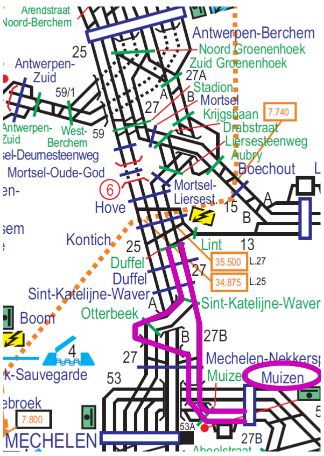
Partial Blockage of a Multi-Track Corridor

Whereas the first and second AIP in Chapters 5 and 7 consisted of two tracks, this chapter focusses on corridors with more than two tracks, possibly with stops located along them. Over the last decades, several double-track corridors have been upgraded to four tracks, of which Figure 8.1 shows two examples.

Section 8.1 provides a description of the AIP and introduces the practical case study. The model frameworks and optimization models used in Chapters 5 and 7 have to be extended and adapted, which is considered in Section 8.2. The model's results are compared with those obtained by the FIFO heuristic. Section 8.3 elaborates on how the version for the first and second AIP, presented in Section 5.6.1, is extended



(a)



(b)

Figure 8.1: Examples of the third AIP on the Belgian network, holding four tracks between two stations: (a) Schaarbeek - Leuven and (b) Mechelen - Antwerpen-Berchem. For the latter, the purple lines indicate which tracks can be reached by a train originating from Muizen. Adapted from [26].

to incorporate track assignment decisions. Next, Sections 8.4 and 8.5 apply the developed model on both a practical and artificial case study. The latter is used to assess the impact of the newly introduced parameters. The basic model for this AIP does not account for the segments S_1 and S_2 due to their differing configurations. Section 8.6 extends the model of Section 8.2 to incorporate conflicts arising at these segments S_1 and S_2. Finally, Section 8.7 presents the conclusions for this third AIP.

8.1 Archetypical infrastructure piece description

In theory, the two-track AIPs in Chapters 5 and 7 can also be modelled as a general N-track section. However, previous AIPs had a more standardized layout. Segment D was aligned with double switches at both ends, allowing trains in both directions to reach the remaining track. This no longer holds true for the third AIP. For example, consider the network in Figure 8.1b: trains running from Muizen towards Antwerpen-Berchem cannot reach the two left-most tracks between Mechelen and Kontich as indicated by the thick purple routes. Hence, initially only the segment D is considered for this AIP, making abstraction of the junctions S_1 and S_2 aligning it.[1]

Figure 8.2 presents the AIP under consideration. Two stations A and B have a segment D with a number of parallel tracks in-between them, denoted by the set M_T (with $|M_T| = m$). Due to a disruption, n of these m tracks are closed ($n < m$), and the remaining r tracks may impose a lower-than-regular disruption speed v_r. Stations A and B are separated from the segment D by (station) junctions S_1 and S_2, which allow trains to change tracks. Additionally, trains in direction 1 may also change tracks before station A. Similar arguments arise for trains in direction 0. Alongside segment D, stops o_s may be located next to (a subset of) the tracks in segment D.

[1]Section 8.6 extends the initial model to incorporate these and conflicts which may arise here.

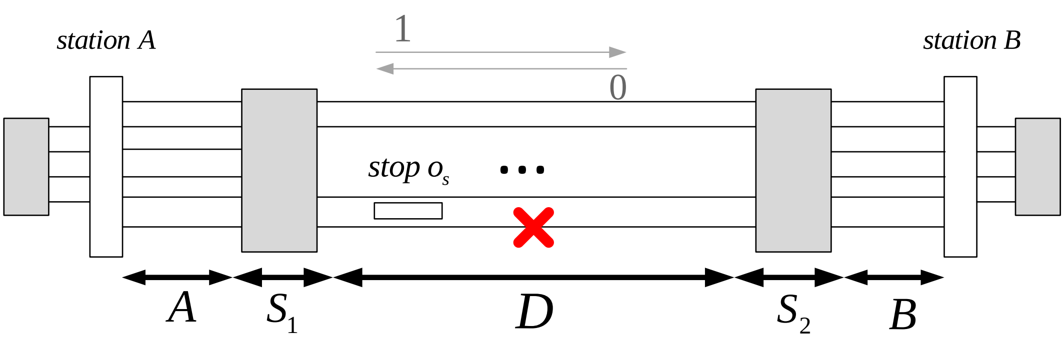


Figure 8.2: General configuration of the third AIP, holding a number of tracks between two stations A and B. Within the segment D, stops o_s may be located. Abstraction is made of the configuration of the segments S_1 and S_2.

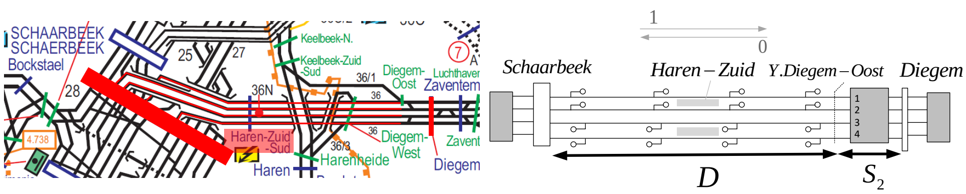
8.1.1 Case study Schaarbeek - Diegem

The corridor between the station of Schaarbeek and the junction of Diegem-Oost (Y.Diegem-Oost) consists of four tracks with a stop at Haren-Zuid, as Figure 8.3a shows. Unfortunately, no data on the infrastructure itself, i.e. location of signals and length of block sections, was at hand. Based on [53], the length of segment D was determined as approximately 4.4 km. Using an average block section length of 1,600 m, this is modelled as three block sections within segment D. The stop Haren-Zuid is located within the middle one, resulting in the schematic representation in Figure 8.3b. Tracks are numbered from 1 (top) to 4 (bottom). In normal track regime, trains running in direction 1 use tracks 1 and 2, trains in direction 0 the other ones.

The morning peak of the 2016 timetable between 6:30 and 9:30 was considered. Departure times from Schaarbeek and Diegem as found in the line folder of line 36 [38] were used as entry times for the directions 1 and 0 respectively. If the train did not stop at these locations, entry times in direction 1 were based on the departure time from station Brussel-Noord increased with 4 min, representing the running time of trains which do stop at Schaarbeek. Trains in direction 0 entered the segment D 13 min before arriving at Brussel-Noord, also based on the running times of stopping ones.[2] Appendix H presents an overview of entry times, track assignments and train types for each direction.

The complete timetable included 25 and 33 trains in directions 1 and 0 respectively over the full 3 h period, an off-balance due to the morning peak demand. Both directions had two S trains per hour which stop at Haren-Zuid, dwelling for a minimum of 45 s. Six different train types run on this infrastructure, which were assigned to two categories, holding the same characteristics as IC and L trains as before (see Section 5.4). P, L and S trains were classified as L trains. With the inner two tracks constituting the first part of the high-speed line, both international ICE and Thalys trains run on this part of the network. For the same reasons as in Section 7.1.1, both were assigned characteristics of the IC trains, but received a prioritized

[2]Note that these simplifications are not required in real-life application by Infrabel. The required data would be at hand, and can be used directly without changing the model itself.



(a) (b)

Figure 8.3: (a) Technical representation of the infrastructure for the Schaarbeek-Diegem case study, indicating the part under study in red. Adapted from [26]. (b) Translation of (a) as it can be used by the model. Segment D holds four tracks consisting of three block sections, with the stop Haren-Zuid located in the middle one.

treatment, i.e. no cancellation or delays. Respectively one and two international trains run in directions 1 and 0 over the complete scheduling horizon.

Finally, no information on the original track assignment was at hand. Constructing the case study, trains in direction 1 were divided over tracks 1 and 2 to get an (almost) balanced timetable, similar as for trains in direction 0 over tracks 3 and 4. Restrictions allowed international trains only to use tracks 2 and 3, and stopping S trains were assigned to tracks 1 and 4 to avoid hindrance.

8.2 Model adaptations and extensions

This section discusses how the model of Chapter 7 is adjusted. Figure 8.4 illustrates how the model framework for the first two AIPs is extended. A timetable holding all trains running over the corridor is constructed for each track: each train has four copies, of which only one is used in the original and disruption timetable. All processes within the red box have to be conducted for each track separately. Hence, also the track on which the train runs is required as input and collected as output. Finally, the complete disruption timetable can be reconstructed.

8.2.1 Decision variables

Providing multiple alternative tracks on which a train can be scheduled during the disruption, extends the problem from scheduling on a single machine to a parallel-machine scheduling problem. Rocha et al. [49] present several MILP formulations for the parallel-machine scheduling problem with sequence-dependent set-up times, one of which is based on the Manne concept [33]. However, cancellation and deviation, i.e. rejection, still have to be included. As a train may have to perform a stop alongside segment D, also stop-skipping is possible.

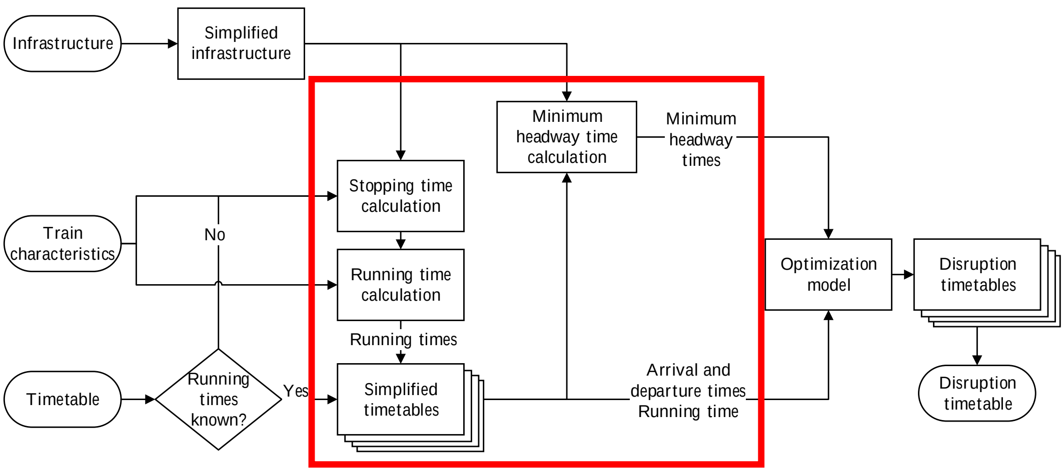


Figure 8.4: Model framework for the third AIP, for which all processes within the red box are conducted separately for each track.

Due to the multitude of tracks, some of the sets and decision variables get additional indices and have to be redefined. Indices k refer to tracks unless specified otherwise. For each train t, k_t^{orig} denotes the originally scheduled track on which it enters the disrupted area. Similarly, k_t denotes the newly assigned track. The following sets and variables are used within the optimization model:

Sets

- T Set of trains
- M_T Set of tracks, i.e. machines
- O Set of open-track stops along the segment D.
- S Set of performed stopping operations within the original timetable, including combinations (t,b) of trains t stopping on a block section b.

Variables

- $t_t \geq 0$ Starting time of train t on any of the tracks.
- $C_t \geq 0$ Completion time of train t, i.e. departure time from the corridor.
- $D_t \geq 0$ Delay of train t when leaving the corridor.
- $x_t \in \{0, 1\}$ Cancellation variable of train t, $x_t = 1$ means that the train is cancelled.
- $dev_t \in \{0, 1\}$ Deviation variable of train t, $dev_t = 1$ means that the train is rerouted outside of the corridor under consideration.
- $\sigma_t \in \{0, 1\}$ Stopping variable of train t, indicating whether train t performs all of its stops ($\sigma_t = 1$) or none ($\sigma_t = 0$).
- $m_{tk} \in \{0, 1\}$ Track assignment variable, $m_{tk} = 1$ assigns train t to track k.
- $q_{ijk} \in \{0, 1\}$ Binary order variable of trains i and j on track k. Following the Manne concept [33] and the formulation of Rocha et al. [49], $q_{ijk} = 1$ if train i is scheduled before train j on track k, not necessarily directly before. Additionally, if one of both is scheduled on another track, q_{ijk} may take any value in $\{0, 1\}$.

Next to these sets and variables, the model contains several parameters which will be introduced while formulating the constraints.

8.2.2 Reformulation of essential constraints

Constraints of the model for the second AIP (Section 7.2) are reformulated to incorporate track assignment. Additionally, some extensions are required. However, as changing the track of a train is not penalized and the decision variables incorporated in objective function (7.5) have not been changed, the latter can be used as before.

Adaptations to the original model

Each track has its own timetable with all trains included. Hence, running, stopping and set-up times can be determined for each (pair of) trains on every track by the methods developed in Sections 5.5.2 and 7.2.1. These running times may differ due to different reference speed or speed restrictions v_r, resulting in different process and set-up times.

Release time r_t of each train on any of the tracks is assumed to be the same and Constraints (5.3) can be used as before:

$$t_t \geq r_t(1 - x_t - dev_t) \qquad \forall t \in T \tag{8.1}$$

Minimum process times depend on the assigned track and may decrease when train t skips a scheduled stop. Additionally, it has to be taken into account that not all tracks have a platform at the stop:

$$C_t \geq t_t + p_{tk} - M(1 - m_{tk}) + \sum_{b:(t,b)\in S} \left(\Delta r^{stop}_{tbk}\right) \sigma_t \qquad \forall t \in T, \forall k \in M_T \tag{8.2}$$

$$\sigma_t \leq \sum_{k\in M_T} \pi_k m_{tk} \qquad \forall t \in T \tag{8.3}$$

Constraints (8.2) express the minimum process time in a similar way as in Section 7.2.1 with track-specific $\Delta r^{stop}_{tbk} = R^{stopping}_{tbk} - R_{tbk}$. Parameter p_{tk} represents the minimum running time over track k without any stops. Using the assignment variable m_{tk}, constraints are relaxed when train t does not operate on track k. Parameter π_k indicates whether platforms are located at all stops along track k ($\pi_k = 1$) or not. Hence, Constraints (8.3) only allow a train to perform its stops, i.e. $\sigma_t = 1$, when this condition is fulfilled.

Next, delays can be calculated as before, by relating the completion time C_t with the due date d_t:

$$D_t \geq C_t - d_t(1 - x_t - dev_t) \qquad \forall t \in T \tag{8.4}$$

$$D_t \geq 0 \qquad \forall t \in T \tag{8.5}$$

Again, the question which d_t to use, arises. Before, it represented the earliest possible departure time from the corridor given the lower disruption speed, i.e. the adapted departure time. Here, tracks may differ in allowed speed and hence the possible departure time. Parameter d_t is set to the adapted departure time from the scheduled track. As such, scheduling trains on a faster track may lead to negative delays for Constraints (8.4). Constraints (8.5) ensure this does not improve the objective function value.

Two trains i and j running on the same track k require a minimum time separation depending on both the track k and the stopping patterns of the trains. For each track k, the set-up time can be determined between any pair of trains $(i, j) \in T \times T$ as described in Section 7.2.1, resulting in $s^{disrupted}_{ijk}$ and $s^{stopping}_{ijk}$. However, it is not sure whether this headway is sufficient when trains are scheduled on a track differing from their originally assigned one, i.e. $k_t \neq k_t^{orig}$, as they have to cross a switch or junction to reach it. The first and second AIPs modelled these movements explicitly, enabled by the standardized infrastructure. For a four-track junction with different configurations, this is no longer possible when one wants to have a general model. Hence, the increase in set-up time is modelled by adding small buffers h^{before} and h^{after} when a train crosses switches, respectively before and after it.

For instances of the first AIP, the increase in blocking times when crossing the switch was in most cases lower than 30 s. Extending blocking and set-up times with this value for h^{before} and h^{after}, is expected to avoid most of the conflicts:

$$s_{ijk} \leftarrow \begin{cases} s_{ijk} + h^{before} + h^{after} & \text{if } k \neq k_i^{orig} \wedge k \neq k_j^{orig} \\ s_{ijk} + h^{before} & \text{if } k = k_i^{orig} \wedge k \neq k_j^{orig} \\ s_{ijk} + h^{after} & \text{if } k \neq k_i^{orig} \wedge k = k_j^{orig} \\ s_{ijk} & \text{if } k = k_i^{orig} = k_j^{orig} \end{cases} \tag{8.6}$$

Equation (8.6) can be applied to both the values for stopping $\left(s_{ijk}^{stopping}\right)$ and stop-skipping trains $\left(s_{ijk}^{disrupted}\right)$.

Set-up times only have to be enforced if both trains are running on the same tracks, i.e. $m_{ik} = m_{jk} = 1$, resulting in the need to adapt Constraints (7.3) and (7.4), based on the formulation by Rocha et al. [49], but incorporating stop-skipping:

$$t_j \geq C_i + s_{ijk}^{disrupted} + \Delta s_{ijk}\ \sigma_i - M(3 - q_{ijk} - m_{ik} - m_{jk}) \quad \forall i,j \in T, \forall k \in M_T \tag{8.7}$$

$$t_i \geq C_j + s_{jik}^{disrupted} + \Delta s_{jik}\ \sigma_i - M(2 + q_{ijk} - m_{ik} - m_{jk}) \quad \forall i,j \in T, \forall k \in M_T \tag{8.8}$$

The second and third terms of the right-hand sides in Constraints (8.7) and (8.8) account for the alteration of set-up times in case of performing the stops. Compared to Constraints (7.3) and (7.4), their relaxation is modelled differently. Assume both trains i and j run on the same track k, and j is scheduled after i ($q_{ijk} = 1$), i.e. $q_{ijk} + m_{ik} + m_{jk} = 3$. It is only in those cases that Constraints (8.7) are enforced. If one of the conditions is not fulfilled $q_{ijk} + m_{ik} + m_{jk} < 3$ and they are relaxed. Constraints (8.8) follow from Constraints (8.7) by using symmetry: $q_{jik} = 1 - q_{ijk}$.

Finally, for trains running in the same direction, the order is assumed to be fixed regardless of the track on which the trains operate. Constraints (8.9) are added to the model:

$$q_{ijk} = \begin{cases} 0 & \text{if } r_i > r_j \\ 1 & \text{if } r_i < r_j \\ 1 & \text{if } r_i = r_j \wedge p_i < p_j \\ 0 & \text{if } r_i = r_j \wedge p_i \geq p_j \end{cases} \qquad \forall i,j \in T, \forall k \in M_T \tag{8.9}$$

However, trains i and j in the same direction originally scheduled on different tracks, may have exactly the same release time ($r_i = r_j$). Dispatchers mostly allow the faster train first onto the track. Scheduling the slow one first, would lead to a higher total delay as the fast one gets behind the slower one. The third case in Constraints (8.9) schedules train i before train j if train i is the faster one.

Extensions to the original model

Additional constraints are required as only one machine can be chosen:

$$\sum_{k \in M_T} m_{tk} + x_t + dev_t = 1 \qquad \forall t \in T \tag{8.10}$$

$$m_{tk} \leq z_{tk} \qquad \forall t \in T, \forall k \in M_T \tag{8.11}$$

Constraints (8.10) ensure that each train t is assigned to exactly one track $k \in M_T$ if it is not cancelled or deviated. These constraints allow to model the relaxation of Constraints (8.2), (8.7) and (8.8) by means of the track assignment variables. In case a train is cancelled or deviated, $\sum_{k \in M_T} m_{tk} = 0$, i.e. every $m_{tk} = 0$, and the constraints are relaxed.

Before, trains could reach each track in segment D by means of the switches S_1 and S_2. This may no longer hold true as illustrated by the example in Figure 8.1b. Moreover, additional *track restrictions* may arise from aspects such as safety systems. Parameter z_{tk} indicates whether train t can operate on track k ($z_{tk} = 1$), or not ($z_{tk} = 0$).

8.3 Track assignment in the FIFO heuristic

Section 5.6.1 presented an extended version of the well-known FIFO heuristic to incorporate cancellation of trains, which resembles a rather naive dispatcher and served as a point of reference for the first and second AIP in Chapters 5, 6 and 7. Additionally, international trains got absolute priority and stop-skipping was not considered, in line with current practice. Assigning trains to tracks as for this third AIP, requires additional extensions.

In regular operations, tracks mostly serve trains in only one direction. Therefore, if multiple tracks remain during a disruption, dedicating each of them to serve traffic in a specific direction seems a reasonable reaction if the number of trains in both directions does not differ too much. Hence, it is assumed dispatchers assign the upper remaining tracks in Figure 8.2 to trains in direction 1, and the lower ones to direction 0. In case an odd number of tracks remain, the middle one can serve both directions.

Hence, tracks are characterized by the directions they serve, trains by the allowed tracks following the configuration of segments S_1 and S_2. Within a FIFO logic, the dispatcher assigns trains to tracks in order of release time on the disrupted area. Trains in the same direction with exactly the same release times are handled from fast to slow. When considering a train, the earliest time it can enter onto (one of) its allowed track(s) is determined. Trains are assigned to those tracks where they can enter the earliest, taking into account possible conflicts with international trains.

8.4 Application on the Schaarbeek-Diegem case study

The presented model is first applied to the practical case study presented in Section 8.1.1. Several scenarios have been evaluated both based on those interesting to illustrate the model's constraints, and to identify possible variations depending on the parameter values.

Table 8.1 presents a description of each scenario, reporting the speed on each track. Column 'RES' indicates whether track restrictions were imposed (x) or not. Regardless of this value, international trains were not allowed to run on the outer tracks. Furthermore, cancellation and stop-skipping penalties have been set to $w^{cancel} = 7,200$ and $w^{stop-skip} = 1,080$, using $\omega_{b,H}^{dir_t} = 2$ and $\delta_{b,dir_t}^{stop} = 0.075$. International trains could not be delayed, nor cancelled.

Additionally, Table 8.1 presents total delay, number of train cancellations and stop-skipping for the results obtained with the MIPfocus settings. Out of all three computational settings, the MIPfocus settings rendered the best results for all scenarios. Asterisks indicate those results which could be proven to be optimal within 900 s of computations.

Throughout the analysis in the remainder of this section, the number of time-distance diagrams shown will be limited to those which are considered interesting to illustrate observations. Each time-distance diagram represents the traffic on one of the tracks.

Closing one track

Scenario 1 included the closure of track 1, the uppermost in Figure 8.3b, without S trains performing their stop at Haren-Zuid. Table 8.1 shows that delays remained rather limited as plenty of capacity was left. Although cancellation did not come into play, optimality of the solution could not be proven. This could be attributed to degeneracy of the problem: no direct incentive to keep trains running on their originally scheduled track is included, the inclusion of h^{before} and h^{after} partly promoted such assignments. However, if a train could be scheduled on another track without causing a delay to itself, or another train due to these extra buffers, it may do so. In other words, scheduling it on any track without a delay renders exactly the same result. Hence, a large number of solutions with the same objective function value existed.

Table 8.1 reports that trains were operated in both directions for each track in scenario 1. Because of the small incentive, the main direction of traffic on a track was still conserved (see Table 8.1), i.e. track 2 is mainly used by trains running in direction 1, tracks 3 and 4 trains in direction 0. Trains running originally on the first track, were divided over the remaining tracks with respectively eleven, two and two trains added.

Table 8.1: Description of each scenario for the case study Schaarbeek-Diegem, presenting information on the set-up, i.e. speed v_r for each track r and whether track restrictions apply ('RES'). Solution statistics report total delay, together with the number of cancelled trains and skipped stops, making up the weighted objective function. Finally, the number of trains running per track per direction is reported. Empty cells indicate either a 0 value, e.g. number of cancelled trains, or unavailability, e.g. track 1 is closed for scenario 3. An asterisk (*) is used to indicate scenarios for which results are the same as those obtained within 900 s and proven optimal.

#	Track speed v_r (km/h)				RES	Total delay (s)	Number cancelled	Number stops skipped	Objective function value	Number of trains								FIFO	
										Track 1		Track 2		Track 3		Track 4		Absolute value	Relative increase
	1	2	3	4						0	1	0	1	0	1	0	1		
1		80	80	80		1,311			141			1	21	17	2	15	2	141	0%
2		80	80	80		1,236			222			6	17	14	2	13	6	222	0%
3		80	80	80	x	1,311			141*			2	19	9	5	22	1	141	0%
4		80	80	80	x	1,236			222*			4	19	11	5	18	1	222	0%
5			80	80		4,828			3,034					19	10	14	15	6,358	110%
6			80	80	x	5,255		1	4,593*					4	24	29	1	6,358	38%
7			80	80	(x)	4,834			3,040*					3	24	30	1	6,358	109%
8	80	80				6,056			3,638	22	7	11	18					12,760	251%
9	80	80			x	6,628			4,210*	4	23	29	2					12,760	203%
10	80	40				16,108			15,535	23	12	10	13					101,414	553%
11	80	40			x	19,249	1	3	28,781	9	24	23	1					101,414	252%
12	80					39,300	13	7	138,174	26	19							247,193	79%
13	80				x	23,364	19	7	165,852*	14	25							225,092	36%

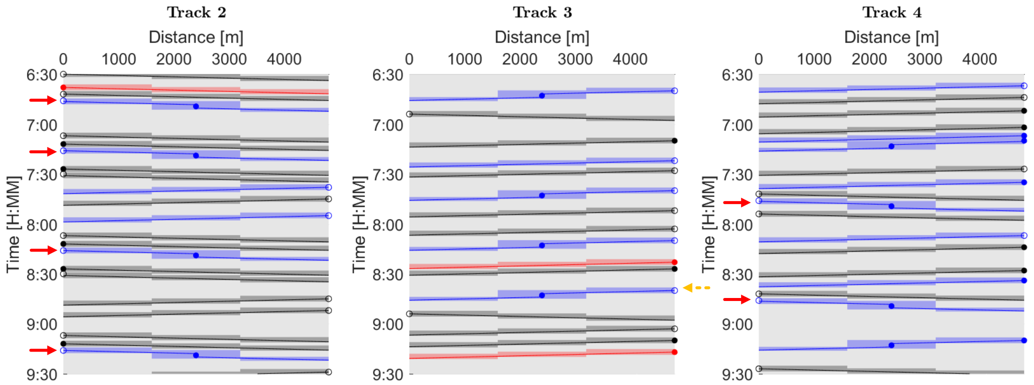


Figure 8.5: Time-distance diagrams for tracks 2 (left), 3 (middle) and 4 (right) in scenario 2 of the Schaarbeek-Diegem case study.

Including stopping operations for the S trains (scenario 2), led to a slight increase in total delay. On the resulting time-distance diagrams in Figure 8.5, red arrows indicate how four out of six stopping trains of track 1 were moved to track 2, and the two remaining ones to track 4. Stop-skipping did not come into play, but scheduling a stopping train in-between two other ones, required a larger gap. Inserting it in-between two trains in the same direction resulted in smaller set-up times with its new predecessor and successor, and hence, was less likely to cause conflicts. If it was not possible, gaps seem to have been created by moving other trains, e.g. the train on track 3 indicated by the orange dotted arrow moved from track 4 to 3 to create capacity for trains running in direction 1 on track 4.

Scenarios 3 and 4 were similar to scenarios 1 and 2 respectively, but included track restrictions. S trains in direction 1 could no longer run on track 4, which affected those indicared in Figure 8.5 (right). Hence, they were moved towards track 3. Imposing the track restrictions reduced solution space, resulting in finding the optimal solutions within less than 31 s. Objective function values remained the same, illustrating the degeneracy of the problem: solutions differed in structure, yet, not in objective function value. This strengthens the belief that solutions found for scenarios 1 and 2 were the optimal ones, although (slightly) better ones might have existed.

Closing two tracks

Scenarios 5 and 6 included the closure of tracks 1 and 2 without and with track restrictions respectively. All trains in direction 1 had to be moved to one of the remaining tracks, i.e. scheduled in-between traffic in the busiest direction. Table 8.1 shows total delay strongly increased from scenario 2 to 5, but no train skipped its stop or had to be cancelled. Figure 8.6 shows the resulting time-distance diagrams. Both tracks 3 and 4 processed two large batches of trains: track 3 first had only traffic running in direction 1, followed by a batch in direction 0, as opposed to track 4. This was attributed to the track restrictions for international trains, which could only use track 3. Hence, one of them ran in direction 1 on track 3 at the beginning of the time horizon and other trains in direction 1 started to cluster around it due to smaller set-up times. Trains in direction 0 were forced to be scheduled on track 4. For the second batch, the opposite occurred. As indicated by the red dashed lines, batches did not switch at the same time for both tracks. The number of trains in direction 0 was much higher than in direction 1, resulting in the need for larger batches.

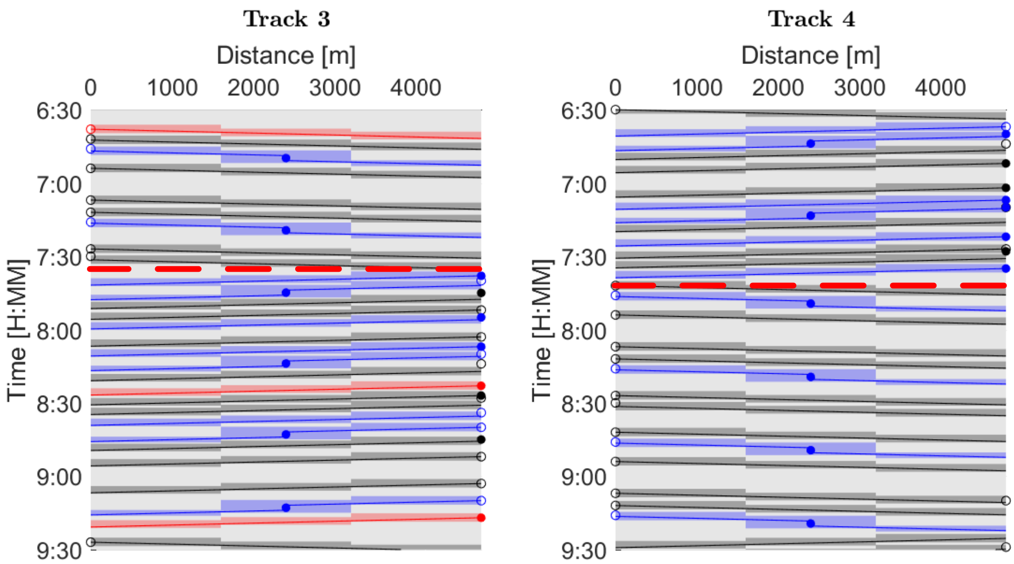


Figure 8.6: Time-distance diagrams for tracks 3 (left) and 4 (right) in scenario 5 of the Schaarbeek-Diegem case study.

Applying track restrictions in scenario 6 did not allow traffic of track 1 to be operated on track 4. Hence, these trains were either moved to track 3 or cancelled. The objective function value increased with 51% for scenario 6 compared to the solution of scenario 5. Figure 8.7 shows the resulting time-distance diagrams, illustrating how track 4 got dedicated to direction 0 except for one train (indicated by the red arrow) near the end of the scheduling horizon. Scheduling this one on track 3 would have resulted in a conflict with the prioritized international train.

On track 3, some trains were operated in direction 0. The train indicated by the red arrow in Figure 8.7 (left) suggests this was caused by capacity reasons: it resulted in delays for trains moving in the other direction on track 3, but scheduling it on track 4 would have caused much more delay. Due to gaps in the time-distance diagram of track 3, trains in direction 1 can still perform their stops. However, one train (encircled in red) had to skip its stop as it would have caused a delay to the international train otherwise. Without track restrictions (scenario 5) this one was scheduled on track 4.

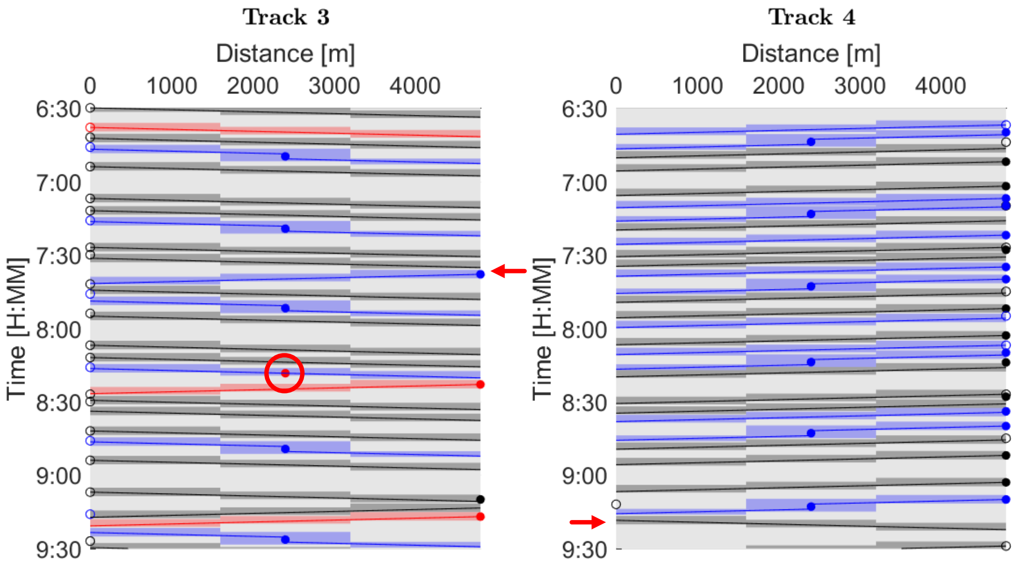


Figure 8.7: Time-distance diagrams for tracks 3 (left) and 4 (right) in scenario 6 of the Schaarbeek-Diegem case study.

In an optimal solution, one may have expected reserving each track for traffic in only one direction. Probably, due to both the international trains and the higher number of trains in direction 0, this did not happen. To schedule all traffic in direction 0 on the same track, stop-skipping may have been required in Figure 8.7 (right), which suggests all trains in direction 0 could be scheduled on track 4, but it is not optimal regarding the objective function, and not feasible in light of the international trains. The large difference between objective function values for scenario 5 and 6 also supports these conclusions, as it seemed to be better to schedule two batches on track 3 (if possible in terms of track restrictions). Scenario 7 (not shown here) relaxed the track restrictions for the international trains. Still, not all trains in direction 0 got scheduled on track 4 in the optimal solution as reported in Table 8.1.

Scenarios 8 and 9 also included the closure of two tracks without and with track restrictions respectively, but for those in the most busy direction, i.e. tracks 3 and 4. Table 8.1 shows a total delay increase of 20% for scenario 8 relative to scenario 5. Without track restrictions, trains in direction 0 mostly moved towards track 1, whereas track 2 showed a mix of both. One explanation could be the need to reschedule two international trains from track 3 to track 2, clustering some trains in direction 0 around it. Secondly, the first international train in direction 1 pushed some trains in direction 0 already towards track 1, especially the (stopping) ones of track 4, avoiding conflicts. Finally, less intuitive observations may be caused by suboptimality of the solution.

Including track restrictions in scenario 9 (not shown here) disabled scheduling the stopping trains on track 1 and dedicated track 2 almost exclusively for traffic in direction 0. Table 8.1 reports a lower objective function value than for scenario 7, in which the other two were closed but with the same restrictions. This can be attributed to avoidance of the stop-skipping indicated in Figure 8.7 (left): the train changed tracks, no longer conflicting with the international one.

Closing two tracks with different speed restrictions

Scenarios 10 and 11 repeated scenarios 8 and 9, i.e. tracks 3 and 4 were closed, with a lower speed right next to the closed tracks, i.e. $v_2 = 40km/h$. Hence, scheduling trains on track 1 instead of 2 reduced running and set-up times. Section 6.5 discussed how reducing speed v_D leads to the tendency of forming larger batches. Figure 8.8 plots the time-distance diagrams for tracks 1 and 2 in scenario 10, showing track 2 (Figure 8.8, right) indeed operating larger batches. Similar to previous scenarios, trains clustered around the international trains in the same direction. Compared to scenario 8 less trains could be scheduled on track 2. To avoid cancellations, stop-skippings and high delays, batches in direction 1 on track 1 grew as they could not be scheduled on track 2.

Applying track restrictions (scenario 11) resulted in dedicating track 2 to trains in direction 0. Because of the higher number of trains and the lower speed and capacity on track 2, also additional ones had to be moved towards track 1. One train of track 4 could not be added to track 2, and it was probably not possible to push other trains from track 2 towards track 1 to create capacity for it, resulting in a cancellation. Additionally, three trains had to skip their stop: one in direction 0 to avoid delays for the following international train and another two in direction 1.

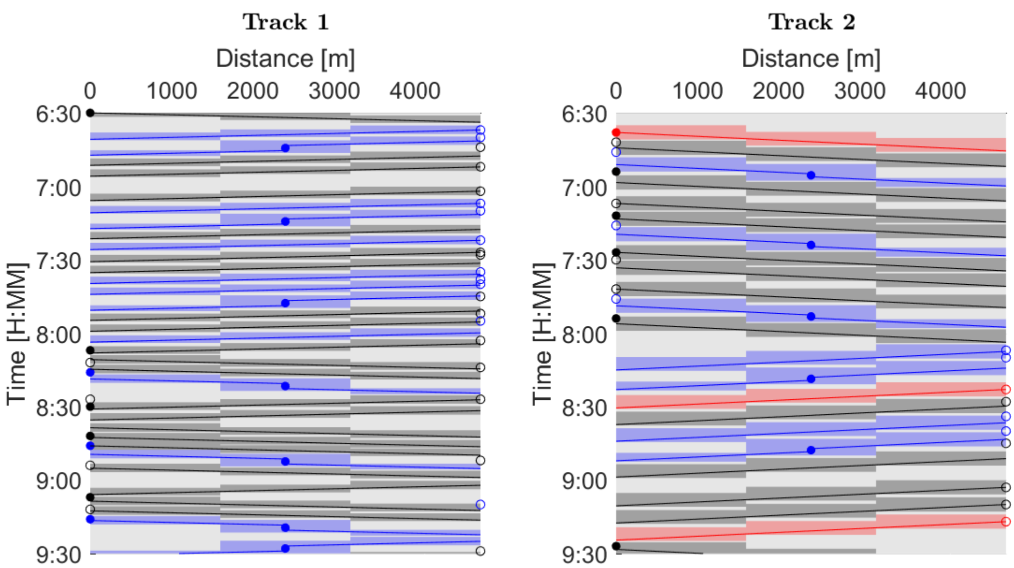


Figure 8.8: Time-distance diagrams for tracks 1 (left) and 2 (right) in scenario 10 of the Schaarbeek-Diegem case study.

Comparison of objective function values for scenarios 10 and 11 with scenarios 8 and 9 respectively, learns that, obviously, imposing speed restrictions has a strong adverse effect on level of service: Table 8.1 reports increases of respectively 327 and 584%. Hence, it seems to be advisable to keep speed restrictions to an absolute minimum. Similar arguments hold for the number of track restrictions, which could be accomplished by performing track changes outside of the corridor, i.e. grey boxes at the extreme ends in Figure 8.3b, if possible in terms of infrastructure. Such aspects are not taken into account by this model.

Closing three tracks

Finally, scenarios 12 and 13 included the closure of three tracks with only track 1 in Figure 8.3b available to operate traffic. International trains were allowed to be scheduled on track 1, whereas previous restrictions, i.e. only allowed on tracks 2 and 3, would have led to their unavoidable cancellation. Table 8.1 reports 13 and 19 train cancellations for scenarios 12 and 13 respectively.

In scenario 12 a large number of S trains skipped their stop in an attempt to reduce total delay. One would expect the track to be dedicated to traffic in one direction, preferably the more crowded direction 0. However, the solution had 25 and 19 trains running in directions 0 and 1 respectively with a large batch in direction 0 operating midway the disruption (orange dotted box in Figure 8.9, left). Several explanations seem plausible. First, it was not possible to prove optimality of the solution within the 120 s time limit. Second, the S trains running in direction 1 resulted in a higher total penalty when they are cancelled, as they also have to skip their stop: 8,280 compared to the regular w^{cancel} of 7,200. Hence, the additional penalty seemed to be sufficiently large to schedule them in-between trains in direction 0. As the red circles in Figure 8.9 (left) show, there is a tendency to form (small) batches around these stopping trains. Third, trains also clustered around the prioritized international ones.

However, scheduling all 45 trains in scenario 12 extended operations until after the scheduling horizon t_{end} = 9:30. As this period is not taken into account by the model anymore, additional conflicts may arise. Nonetheless, one may consider it plausible to execute this result by either moving some trains to the other tracks again, or by scheduling the traffic released after the disruption on the three tracks which were closed. Within the latter option, track 1 would be reserved to clear all queuing trains. On the other hand, if the disruption would last longer than the 3 h-scheduling horizon, a rolling-horizon approach as proposed before, should be able to deal with the arising issues.

Scenario 13 included additional track restrictions, resulting in the unavoidable cancellation of 19 trains originally scheduled on track 4, i.e. all cancellations were decided on beforehand. The time-distance diagram in Figure 8.9 (right) shows the optimal solution required one stop-skipping, and 39 trains were scheduled. This may seem to be a high number of trains running, especially when considering batches had at most seven trains, but probably resulted from both the short length of segment D, and the relatively high disruption speed. Moreover, it seems to be in line with

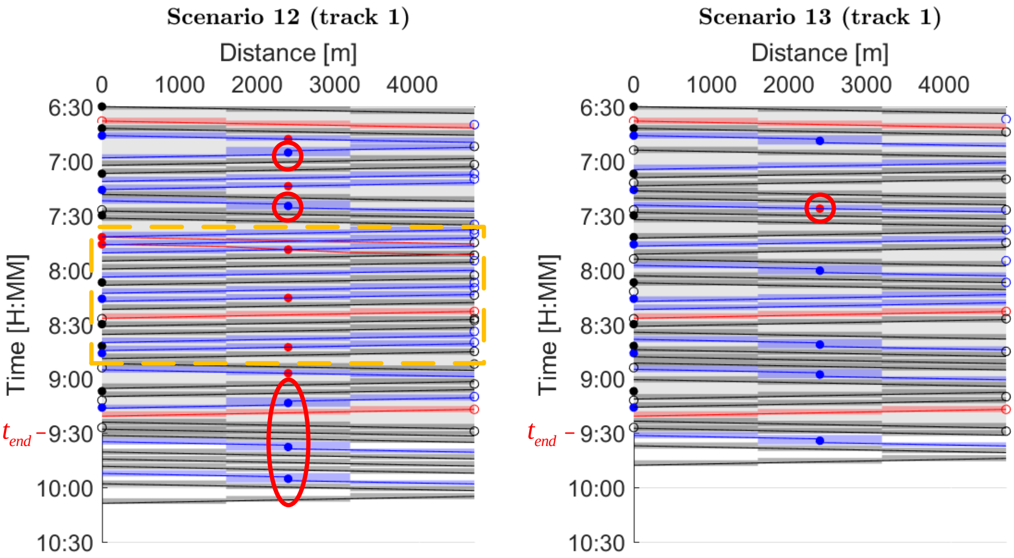


Figure 8.9: Time-distance diagrams for track 1 in scenarios 12 (left) and 13 (right) of the Schaarbeek-Diegem case study.

the absence of cancellation in cases with a high-frequency timetable on two tracks (see Chapter 6 and Section 7.4).

8.5 Parameter impact assessment

Section 8.2 introduced a number of new parameters to the models developed in Chapters 5 and 7, such as the number of closed tracks and possible track restrictions. This section assesses their impact by solving various scenarios of an artificial part of infrastructure and timetable.

A four-track corridor between two stations is considered as shown in Figure 8.10. Unless stated otherwise, all trains could reach all tracks in segment D by means of segments S_1 and S_2, or before stations A and B. Segment D contained five block sections, each with a length of 1,600 m and signals regulating traffic in both directions. A stop o_1, located along the middle block section, had platforms next to each track.

On this corridor, three different timetables with varying frequency may be operated: a low-, mid- and high-frequency one with respectively eight, ten and twelve trains per hour per direction. Tables 8.2a and 8.2b provide an overview of all trains t running on the corridor, indicating on which track they ran originally $\left(k_t^{orig}\right)$, and in which timetable(s) they were operated, for directions 1 and 0 respectively. Entry times are formulated as a number of minutes after each hour. The outer tracks were only used by IC trains, with a regular headway of 10 min for the high-frequency timetable. L trains were operated in each timetable, and performed a stop at o_1, dwelling for $t_{dwell,1}^{min} = 45\ s$. Timetables in direction 0 (Table 8.2b) were an exact

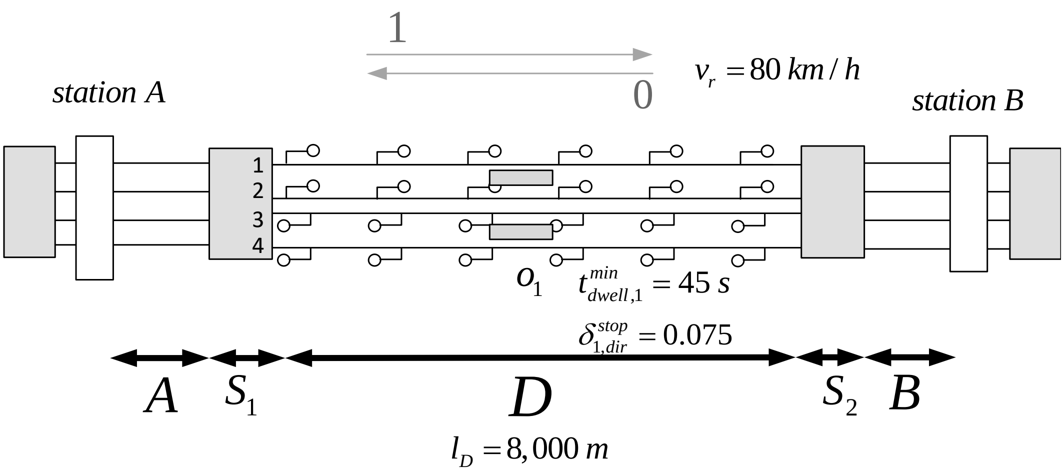


Figure 8.10: Representation of the infrastructure used for the artificial case study to assess the impacts of parameters for the third AIP. Segment D holds five block sections, with a stop o_1 located within the third one.

copy of those in direction 1 (Table 8.2a) except for the assigned tracks and entry times. On the outer tracks 1 and 4, entry times were shifted by 3 min, on the inner ones by 8 min.

During the 3 h-scheduling horizon, trains could be cancelled or could skip their scheduled stop for corresponding penalties $w^{cancel} = 7,200$ and $w^{stop-skip} = 1,080$. Disruption speed is reduced to 80 km/h. Section 8.5.1 examines the effect of closing

Table 8.2: Timetable information of the artificial case study to assess the impact of parameters for the third AIP. Information concerns the train type, its entry time, assigned track and the timetable(s) it runs in (x).

(a) Direction 1

Train type	Entry time (min)	Track k_t^{orig}	Timetable		
			Low	Mid	High
IC	0	1	x	x	x
IC	10	1			x
IC	20	1	x	x	x
IC	30	1	x	x	x
IC	40	1			x
IC	50	1	x	x	x
IC	10	2		x	x
IC	40	2		x	x
L	5	2	x	x	x
L	20	2	x	x	x
L	35	2	x	x	x
L	50	2	x	x	x

(b) Direction 0

Train type	Entry time (min)	Track k_t^{orig}	Timetable		
			Low	Mid	High
IC	13	3		x	x
IC	43	3		x	x
L	8	3	x	x	x
L	23	3	x	x	x
L	38	3	x	x	x
L	53	3	x	x	x
IC	3	4	x	x	x
IC	13	4			x
IC	23	4	x	x	x
IC	33	4	x	x	x
IC	43	4			x
IC	53	4	x	x	x

an increasing number of tracks, in relation with the number of trains running over the corridor. Section 8.5.2 investigates the effect of track restrictions for a closure of two tracks. Chapter 6 and Section 7.4 elaborated on the impact of parameters respectively for the first and second AIP. The discussion of their impact on results for the third AIP is limited to a brief qualitative description in Section 8.5.3.

8.5.1 Frequency and number of closed tracks

A first analysis focusses on the interplay between number of closed tracks, varying from one to three, and frequency of the original timetable. The number of closed tracks varied from one to three, resulting in nine scenarios. One could expect that with three tracks remaining, two of them get a dedicated direction as this would increase capacity following the shorter set-up times. The third track may show a number of batches in both directions. Closing an additional track redistributes trains, increasing the tendency to get a dedicated direction. Finally, closing three tracks potentially results in only one direction being served, as this may increase the number of trains that can be scheduled.

Table 8.3 reports for each scenario the timetable used and number of tracks closed. Closing tracks occurred from top to bottom in Figure 8.10, i.e. tracks 1 and 2 are closed for scenarios A2, A5 and A8. Next, Table 8.3 presents statistics about the solutions obtained with the MIPfocus settings in terms of total delay, number of cancellations and skipped stops, together with the resulting objective function value. The last columns display the number of trains running on the remaining tracks in each direction. Only the extended FIFO heuristic has been applied as a (max,wait) heuristic would hamper the creation of expected results: tracks would not be allowed to have a dedicated direction unless N_{batch}^{max} is very high.

Each scenario was initially run with all three computational settings as discussed in Appendix B. From observation of the time-distance diagrams for scenarios A3, A6 and A9, one could conclude that the MIPfocus settings did not manage to produce optimal solutions, even without knowing its objective function value. Solutions obtained had small batches running in the middle of the scheduling horizon, whereas Table 8.3 indicates a large number of trains were cancelled for these scenarios.

Table 8.4 reports the objective function values of solutions obtained with all three computational settings. The last two columns report the increase in objective function value compared to the 900 s time limit, which achieved the best results for all scenarios. Comparing objective function values, shows indeed MIPfocus settings performing much worse than regular settings. The difference increased with increasing frequency and accumulated to +9%. This follows from the increase in problem size.

Table 8.3: Description of each scenario when varying both frequency and number of closed tracks, presenting information on the set-up, i.e. which timetable is used and the number of closed tracks. Solution statistics report the total delay values, together with the number of cancelled trains and skipped stops, constituting the weighted objective function. Finally, the number of trains running per track per direction is reported. Empty cells indicate either a 0 value, e.g. number of cancelled trains, or unavailability, e.g. track 2 is closed for scenario A2.

#	Timetable (trains/h /direction)	Number of tracks closed	Total delay (s)	Number cancelled	Number stops skipped	Objective function value	Number of trains						FIFO	
							Track 2		Track 3		Track 4		Absolute value	Relative increase
							0	1	0	1	0	1		
A1		1	2,982			1,422	4	7	0	17	20	0	1,422	0%
A2	Low (8)	2	6,037			3,229			0	24	24	0	3,229	0%
A3		3	59,493	4	18	104,145					24	20	218,619	110%
A3 (180)		3	42,773	6	16	100,237					24	18	218,619	118%
A4		1	3,042			1,482	0	17	0	13	30	0	1,842	24%
A5	Mid (10)	2	7,317			3,729			30	0	0	30	4,089	10%
A6		3	47,142	17	18	185,388					24	19	307,880	66%
A6 (180)		3	54,492	15	16	175,716					25	20	307,880	75%
A7		1	5,364			3,024	0	29	29	0	7	7	3,384	12%
A8	High (12)	2	12,309			7,941			36	0	0	36	8,301	5%
A9		3	50,585	27	21	264,192					21	24	396,774	50%
A9 (180)		3	42,386	27	17	251,569					23	22	396,774	58%
A10	Low (8)	1	2,982			1,422.01	0	6	24	0	0	18	1,422	0%
A11	High (12)	3	5,712	36	12	275,844					36	0	301,404	9%

Nevertheless, the presented analysis will be focussed on the results obtained with the MIPfocus settings, as it is advised to use these in real-life operations (Appendix B). Additionally, for 6 out of 9 scenarios it managed to obtain the best results. In scenarios with three closed tracks, one may allow more time to find a solution as it may lead to much better services. However, a 900 s time limit is exaggerated. Therefore, scenarios A3, A6 and A9 were rerun with MIPfocus settings, but allowing 180 instead of 120 s. They correspond to rows A3 (180), A6 (180) and A9 (180) in Table 8.3. Table 8.4 shows results were close to those after 900 s, but improved with about 5% compared to a MIPfocus with 120 s time limit.

Closing one track

Tracks 3 and 4 got a dedicated direction for scenario A1 (Figure 8.11), as expected. On track 4, the six indicated trains got a small entrance delay, due to overlapping release times. However, these did not resemble conflicts within the original timetable as trains were scheduled on different tracks. For example, the second L train on track 3 (Table 8.2b) entered the corridor 23 min after each hour, just as the third IC train on track 4. The low-frequency timetable had two of such overlapping trains per direction per hour, resulting in twelve conflicts to be solved over the full scheduling horizon. In these cases, Constraints (8.9) ensured the faster one entered first. The six overlaps in direction 1 could be solved by scheduling involved trains on different tracks such as those indicated with orange dashed arrows for tracks 2 and 3 in Figure 8.11 (left) and (middle) respectively.

For track 3, it can be observed that ample capacity was left to plan some of the trains in direction 1 of track 2. Question arises whether it would have been possible to move some trains from track 2 to 3 and use the additional capacity of track 2 to schedule the trains indicated by red full arrows. However, moving the first red train to for example track 3, would cause a conflict with the indicated train (orange dashed arrow) on that one and result in a total delay which is higher than

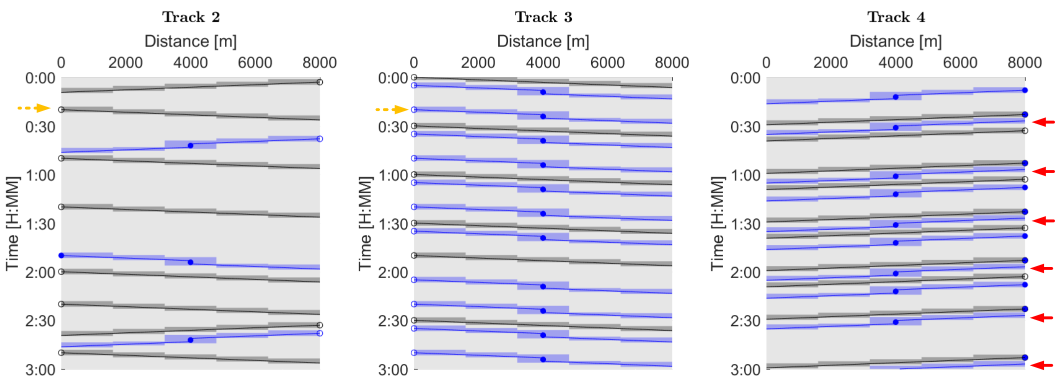


Figure 8.11: Time-distance diagrams for tracks 2 (left), 3 (middle) and 4 (right) for scenario A1 on the third AIP.

Table 8.4: Comparison of objective function values obtained with three or four different computational settings for scenarios reported in Table 8.3. The first columns mention the absolute value, the last ones the relative increase for the MIPfocus ones compared to the 900 s time limit, i.e. $\left(\frac{value_A}{value_{900s}} - 1\right) * 100\%$.

#	Objective function value				Relative increase vs. 900 s (%)	
	Regular 120 s	Regular 900 s	MIPfocus 120 s	MIPfocus 180 s	MIPfocus 120 s	MIPfocus 180 s
A1	1,422	1,422	1,422	-	0%	-
A2	3,229	3,229	3,229	-	0%	-
A3	100,237	100,237	104,145	100,237	4%	0%
A4	1,482	1,482	1,482	-	0%	-
A5	3,729	3,729	3,729	-	0%	-
A6	187,027	174,613	185,388	175,716	6%	1%
A7	3,024	3,024	3,024	-	0%	-
A8	7,941	7,941	7,941	-	0%	-
A9	251,569	242,921	264,192	251,569	9%	4%

the one obtained with the solution reported. To prove this, scenario A10 promoted scheduling all trains in directions 1 and 0 on tracks 3 and 4 respectively. A small penalty ($w^{change} = 0.001$) for deviation from these “desired” track assignments was added to the objective function (7.5):

$$\min \sum_{t \in T} w_t^{cancel} x_t + \sum_{t \in T} w_t^{deviation} dev_t + \sum_{t \in T} w_t^{delay} D_t + \sum_{t \in T} \left(\sum_{b:(t,b) \in S} w_{t,b}^{stop-skip} \right) (1 - \sigma_t)$$
$$+ w^{change} \left(\sum_{t \in T_1} (1 - m_{t3}) + \sum_{t \in T_0} (1 - m_{t4}) \right) \tag{8.12}$$

Due to the original conflicts, at least some rearrangement would be needed. The resulting time-distance diagrams differed strongly from those in Figure 8.11 as the number of trains per track in Table 8.3 already hints at. However, the objective function value was only 0.006 higher, which is completely attributed to the track change penalty of the six trains scheduled on track 2, which confirms the degeneracy of the problem. Although the solution shown in Figure 8.11 may seem not to be the best one, it still is (one of them).

The four additional trains per hour for the mid-frequency timetable (scenario A4) did not result in more overlaps, but showed an increase in total delay of 60 s. This delay followed from the need to reassign more trains to other tracks, which may require an increase in set-up times, as h^{before} and h^{after} are set to 30 s. In the returned solution, trains in direction 0 did not have a dedicated track but dominated the pattern on both tracks 3 and 4 as Table 8.3 reports.

The FIFO heuristic performed as well as the model for scenario A1 as it solved the overlapping release times in exactly the same way: the faster trains entered the corridor before the slower ones. Besides these overlaps, no additional conflicts had to be solved. However, the exact track assignments differed. For increasing frequency in scenario A4, it failed in assigning trains to track 3, which could be used in both directions and performance worsened compared to the model.

According to Tables 8.2a and 8.2b, scenarios A7 to A9 had two additional overlaps per hour per direction. As a result, the objective function value increased for scenario A7. Although the number of overlapping trains doubled compared to scenario A1, the objective function value increased more than proportionally (+123%). This could be explained by the different nature of the overlaps: in scenario A1 pairs included both an IC and an L train, whereas the additional ones in A7 involved only IC trains.

Finally, the solution for scenario A7 displayed the expected balancing of the number of trains over the remaining tracks (see Table 8.3). In light of avoiding conflicting paths in segments S_1 and S_2 (Figure 8.10), one may switch the time-distance diagrams of tracks 3 and 4. Hence, the two outer tracks process traffic in a single direction, the inner one in both. Such a result was obtained by the FIFO heuristic, but it had a 12% increased objective function value.

Closing two tracks

Closing a single track did not result in significant delays: even for the high-frequency timetable (scenarios A7) average delay was less than 3 min. Before, three tracks remained and more room for scheduling heterogeneous traffic was present. This no longer holds true when a second track is closed: to avoid cancelled or stop-skipping trains, one expects that it is better to assign dedicated tracks. Scenarios A2, A5 and A8 included the closure of two tracks for the low-, mid- and high-frequency timetables respectively.

All solutions dedicated tracks to one direction (see Table 8.3). Therefore, delays could be exclusively attributed to two factors. First, the increase in set-up time when scheduling one (or both) train(s) on another track than originally planned. Second, overlap in release times for trains running in the same direction.

The FIFO heuristic applied a similar logic, managing to find the same result as the model for low-frequency scenario A2. For increased frequency in scenario A5, it performed 10% worse, but results were more realistic. The sole difference was found in the assignment of trains to tracks, or better, the assignment of a direction to each track, which was opposite for both solutions. Note that more trains in direction 0 run on track 3 than on track 4 in the mid-frequency timetable: six compared to four per hour (Table 8.2b). Hence, scheduling all trains in direction 0 on track 4 resulted in a larger number of track changes and increased set-up times compared to the other way around. Therefore, the model returned such results.

Two remarks have to be made. First, this did not come into play for scenarios A2 and A8, as both have equal number of trains originally scheduled on track 3. Second, the FIFO solution is actually easier and more intuitive to implement in a real-life situation. Figures 8.12a and 8.12b sketch how trains in both directions have to move over the segments S_1 and S_2 to reach their assigned track in the model's result, and the FIFO solution respectively. Clearly, the latter solution avoided crossing paths

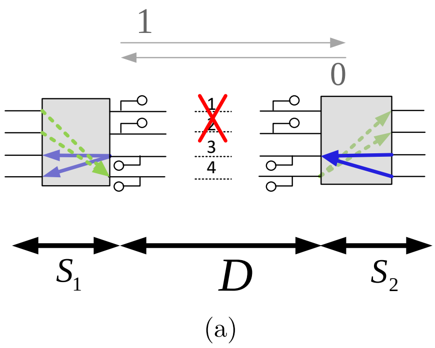


Figure 8.12: Movement of trains over the segments S_1 and S_2 to reach their newly assigned tracks for solutions to scenario B5 obtained with (a) the model, and (b) the FIFO heuristic.

for trains in opposite directions. Although h^{before} and h^{after} may partially account for conflicts between trains assigned to the same track, they do not if tracks differ, and the solution in Figure 8.12b is expected to result in more stable operations. To improve the model's realism in such situations, Section 8.6 presents an extension to account for conflicting paths on S_1 and S_2.

Closing three tracks

When the disruption affected three tracks, the model attempted to schedule 16, 20 and 24 trains per hour on the single remaining one for scenarios A3, A6 and A9 respectively. Additionally, either eight or twelve trains per hour had a planned stop at o_1. Scenarios in Section 7.4 suggest that it is not possible to schedule much more than 12 trains per hour for balanced timetables, without cancelling or skipping stops. Hence, a large number of trains is expected to be cancelled. For other AIPs, cancellation of trains resulted in a difficult balancing between total delay increases and penalties for cancellation, and scenarios had to be rerun with a 180 s time limit for the MIPfocus as discussed before.

For scenario A3 (180), the number of cancelled trains increased compared to A3, but led to a 4% objective function decrease. Batch sizes varied from four to nine as the time-distance diagram in Figure 8.13 (left) shows. As batches in direction 0 grew faster in size, four trains in direction 1 had to be cancelled. Note that some stops were performed midway the disruption timetable, encircled in red. On the one hand, increasing the size of the next batch in direction 1 led to some buffer time, used by the L trains to perform their stop. On the other hand, this may point at suboptimality of the obtained results. Other stops were only performed near the end of the scheduling horizon, but also led to an extension of the disruption timetable after t_{end}. The latter effect increased with increasing frequency as shown in Figure 8.13. Such observation are in line those for the Schaarbeek-Diegem case in Figure 8.9 (left).

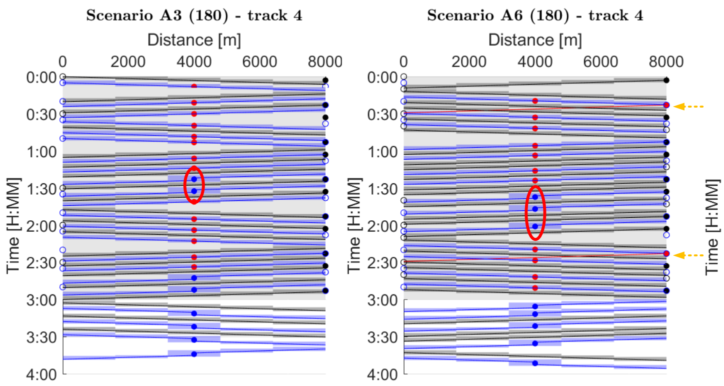


Figure 8.13: Time-distance diagrams for tracks 4 for scenarios A3 (left), A6 (middle) and A9 (right) on the third AIP, obtained with a 180 s time limit for the MIPfocus settings.

In scenario A6 (180), three additional trains could be operated. However, the increase in objective function was not equal to exactly the additional penalty of 64,800 ($9w^{cancel}$), see Table 8.3. The time-distance diagram in Figure 8.13 (middle) shows different batch sizes which initially decreased total delay. However, performing a stop midway the scheduling horizon again increased total delay. As a result, ten trains in direction 1 had to be cancelled versus merely five in direction 0. For the latter group, the orange dotted arrows indicate this happened, amongst others, for trains with overlapping release times.

The increased number of stops midway through the timetable already hints at suboptimality. Indeed, Table 8.4 mentions that the 900 s time limit found a better solution. Its time-distance diagram (not shown here) did not show any stops midway the timetable, but also had smaller batch sizes.

Scenario A9 (180) (Figure 8.13, right) rendered solutions with an objective function value decrease of 4% compared to scenario A9. Despite the high frequency, the remaining track did not get dedicated to one direction. This raises questions about the optimality of the obtained result. Therefore, scenario A11 included track restrictions for trains in direction 1: they could not reach track 4 and only traffic in direction 0 could still be scheduled, equivalent to dedicating track 4 to direction 0. Table 8.3 shows that the objective function value for the optimal solution was higher than for scenario A9 (180): 275,844 versus 263,200 (+5%).

Conclusion

In case of low frequencies and/or a low number of closed tracks, one may expect disruption timetables with only minor delays. For the low-frequency scenarios, two main conclusions could be drawn. First of all, the problem is degenerate as multiple optimal solutions seem to exist. However, this may also be caused by both the set-up of scenarios with equivalent track characteristics, as well as the specific modelling approach of omitting the segments S_1 and S_2. Secondly, the FIFO heuristics managed to find good results for these low-impact disruptions. Hence, one could conclude these basic rules are sufficient for such situations. On the other hand, with increasing frequency, application of the developed model tends to become more interesting.

The scenarios with closure of two tracks illustrated how the model may come up with less realistic results, as conflicting train paths outside segment D are not considered. Therefore, the inclusion of segments S_1 and S_2 as in Section 8.6 may present more promising results. Especially in case in which additional constraints come into play, such as for the Schaarbeek-Diegem scenarios, the model may still be of added value.

Finally, one may conclude that the model is somewhat less suitable for extreme cases with only one remaining track. Allowing some more time to find a solution leads to improved results and is acceptable in such circumstances. Nonetheless, the model clearly outperformed the FIFO heuristic, or approaches which dedicate the track to traffic in one direction. Although results are suboptimal, the model could provide dispatchers with better solutions than merely based on intuition and experience.

8.5.2 Track restrictions

Track restrictions may have a severe impact on solution quality, as has been illustrated for the Schaarbeek-Diegem case (Section 8.4). These restrictions follow from the physical inability to reach some of the tracks using segments S_1 and S_2. The top parts of Figures 8.14a and 8.14b show two possible configurations for these segments, resulting in respectively the following restrictions:

1. Trains on the outer tracks cannot reach the other outer track, e.g. trains originally scheduled on track 4 cannot reach track 1 and vice versa.

2. Trains can only reach tracks neighbouring their originally assigned one, e.g. trains of track 3 cannot reach track 1 and vice versa.

Figures 8.14a and 8.14b indicate which tracks can still be reached in case of a closure of tracks 3 and 4.

Following the results in Sections 8.4 and 8.5.1, some preliminary conclusions may be drawn before solving scenarios. If three tracks are being closed, e.g. tracks 1, 2 and 3, these restrictions merely limit the trains which can still be scheduled on the remaining track. Trains originally scheduled on track 1 cannot be operated for any of both sets. A similar situation amounts for those trains of track 2 under the second set of restrictions.

Results for scenarios 3 and 4 in Table 8.1 with a single closed track indicate that track restrictions did not alter solution quality, but only the track assignment. Additionally, scenarios A1, A4 and A7 in Table 8.3, without restrictions, resulted in limited delays and may resemble degeneracy. Track restrictions are not expected to have significant impact: for both sets, all trains can still reach at least one track, mostly even two.

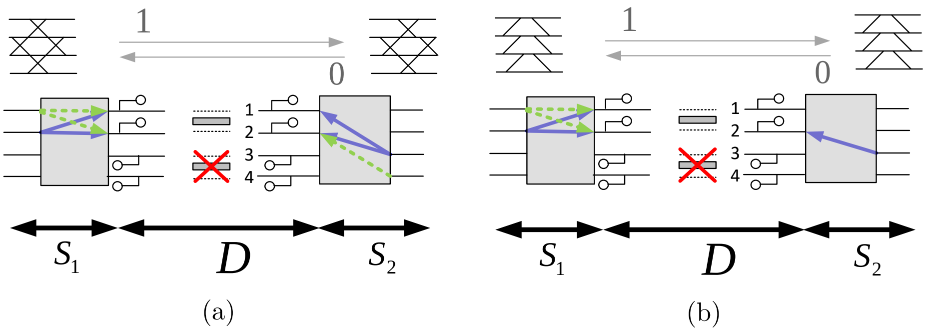


Figure 8.14: Indication of which tracks within segment D can be reached using segments S_1 and S_2 under two possible sets of track restrictions. The disruption includes a blockage of tracks 3 and 4. On top, possible infrastructure configurations for S_1 and S_2 imposing these restrictions are drawn.

Therefore, the scenarios considered here, included the closure of two tracks, i.e. tracks 3 and 4. Figure 8.14 indicates which tracks could still be reached by trains originally scheduled on tracks 1 and 2 (direction 1) and 3 and 4 (direction 0). Table 8.5 reports nine scenarios, specifying the timetable used and applied restrictions. For the latter, 'outer' and 'only next' refer to the first and second sets respectively. Furthermore, only number of cancellation, objective function and track assignments are reported as delay statistics showed only slight variation among scenarios and stop-skipping was never conducted.

Table 8.5: Description of each scenario using three possible sets of track restrictions for each timetable, presenting information on the set-up, i.e. which timetable is used and restrictions applied (as described in Section 8.5.2). Solution statistics report the number of cancelled trains and objective function values. Stop-skipping was not conducted for any scenario. Finally, the number of trains running per track per direction is reported.

#	Timetable (trains/h /direction)	Restrictions	Number cancelled	Objective function value	Number of trains			
					Track 1		Track 2	
					0	1	0	1
B1		none	0	3,229	0	24	24	0
B2	Low (8)	outer	0	3,229	0	24	24	0
B3		only next	12	87,822	0	24	12	0
B4		none	0	3,729	30	0	0	30
B5	Mid (10)	outer	0	4,089	0	30	30	0
B6		only next	12	88,662	0	30	18	0
B7		none	0	7,941	36	0	0	36
B8	High (12)	outer	0	8,661	0	36	36	0
B9		only next	18	133,704	0	36	18	0

All scenarios led to dedicating tracks to specific directions as Table 8.5 reports. As expected, the impact of the first set of track restrictions was rather limited (scenarios B5 and B8) to non-existing (scenario B2). Comparison with scenarios without restrictions indicate that differences were completely attributed to changing the direction of the tracks. Figure 8.14a shows that trains originally scheduled on track 4 could only reach track 2 under these restrictions. Hence, it was beneficial to schedule all trains in direction 0 on track 2. Regardless of the objective function value increase for scenarios B5 and B8 compared to B4 and B7 respectively, the obtained results were more realistic from a practical point of view, i.e. less or no crossing paths segments S_1 and S_2, as discussed in Section 8.5.1 and Figure 8.12.

Under the second set of track restrictions ("only next"), none of the trains originally scheduled on track 4 could reach any other track, illustrated by Figure 8.14b. As a result, 12 to 18 train cancellations could not be avoided.

Conclusion

To conclude, the impact of track restrictions highly depends on their nature, i.e. the configuration of segments S_1 and S_2, and disruption impact, i.e. which tracks are blocked. However, they provide less freedom to the model and may lead to more realistic results, although worse in objective function value. Concluding that imposing restrictions is beneficial to obtain more realistic results may be incorrect, as track restrictions should follow from the physical infrastructure. To increase realism, conflicts on segments S_1 and S_2 have to be taken into account as will be done in Section 8.6.

8.5.3 Other parameters

Results in Section 8.5.1 suggest frequency and timetable structure remain the major parameters, especially in case of overlapping release times, which should be kept in mind in the following discussion.

First of all, the primary impact of the length of the segment D, disruption speed v_D, and the first buffer strategy which added buffer times between all pairs of subsequent trains (Section 6.7), was on running times, and resulting batch sizes. Similar to those observations, one may expect batch sizes to increase on tracks which operate traffic in both directions. Likely, the tendency towards dedicating tracks increases correspondingly: the number of long set-up times between trains in opposite directions could be decreased as such.

An interesting scenario is one in which disruption speed differs per track. As scenarios for the Schaarbeek-Diegem case study illustrated, one could expect larger batches on the track with the longer running times. If tracks would be dedicated, it seems to be good practice to assign prioritized trains to the track with the highest speed, although such a reasoning neglects potential track restrictions and conflicts.

Secondly, the second buffer strategy, which increased time separation between trains in opposite directions, is expected to promote track dedication. This approach could avoid, or even abolish the need for, long buffer times.

Thirdly, unbalances in passenger numbers (Section 6.6), and stop-related parameters, i.e. fraction of passengers, stopping patterns and minimum number of stops, are likely to present the same impact as discussed before. The reader is referred to the respective Sections 6.6 and 7.4.

8.6 Overlapping track assignments in S_1 and S_2

The initial model for the third AIP presented in Section 8.2, made abstraction of segments S_1 and S_2. However, this may lead to conflicts at these locations when implementing the model's results, depending on these segments' exact configuration. Increased headways h^{before} and h^{after} in Equation (8.6) already (partly) accounted for possible conflicts between train pairs scheduled on the same, but not their originally assigned, track. However, if trains get assigned to different tracks, they also may have to cross each other on segments S_1 and S_2, resulting in conflicts if headway times are not sufficient. Previously, this led to less realistic results such as those for scenario A5 in Section 8.5.1 and Figure 8.12.

Section 8.6.1 presents how the model can be adjusted using a specific example, whereafter Section 8.6.2 applies it to several scenarios. Therefore, the disrupted area refers to segments S_1, D and S_2 in the remainder.

8.6.1 Model reformulation

Incorporating these aspects into the model requires a two-step approach. First, track assignments which result in overlapping or crossing trains paths, referred to as *overlapping track assignments*, have to be identified. Next, separating them in time requires additional set-up time constraints. In equations, the abbreviation *seg* will refer to either one of segments S_1 and S_2.

Identifying overlapping track assignments

For a specific configuration of segments S_1 and S_2, overlapping track assignments for trains i and j depend on the originally assigned tracks (k_i^{orig} and k_j^{orig}). Consider the example in Figure 8.15, where A, B and C represent trains t originally scheduled on tracks 1, 2 and 3 respectively, with arrival times t_t and completion times C_t. During the disruption, they are reassigned to respectively tracks 3 ($k_A = 3$), 1 ($k_B = 1$) and 2 ($k_C = 2$). Clearly, any pair of trains has overlapping paths in both segments S_1 and S_2. To avoid knock-on delays, their timings should be sufficiently separated.

Overlapping track assignments are represented by the originally assigned and newly assigned tracks of both trains in a pair. For example, the combination $\left(\left(k_A^{orig} = 1, k_A = 3\right), \left(k_B^{orig} = 2, k_B = 1\right)\right)$ represents one of such assignments for trains A and B, abbreviated as $((1,3),(2,1))$. In general, the sets of overlapping track assignments $\mathscr{C}_{seg}$ for specific configurations of segments S_1 and S_2, contain pairs $\left(\left(k_i^{orig}, k_i\right), \left(k_j^{orig}, k_j\right)\right)$. For the example in Figure 8.15, $\mathscr{C}_{S_1} =$

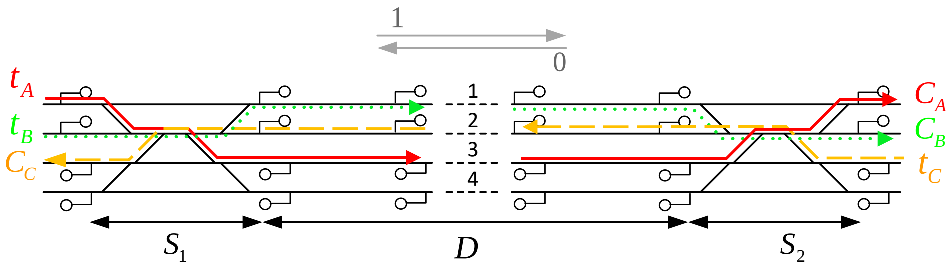


Figure 8.15: Example of the infrastructure for the third AIP with specific configurations for segments S_1 and S_2. Taking these segments into account highlights additional potential conflicts arising from overlapping routes for specific track assignments. Trains A, B and C are representative for trains originally scheduled on tracks 1, 2 and 3 respectively, each of them has an arrival (t_t) and a completion time (C_t).

$\{((1,3),(2,1));((1,3),(3,2));((2,1),(3,2))\}$. Appendix E presents a methodology to determine these sets for different configurations of S_1 and S_2.

Set-up time constraints for overlapping track assignments

The overlapping track assignments do not necessarily result in conflicts if they are sufficiently spaced in time. For each pair of trains, headway times can be determined using the blocking time theory. For those not running on the same track in segment D, these headway times may differ for segments S_1 and S_2. Moreover, the order between trains at S_1 and S_2 may differ: trains could overtake and cross each other when running in respectively the same and opposite directions. Hence, the track-specific order variables q_{ijk} of the model in Section 8.2 are replaced by two sets $q_{ij}^{S_1}$ and $q_{ij}^{S_2}$ for segments S_1 and S_2 respectively. They have a similar interpretation as before: $q_{ij}^{seg} = 1$ if train j is scheduled, not necessarily directly, after train i on segment seg. For example, if train A overtakes train B in Figure 8.15, $q_{AB}^{S_1} = 1$ and $q_{AB}^{S_2} = 0$.

These variables are used to enforce set-up time constraints at both segments S_1 and S_2. Recall that each train is characterized by two timings: its arrival t_t and completion time C_t, occurring at the outer parts of the corridor as shown in Figure 8.15. Only one of both has to be used to define the set-up time constraints for a specific segment, i.e. S_1 or S_2. Set-up times are modelled between so-called *reference events* $E_t^{ref,seg}$, being the timings relevant to the considered segment. A train t running in direction 1, e.g. train A in Figure 8.15, enters S_1 at t_t and exits S_2 at C_t, i.e. $E_t^{ref,S_1} = t_t$ and $E_t^{ref,S_2} = C_t$. The following constraints are required to model conflicts between trains i and j running on different tracks ($k_i \neq k_j$):

$$E_j^{ref,seg} \geq E_i^{ref,seg} + s_{(i,j,k_i,k_j)}^{seg} - M\left(3 - q_{ij}^{seg} - m_{i,k_i} - m_{j,k_j}\right) \quad \forall i,j \in T, \forall \left(\left(k_i^{orig}, k_i\right), \left(k_j^{orig}, k_j\right)\right) \in \mathscr{C}_{seg} \tag{8.13}$$

$$E_i^{ref,seg} \geq E_j^{ref,seg} + s_{(j,i,k_j,k_i)}^{seg} - M\left(2 + q_{ij}^{seg} - m_{i,k_i} - m_{j,k_j}\right) \quad \forall i,j \in T, \forall \left(\left(k_i^{orig}, k_i\right), \left(k_j^{orig}, k_j\right)\right) \in \mathscr{C}_{seg} \tag{8.14}$$

Constraints (8.13) and (8.14) show strong similarities with the set-up time Constraints (8.7) and (8.8) for trains running on the same track during the disruption ($k_i = k_j$). These new ones enforce sufficient time separation for any pair of overlapping track assignments (set $\mathscr{C}_{seg}$) on segment *seg* between the reference events $E_i^{ref,seg}$ and $E_j^{ref,seg}$. For example, Constraint (8.13) between trains A and B on segments S_1 (Figure 8.15) reduces to $t_A \geq t_B + s_{(A,B,3,2)}^{S_1} - M\left(3 - q_{AB}^{S_1} - m_{A,3} - m_{B,1}\right)$.

The set-up times $s_{(i,j,k_i,k_j)}^{seg}$ resemble the minimum headway time between trains, and depend on the specific track assignments. Section 5.5.2 and Figure 5.8 already presented the basic methodology. As trains i and j are not scheduled on the same track on segment D, those parts of the blocking time stairways can be neglected. Assume $E_t^{ref,seg}$ here either represents the release time of train t on the disrupted area (as replacement for t_t) or the departure time of train t if it travels over track k, i.e. $r_t + p_{tk}$. Set-up times are then determined using the start and end of the blocking times on segment *seg* for trains j and i respectively ($\tau_{j,seg}^s$ and $\tau_{i,seg}^e$):

$$s_{(i,j,k_i,k_j)}^{seg} = \left\{\tau_{i,seg}^e - \tau_{j,seg}^s\right\} + E_j^{ref,seg} - E_i^{ref,seg} \quad \forall i,j \in T, seg = \{S_1, S_2\} \tag{8.15}$$

Additional considerations

Four additional aspects have to be considered. First of all, trains scheduled on the same track during the disruption, cannot overtake or cross each other. Hence, deciding on $q_{ij}^{S_1}$ fixes $q_{ij}^{S_2}$ to the same value. Constraints (8.7) and (8.8) can be used as before by replacing the order variable q_{ijk} by either $q_{ij}^{S_1}$ or $q_{ij}^{S_2}$. As such, no hard constraint $q_{ij}^{S_1} = q_{ij}^{S_2}$ is needed. The latter would also forbid overtaking and crossing when scheduled on different tracks. As the segments S_1 and S_2 are explicitly taken into account, the set-up times do no longer have to incorporate h^{before} and h^{after}.

Second, trains i and j running in the same direction and originally scheduled on the same track still have their fixed entrance order. For trains running in direction 1, Constraints (8.9) are still required, but using $q_{ij}^{S_1}$ instead of q_{ijk}. To allow overtaking on segment D, $q_{ij}^{S_2}$ may not be fixed. Similar comments arise for trains running in direction 0, which enter at S_2.

Third, trains changing tracks in segments S_1 and S_2 often do this at a lower speed, and their process times may depend on the assigned track even though the disruption speed v_r is the same on all tracks. No distinction is made between for example changing from track 1 to 2 or from track 1 to 3. Running times can be determined using the method presented in Section 5.5.2.

Fourth, the configurations of segments S_1 and S_2 may differ and hence also the sets $\mathscr{C}_{seg}$ comprise different overlapping track assignments. As a result, one of both may add more Constraints (8.13) and (8.14) than the other.

8.6.2 Assessing the impact of conflicts in S_1 and S_2

Incorporating the segments S_1 and S_2 for the third AIP increases the model's complexity and may result in a large number of additional constraints. Moreover, the dispatcher also has to provide input on these configurations. To illustrate the potential benefit of doing so, this section assesses the impact for a specific case study.

The infrastructure consisted of an extension of the artificial case study in Section 8.5 (Figure 8.10) by adding segments S_1 and S_2 next to the five block sections of segment D, and the stop located in the middle one. Figure 8.16 presents the resulting infrastructure, and indicates that the upper two tracks were blocked, as was the case for some scenarios. Appendix E lists all possible overlapping track assignments for this configuration of segments S_1 and S_2. The timetables were the same as provided in Tables 8.2a and 8.2b, but with entry times denoting the entrance at S_1 and S_2 for trains in directions 1 and 0 respectively.

Table 8.6 presents information on the set-up of the scenarios of set C, which included the closure of one or two tracks. In the latter case, it could be either tracks 1 and 2, or 1 and 4. Additionally, odd numbered scenarios ("switches") did not include the Constraints (8.13) and (8.14) but only accounted for the (possibly) increased running times over S_1 and S_2, as opposed to the even numbered ones ("conflicts"). Next, delay statistics and the number of cancellations are reported, which constitute the objective function value. Finally, the number of trains assigned to each track is reported. To serve as reference, results for scenarios of set A (Section 8.5.1) are presented. Those for blockages of both tracks 1 and 4 were not reported before.

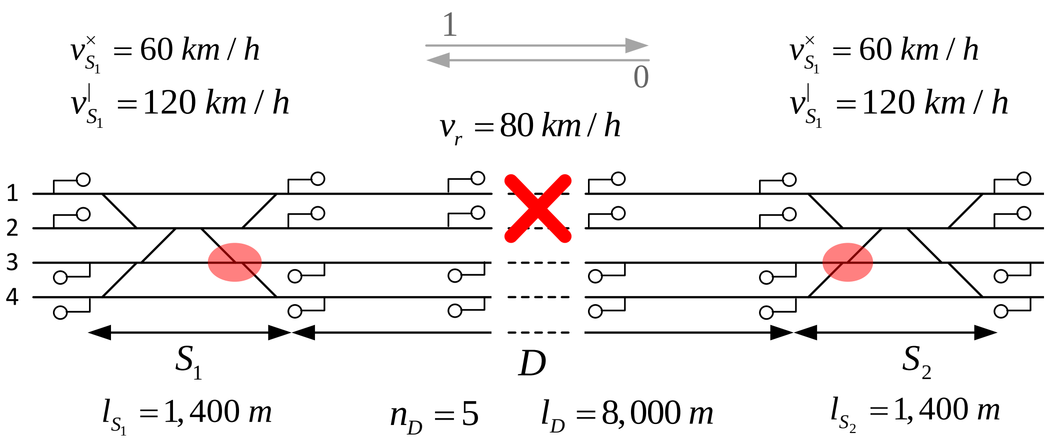


Figure 8.16: Representation of the infrastructure used for the artificial case study to assess the impact of including segments S_1 and S_2 for the third AIP. Segment D holds five block sections as in Figure 8.10, with a stop o_1 located in the third one.

Table 8.6: Description of each scenario when varying both frequency and number of closed tracks for inclusion of the segments S_1 and S_2. It presents information on the set-up, i.e. which timetable is used and the number of closed tracks. Solution statistics report total delay, together with the number of cancelled and skipped stops, constituting the weighted objective function. Finally, the number of trains running per track per direction is reported. Empty cells indicate either a 0 value, or unavailability.

#	Timetable (trains/h /direction)	Tracks closed	Aspects included	Total delay (s)	Number cancelled	Number stops skipped	Objective function value	Number of trains					
								Track 2		Track 3		Track 4	
								0	1	0	1	0	1
(A1)			-	2,982			1,422	4	7	0	17	20	0
C1		1	switches	4,482			1,626	0	24	12	0	12	0
C2	Low (8)		conflicts	4,482			1,626	0	24	12	0	12	0
(A2)			-	6,037			3,229			0	24	24	0
C3		1 & 2	switches	10,263			4,863			0	24	24	0
C4			conflicts	57,353		1	53,033			11	15	13	9
(A4)			-	3,042			1,482	0	17	0	13	30	0
C5		1	switches	4,602			1,746	0	30	18	0	12	0
C6			conflicts	4,602			1,746	0	30	18	0	12	0
(A5)			-	7,317			3,729			0	30	30	0
C7	Mid (10)	1 & 2	switches	13,625			6,797			0	30	30	0
C8			conflicts	29,275	12	14	124,068			6	30	12	0
			-	6,084			2,964	#	0	0	30		
C9		1 & 4	switches	8,874			3,294	0	30	30	0		
C10			conflicts	8,874			3,294	0	30	30	0		
(A7)			-	5,364			3,024	0	29	29	0	7	7
C11		1	switches	7,314			3,030	0	36	18	0	18	0
C12			conflicts	7,314			3,030	0	36	18	0	18	0
(A8)			-	12,309			7,941			0	36	36	0
C13	High (12)	1 & 2	switches	19,387		4	15,451			0	36	36	0
C14			conflicts	65,161	16	14	189,289			11	27	18	0
			-	10,608			5,928	0	36	36	0		
C15		1 & 4	switches	14,334			5,964	0	36	36	0		
C16			conflicts	14,334			5,964	0	36	36	0		

First of all, results reported for scenarios with a single blocked track strongly differed in track assignments compared to their counterparts of set A. Some of the latter were already identified as unrealistic because of the high number of overlapping paths in S_1 and S_2, potentially leading to conflicts. However, results obtained for odd (C1, C5 and C9) and even numbered scenarios (C2, C6 and C10), i.e. those without and with constraints to avoid conflicts respectively, were exactly the same. Merely incorporating the additional travel time over the segments S_1 and S_2, avoided that trains originally scheduled on tracks 3 and 4 were moved to another track, which would increase running times. All trains in direction 1 were assigned to track 2, meaning overlapping paths were only present for trains on the same track, handled by the old set-up time constraints.

When closing tracks 1 and 2, all trains in direction 1 either had to move to tracks 3 and 4, or had to be cancelled. Regardless of the specific track assignment, routes overlapped with trains originally scheduled on track 3 in both segments S_1 and S_2 for the specific configuration: The red shaded areas in Figure 8.16 indicate the smallest possible overlaps. Table 8.6 reports different results for any of the timetables used when considering conflicts (scenarios C4, C8 and C14) or not (C3, C7 and C13).

Especially scenario C4 presented an interesting outcome: tracks processed trains in both directions. Figure 8.17 present time-distance diagrams for tracks 3 and 4 of the results obtained for scenario C4. The train encircled in red on track 3 skipped its stop. This was done to avoid a potential conflict on segment S_1 with the first indicated train on track 4. Performing the stop would have delayed the indicated train even more, resulting in a cascade of additional delays for the remainder of the scheduling horizon. Additionally, the next train on track 4 (also indicated in red) did not follow directly after the first one, as another stopping train in direction 0 on track 3 had to pass segment S_1. Also for track 3 such effects were observed, e.g. the orange dotted arrow in Figure 8.17 (left).

With increasing frequency, such effects expanded: in scenario C8 a relatively small number of trains operated on track 4: the originally scheduled ones. Time-distance diagrams showed that trains originally scheduled on track 3 could easily be scheduled in-between them. However, this would not result in less train cancellations as routes would remain conflicting in segments S_1 and S_2 with those train originating from tracks 1 and 2. Scenario C7 on the other hand, planned all trains as conflicts were not taken into account. The increased running times were the single fundamental difference with scenario A5. Similar comments can be made on results for scenarios C13 and C14 with the high-frequency timetable. Whereas segment D was the main bottleneck for the first and second AIP and the third one without modelling conflicts, one may conclude segments S_1 and S_2 have inherited that role for the third AIP.

A final set of scenarios included the closure of two tracks with the mid- (C9 and C10) and high-frequency timetables (C15 and C16), but this time for the outer ones, i.e. tracks 1 and 4. Similar to scenarios with one closed track, results were the same regardless of taking conflicts into account (C10 and C16) or not (C9 and C15). Due to the lower process times for trains running straight over segments S_1 and S_2, trains originally scheduled on tracks 2 and 3 remained on their own track. Trains originally scheduled on tracks 1 and 4 had to be reassigned and were moved to the tracks on

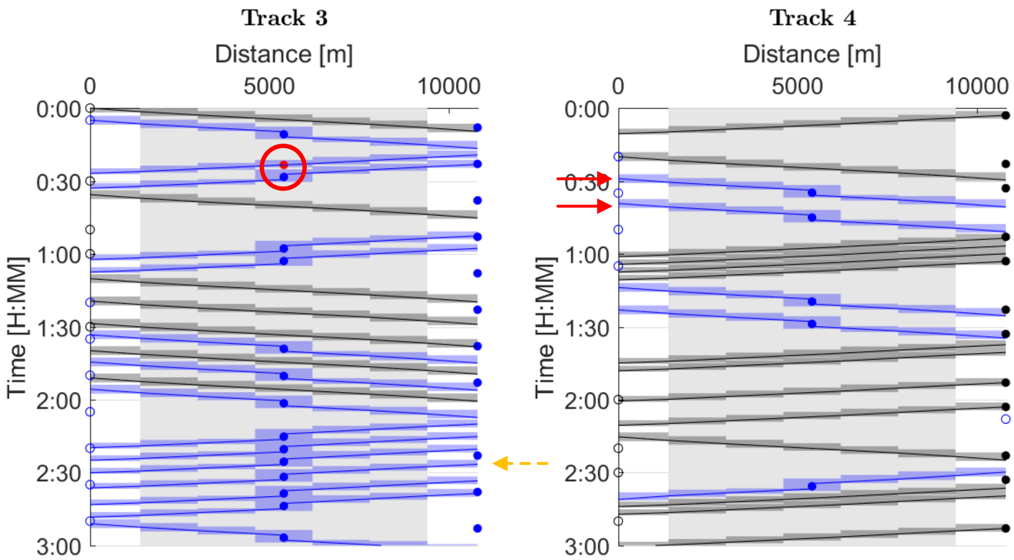


Figure 8.17: Time-distance diagrams for tracks 3 (left) and 4 (right) for scenario C4 on the third AIP.

which trains in their respective directions had already been scheduled because of the lower set-up times. As a result, there were only overlapping routes and resulting conflicts for trains on the same track, which were handled without the additional Constraints (8.13) and (8.14). Hence, including these in scenarios C10 and C16 did not alter results. One can regard such combination of infrastructure on the one hand, and disruption nature on the other hand, as "favourable" in light of conflict avoidance: fewer trains had to be cancelled.

8.6.3 Conclusion

Incorporating the configuration of segments S_1 and S_2 in the initial model for the third AIP, tends to improve the model's potential to produce more realistic disruption timetables. Depending on both the exact configuration of the segments, and the arising track closures, merely accounting for the increased running times over these segments may already result in fewer or no conflicts. Nonetheless, it seems not to be straightforward to predict such situations, especially for more complex configurations. Moreover, explicitly accounting for the overlapping paths may avoid degeneracy of the problem by eliminating a large number of solutions which might have caused conflicts outside of segment D.

Although the model development and the artificial case study handled specific configurations of the segments S_1 and S_2, both the presented methodology for overlapping track assignment identification (Appendix E), as well as the constraints to incorporate them, are considered to be generic. It is likely they could be applied to a large range of infrastructure configurations, including for example fly-overs.

8.7 Conclusion

This chapter extended the models developed in Chapters 5 and 7 by modelling disruptions on N-track corridors as a parallel-machine scheduling problem. The increased complexity lies within assigning trains to tracks, which may have different disruption speeds. Copies of all trains are created for each track, and track-specific set-up time constraints get relaxed if one of both trains does not run on that track.

The switches S_1 and S_2 become junctions and may have highly varying infrastructure configurations, whereas these were more or less standardized for the first and second AIPs. To account for this, some buffer time is included if trains get assigned to another track. Secondly, track restrictions are introduced to capture the reachability of tracks from the bordering stations.

One would expect remaining tracks to get a dedicated direction. Applying the model to a practical case study led to counter-intuitive but plausible observations. First of all, the problem is degenerate for low-frequency timetables, or a small number of closed tracks. Assigning trains to different tracks, or swapping complete track time-distance diagrams, does not lead to different objective function values. Secondly, when track restrictions apply for one or more trains, other ones may cluster around these to form batches in opposite directions on the same track. An example are the international trains which get a prioritized treatment.

Increasing the number of closed tracks for the artificial case studies confirmed previous observations, and led to an increased tendency to dedicate tracks to one direction. For severe disruptions, e.g. affecting three tracks out of four, the MIPfocus settings did not manage to return good results within 120 s, and it is advisable to increase the time limit. Comparison with the FIFO heuristic suggests that this heuristic is sufficient to produce disruption timetables which are not (much) worse than the model's ones, in case of low-impact disruptions. Secondly, the latter tends to generate more realistic results from a practical point of view.

To increase both the realism of the model's disruption timetables, and its value over current practice, a method to account for overlapping track assignments on segments S_1 and S_2 was developed. Results suggest this extension is of high added value. Although they tended to be worse than those before, realism of the resulting disruption timetables seemed to significantly increase for two reasons. One, it accounted for the increased running times when changing tracks, which is potentially sufficient to avoid conflicts for some disruptions. If not, the second benefit was the time separation of conflicts.

Similar to the stop-skipping measure in Chapter 7, track assignment and especially conflict avoidance on the junctions, may complicate decision-making during disruptions on the third AIP. Therefore, one could state the model is of added value for current practice. Next to the parameters mentioned at the end of Section 7.5, dispatchers should provide the model with the number of tracks, the closed ones, disruption speed, and when applicable, the (un)availability of platforms at stops. Accounting for the junctions could either be done by selecting the specific one out of a predefined choice set, or roughly sketching the configurations, after which the methodology to identify overlapping track assignments (Appendix E) is applied .

Chapter 9

Discussion

This chapter summarizes the answers to the research questions formulated in Section 3.2, thereby critically reflecting on the work conducted and presented within this thesis, its results, and possible pathways for future research. Section 9.1 relates the work with both current practice and state-of-the-art in railway disruption management. Next, Section 9.2 reflects on the value of AIPs, and the applicability of the developed models. A brief overview of the impacting parameters is provided in Section 9.3. Based on this discussion, Section 9.4 concludes with the practical relevance of the outcomes of this thesis.

9.1 Relation with current practice and state-of-the-art models

In contrast to current practice in other countries, dispatchers at Infrabel do not have predefined plans at hand to tackle disruptions on the Belgian network. Whereas such plans are typically not directly implementable or do not exist for each possible situation, researchers recognize their value to quickly react to an arising disruption. On the other hand, one should acknowledge disruptions tend to be unique along several aspects. Nonetheless, the models developed in this thesis are considered to be sufficiently flexible to handle disruptions on their respective AIPs. No similar models have been found in literature, supporting that this thesis presents a potentially valuable extension of the state-of-the-art.

Although Nakamura et al. [36] use a distinct approach by predefining parts of the solution for specific infrastructure configurations, their work seems to be most similar to this thesis. Perhaps, the largest difference lies within the increased flexibility of the models presented here. Yet, it should be acknowledged that the latter are more likely to result in conflicts outside of the considered corridor. Despite such differences, their evaluation framework may be used to apply the models in a practical set-up, and deal with the arising conflicts. The latter could for example be achieved with the simulation tool of Van Thielen et al. [56].

The phases of the bathtub concept provide a good basis to discuss the model. During the first transition phase, the developed models are likely to provide better solutions, but especially faster than the current ad-hoc scheduling at Infrabel. However, dispatchers would still have to wait some time before retrieving the solution, meaning that the first minutes are not being covered for. To deal with this lack, one could advise a certain approach for this first short time, e.g. a FIFO one, and take this into account when running the model. As results are obtained quickly, only a limited number of trains should be scheduled in such a way, and results should not be affected too much.

During the second, stable, phase, newly arising delays may affect the generated disruption timetable, which could be partially covered for by requesting the model to include buffer times. Nonetheless, continuous (automated) monitoring would be needed to ensure delays do not impact (too much) on solution quality. This issue could be tackled by embedding the AIP models in a rolling horizon optimization, allowing the dispatcher to re-run the model at either regular intervals, or upon solution quality deterioration. The work by Quaglietta et al. [46] illustrates such an approach could be very effective in dealing with information updates. Finally, the developed models do not account for returning to regular operations in the third phase, which is considered to be less problematic and therefore not covered here.

The developed models tackle the timetabling stage of the rescheduling process; rolling stock and crew have not been accounted for. However, as the presented models outperform the heuristics, one may expect less extensive rolling stock and crew rescheduling for disruption timetables generated by the model. Dispatchers could also anticipate potential infeasibilities by increasing the priority of specific trains, i.e. altering their penalties. Potentially, the AIP models could be integrated in the (modular) iterative frameworks as presented by Besinovic et al. [6] and Dollevoet et al. [17]. For short-term application within the Belgian railway system, possibilities to interact with tools of NMBS should be explored. This was outside the scope of this thesis and is left for future research.

9.2 Disruption management models for archetypical infrastructure pieces

This thesis research developed models for three AIPs of increasing complexity, frequently encountered on the network, outside station areas:

1. Double-track corridors with a double switch on both sides, without stops
2. Double-track corridors with a double switch on both sides, including stops
3. Multi-track corridors, including stops

The question remains how frequent these AIPs are encountered on a given network, making abstraction of the probability of disruptions on them, as no data exists. Figures 9.1 and 9.2 identify the occurrences of the three AIPs considered here for two parts of the network, respectively the area Gent-Antwerpen-Brussels, and south of Brussels. Note that the former area might be less representative due to the high network density: Brussels is the main bottleneck of the Belgian railway network and requires other techniques. Additionally, a number of (larger) station areas are located here. Therefore, the area in Figure 9.2 seems to be more relevant.

Investigating these figures, and the complete technical map of the Belgian railway network [26], it becomes clear the first AIP does not occur that much. However, it was a valuable starting point for the second AIP, which is much more common. One expects this AIP could potentially cover up to 50% of the double-track corridors between the major stations. The other half does not fit within the picture due to, for example, the presence of junctions and stops with three tracks. Finally, corridors with more than two tracks seem to be far less common on the network, and those existing cover rather short distances. Therefore, this AIP may potentially account for up to 5% of the open-track corridors. However, these corridors often also hold more traffic than double-track corridors, which allows more degrees of freedom when optimizing operations and is expected to lead to less intuitive results. Moreover, with the planned extensions, e.g. doubling of tracks between Gent and Brugge (line 50A), it should become more valuable. In addition, Figures 9.1 and 9.2 indicate potential AIPs for future research such as junctions of merging double-track corridors, and short stretches or stops with three tracks, which are often used for overtaking operations.

The machine scheduling problem was selected to model the timetable rescheduling over simulation for two main reasons. First of all, after the literature study, one discerned it as a commonly adopted approach in existing state-of-the-art models for the Train Traffic Rescheduling problem. Secondly, it presents a highly flexible framework to cover for a range of problem extensions. This thesis seems to confirm the latter statement as it was possible to account for headways, i.e. sequence-dependent set-up times, stop-skipping, i.e. controllable process times, and cancellation, i.e. rejection, in both a single- and parallel-machine scheduling model. Observations for several scenarios pointed out that stop-skipping and track assignment seem to be subtle measures often leading to counter-intuitive results. This suggests that limiting the sets of measures which would be required as input for simulation, could have neglected good sets of measures. Hence, the choice for a mathematical model was justified.

Unfortunately, one also had to deal with the downsides of optimization models. The model's complexity quickly increased, often leading to excessive computation times. However, choosing other computational settings mitigated this to a certain extent, allowing to generate solutions which tended to be both good, if not optimal, and understandable. Future research could focus on improving computational performance, which was not within the scope of this thesis. Nevertheless, the fact that most solutions for the practical case studies on the first and second AIP could be proven to be optimal, supports the conclusion that the models may effectively deal with real-life cases. Possibly, the structure of the artificial case studies was a contributor to computation time increases.

To make the models generally applicable, two major assumptions had to be made. First of all, stations aligning the corridor were assumed to have sufficient storage capacity for the queues building up at both ends. Often, less than five trains were

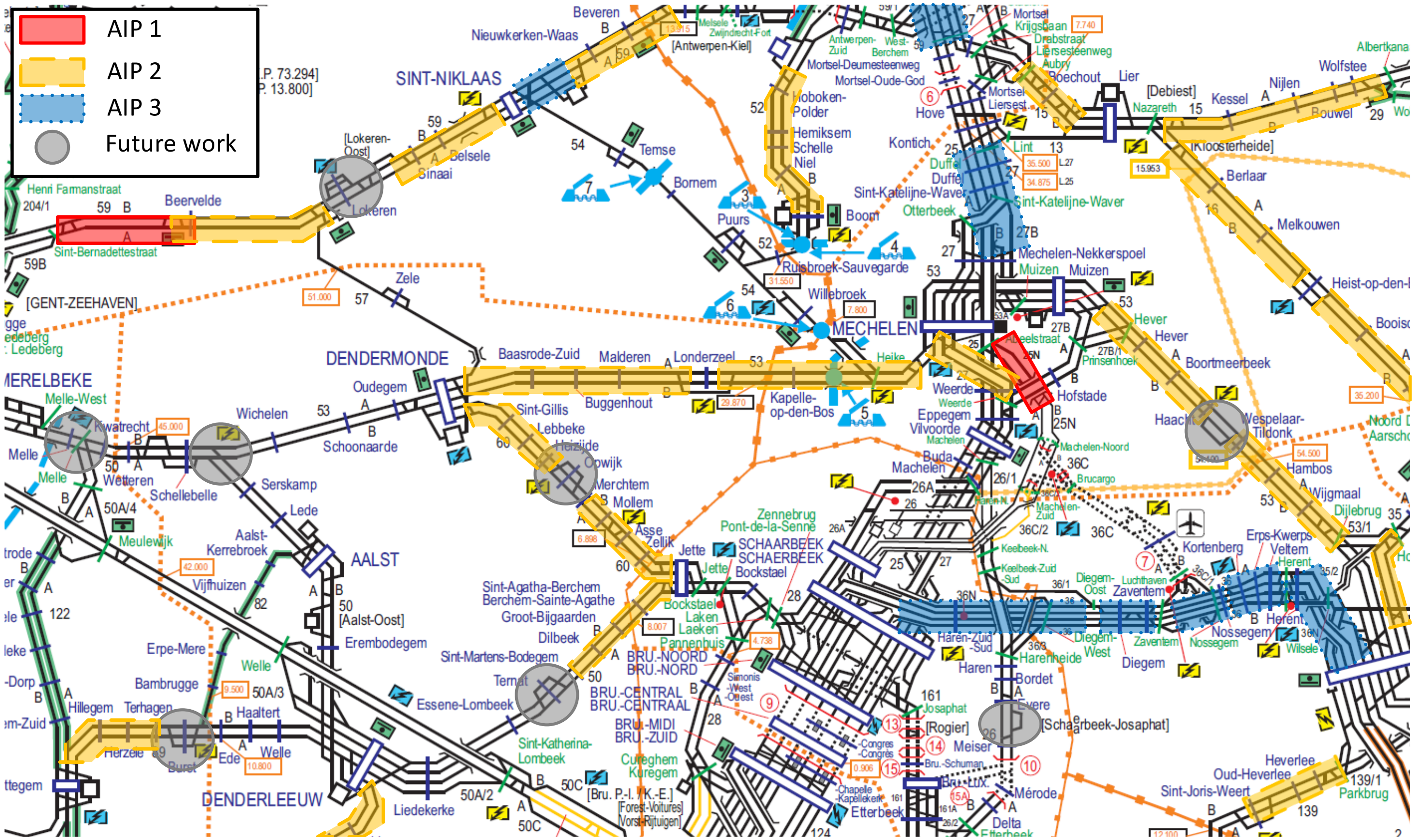


Figure 9.1: Occurrences of the AIPs on a large part of the Belgian railway network, situated between Gent, Antwerpen and Brussels, for which models were presented. Some do not resemble presented AIPs exactly, but may be covered by them. In addition, potential AIPs not covered in this thesis, are indicated for future research. A legend is provided on the left top part. Adapted from [26].

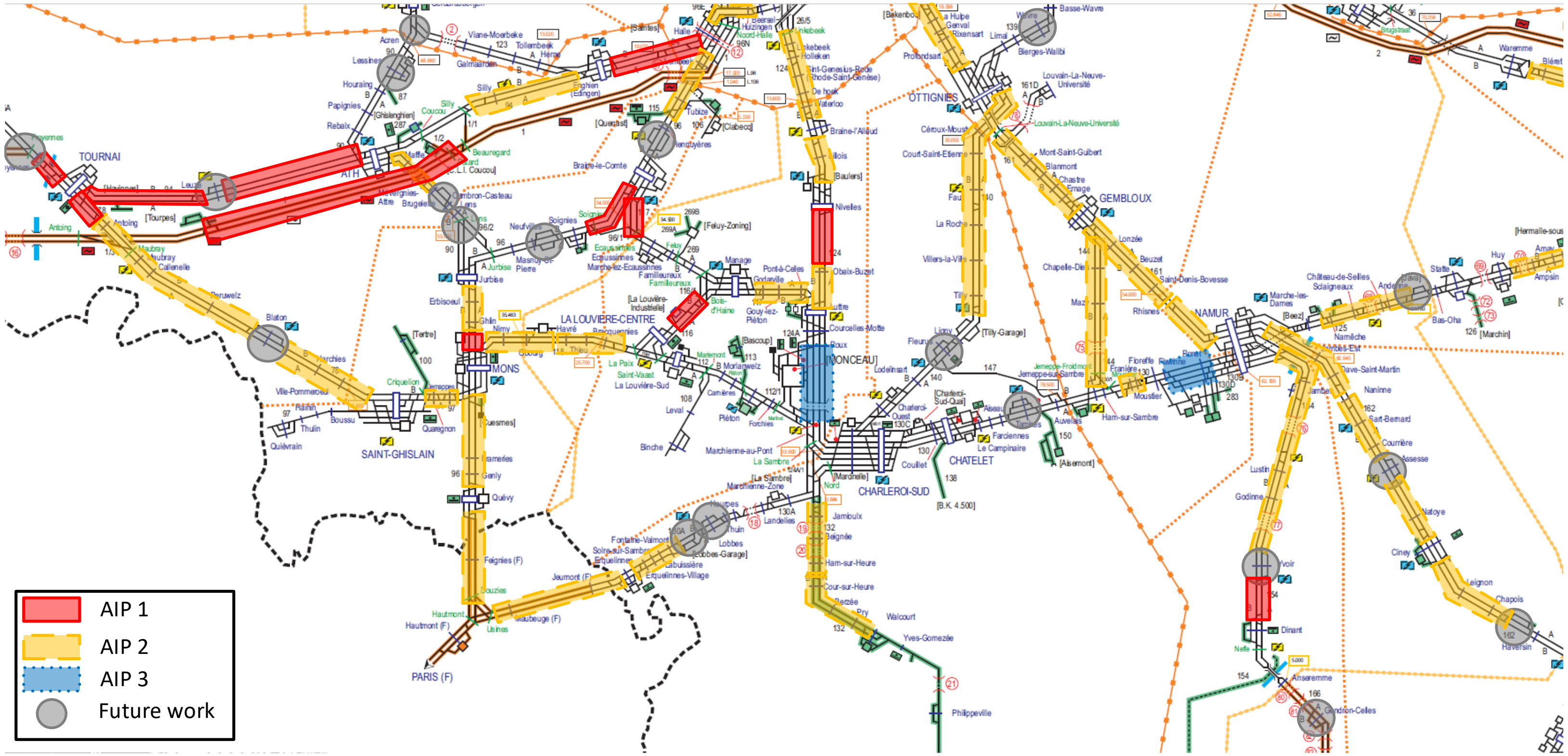


Figure 9.2: Occurrences of the AIPs on a large part of the Belgian railway network, south of Brussels, for which models were presented. Some do not resemble presented AIPs exactly, but may be covered by them. In addition, potential AIPs not covered in this thesis, are indicated for future research. A legend is provided on the left top part. Adapted from [26].

queuing at the same time, which suggests this is a valid assumption, as trains may also be buffered on the segments between the stations and the disrupted area. For cases which may violate this assumption, strategies to buffer trains at downstream stations may have to be developed.

Secondly, the considered corridor is not an isolated part of the network: interactions with other corridors may cause additional conflicts. Incorporating the segments S_1 and S_2 for the third AIP, illustrated it is better to consider an as large as possible part of the network. However, other corridors could not be incorporated in such a general model. One recognizes the potential benefit of an interaction between the develop optimization models and a simulation tool as means to evaluate the solution's impact on the remainder of the network. Future research could simulate the effect of the measures derived by the optimization model within the simulation tool of Van Thielen et al. [56].

Although it would have been of high added value for this thesis research, evaluation of the obtained solutions by comparison with current practice, was not feasible within the time-frame. To circumvent these issues, comparison with heuristics seemed to be most plausible. While the FIFO approach might have been somewhat naive for the first and second AIP, the (max,wait) heuristic incorporated problem-specific knowledge. The models' main merit over the heuristics lied within the capability to efficiently balance between different measures by accounting for interactions. Given the absence of decision support tools, these observations could likely be extrapolated when comparing the model's performance with current practice. Nonetheless, practical evaluations are still highly desirable.

9.3 Parameter and service constraint impact

This thesis developed the models using practical case studies for each AIP, which allowed to iteratively identify additional (model) requirements based on observations. In case only artificial case studies would have been used, considerations such as stops on segments S_1 and S_2 may had been neglected. On the other hand, the artificial case studies allowed a structured analysis of the finalized models, and the impact of their parameters. For detailed conclusions on parameter impact, the reader is referred to the (intermediate) conclusions of Chapter 6, Section 7.4 and Section 8.5, for respectively the first, second and third AIP. The combined approach of practical and theoretical case studies seem to have been of high added value to this thesis.

Without doubts, the primary impacting factor is timetable structure. Results suggest it was the only limiting factor in creating large batches, which is in line with Abril et al. [1], who present an assessment of single-track line capacity. They conclude that running all trains in one direction after each other, followed by the other direction, i.e. maximize batch size, leads to maximum capacity for bidirectional traffic. Within this thesis, release times following from timetable structure, rendered such practice suboptimal. Moreover, Abril et al. [1] prove that including stops may severely hamper the number of trains which can be operated. Stop-skipping in case of limited buffer times and small penalties, seems to resemble their conclusions.

The practical case studies illustrated how service constraints may potentially lead to counter-intuitive results. For example, including passenger numbers, i.e. differing penalties per train, often violated previously identified "rules". Since dispatchers mostly take such aspects into account when constructing a disruption timetable, it is somewhat difficult, if not impossible, to distil general measures that should lead to good disruption timetables in any, or a lot of, scenario(s).

Moreover, the scenario sets on the artificial case studies varied one parameter at a time, whereas the practical case studies suggest strong interactions may exist between them. Assessing all possible combinations of parameters was not considered feasible, nor instructive. Based on the observations, one could carry out a qualitative assessment of the combined impacts.

Consequently, one may conclude that models developed here, could present a valuable contribution to current practice at Infrabel, as they are capable to account for a wide variety of aspects. Yet, only a limited number of parameters should be provided to the model, as other ones, such as the stop-skipping penalty multiplier, could be automatically derived from timetable data.

9.4 Practical relevance

This section discusses a "practical" answer to the central research question:

> How can models with few parameters support dispatchers' decision-making by providing solutions during disruptions with partial blockage on archetypical infrastructure pieces of the Belgian railway network?

Figure 9.3 presents a possible framework to employ the developed models within the practical set-up at Infrabel, combining ideas developed throughout the thesis research, gathered from the frameworks of Nakamura et al. [36] and Quaglietta et al. [46], and the current dispatching context. Three components interact with each other: the dispatcher, the existing TMS, and the "AIP models".

Upon observation and identification of a disruption, the dispatcher collects information on a number of parameters, possibly accompanied with service limitations, and provides these to the AIP models. The latter retrieve the prevailing traffic situation from the TMS, i.e. a timetable holding the expected arrivals at the disrupted area. After a certain time, e.g. 2 min, the model returns the best possible, if not optimal, solution obtained, and presents the associated disruption timetable to the dispatcher. After acceptance, or minor manual changes, the measures are transferred to the TMS, which could evaluate the impact of the AIP model's solution on the remainder of the network. Preferably, it includes a method to solve prevailing conflicts as the one developed in [56].

Using the existing traffic situation, enables the model to account for delays as they are predicted at the start of the scheduling horizon. However, during the implementation and execution of the disruption timetable, new information on both the disruption, e.g. extended duration, or on delays of incoming trains may become

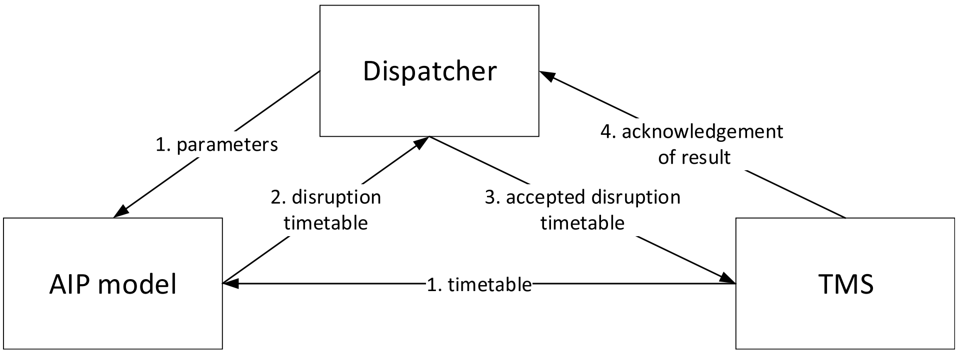


Figure 9.3: Framework for the practical application of the (AIP) models presented within this thesis, at infrastructure manager Infrabel. The AIP model receives input from both the dispatcher and the TMS, after which it returns a disruption timetable. The latter is implemented in the TMS after acceptance.

available. Hence, either the dispatcher, or the TMS should be able to request a new iteration by the AIP model, resembling ideas of rolling horizon approaches.

Figure 9.4 lists the parameters dispatcher have to, or could, provide to the model, depending on the AIP affected by the disruption. Each AIP added a new dimension to the already developed ones, inheriting the input requirements. For the first AIP, the configuration, i.e. orientation, of the switches and their length would be required, together with information on the closed track, a (lower) speed limit, and the scheduling horizon. The configuration of the segment D should include information on the length and number of block sections. In case stops are located along the corridor, also information on their location and fraction of passengers, is required by the model. In addition to these, specifications for the third AIP require information on the number of tracks, on which ones are closed, and the (track-specific) disruption speed. Although not strictly required, providing the model with the configuration of segments S_1 and S_2 (may) significantly improve(s) results.

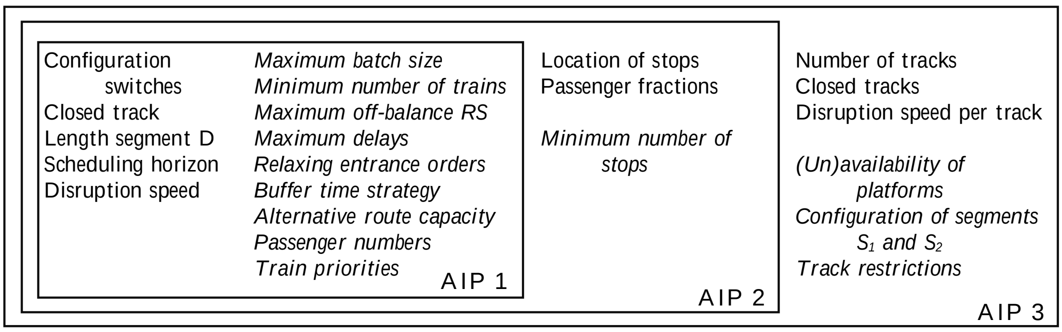


Figure 9.4: Overview of parameters which the dispatcher has to provide to the model in order to retrieve a disruption timetable, depending on the considered AIP. As the latter increases in complexity from 1 to 3, only the supplementary parameters are mentioned for AIP 2 and 3. Non-essential parameters are printed in italics.

One acknowledges dispatchers take many more aspects into account than reflected by these “essential” parameters, which is recognized as one of their major strengths. The “service constraints” developed throughout this thesis, should allow them to impose additional restrictions. Consequently, these constraints combine the best of both: the models’ effectiveness in balancing between measures on the corridor, and the dispatcher’s knowledge on both influential aspects situated outside of it, as well as passenger requirements. For example, imposing a maximum delay could be used to guarantee important passenger connections at the stations aligning the corridor. Dispatchers could also prioritize trains by altering their penalties.

Chapter 10

Conclusion and Future Research

This thesis introduced the concept of archetypical infrastructure pieces (AIPs) as a new approach to generate disruption timetables at runtime. The developed models are complementary both to the state-of-the-art as an unexplored approach for disruption management, as well as to current practice at Infrabel, where dispatchers have to rely on their experience.

The first AIP included a double-track corridor with double switches at both sides, which allowed the identification of a number of parameters which could potentially impact the resulting disruption timetables. The final selection consisted of the timetable, length of the affected corridor, closed track, scheduling horizon, and possible speed restrictions, which should allow to apply the models on a large number of instances, requiring limited effort by the dispatcher. A single-machine scheduling model with sequence-dependent set-up times and rejection was developed, with a weighted objective function penalizing delays and cancellations. Iterativy modelling practical case studies allowed to formulate additional service constraints, resembling aspects dispatchers may consider. In absence of historic data on measures, an extended FIFO, and a newly developed heuristic served as evaluation for the model's performance. A theoretical analysis on artificial cases studies allowed to assess parameter impact, and suggested timetable structure to be the main determinant.

Because of its limited incidence on the complete network, the first AIP was extended to include stops along the corridor, requiring additional input on stop locations and passenger fractions. Skipping stops along the disrupted part of the corridor reduces running times of stopping trains, and may mitigate delays to all other trains. The stop-skipping measure was incorporated by extending the first model with discretely controllable process times. Application to both practical and artificial case studies illustrated the subtle interaction between buffer times and performing stops.

A final AIP consisted of a multi-track corridor with a number of tracks affected by the disruption, and track assignment came into play. The initial parallel-machine scheduling model neglected the junctions aligning the corridor, which resulted in degeneracy of the problem, less realistic solutions, and only a limited benefit over the extended FIFO heuristic. To improve performance, a methodology to identify potentially conflicting track assignments was developed. Its outputs were used to explicitly model conflicts on the aligning junctions. Results tended to be much more realistic, and strengthen the belief in the model's practical applicability.

Assuming the affected corridor could be considered in isolation from the remainder of the network, is inherent to the AIP concept. The use of blocking time theory should ensure the disruption timetable is conflict-free on the considered corridor. Nonetheless, (minor) conflicts are likely to arise outside of it, and additional rescheduling may be required. Future research could go into on the implementation of the generated disruption timetable in the existing TMS at Infrabel, which detects conflicts, preferably supplemented with a method to solve them.

Other future work on this subject may primarily focus on two additional pathways. First of all, the generated measures may render rolling stock and crew schedules infeasible. One could look into embedding the AIP models either in existing tools at NMBS, or in an iterative framework for all planning stages. Secondly, this thesis illustrated the AIP concept seems to be a powerful tool to cover for disruptions on multiple parts of the network. Additional work may identify other AIPs and either develop new models, or extend the presented ones.

To conclude, dispatchers could employ the developed models to quickly generate disruption timetables after providing a limited number of parameters. Their knowledge about aspects not directly related to the corridor, and passenger expectations, may be captured by the service constraints. As such, the combination of dispatcher's experience and the model's strength to balance between measures, is likely to improve current disruption management practices, and most importantly, passenger satisfaction.

Appendices

Appendix A

Time-Distance Diagrams

Throughout this thesis, *time-distance diagrams* are used to visualize either the original or the disruption timetable. This appendix presents an overview of all aspects of these plots by means of the example in Figure A.1, which shows the first hour of the time-distance diagram associated with the result obtained for scenario 4 of the Oostkamp-Aalter case study (Section 7.3). Throughout this appendix, numbers refer to the indications on Figure A.1.

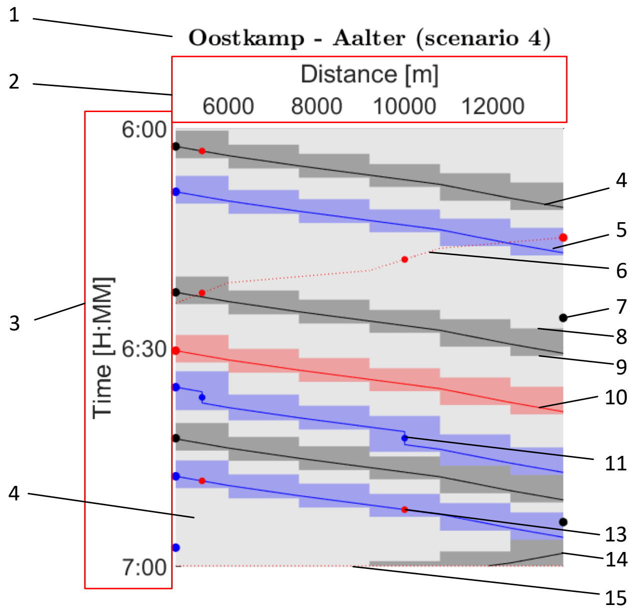


Figure A.1: Example of a time-distance diagram, with indication of all types of information they contain. Illustration by means of first hour of scenario 4 for the Oostkamp-Aalter case study Section 7.3.

General plot information

The title of each time-distance diagram (1) may contain information on either the case study, the specific scenario, parameters settings or the track number (in Chapter 8). Information not mentioned in the title, is added to the captions or considered to be abundant within the section's context.

The horizontal axis indicates the distance along the corridor (2) starting from the far left end. For the example in Figure A.1, this is not 0 m, as the segment A on the left of the corridor also contained three block sections. For the sake of clarity, these are mostly not incorporated as they do not belong to the disrupted area. The vertical axis represents the time dimension (3), either starting at the beginning of the disruption, or at 0:00 for the practical and artificial cases respectively. Most time-distance diagrams show the full time horizon (commonly 2 or 3 h), but some of them also plot the time after. To split the full time horizon into parts, thin horizontal dotted line (15) indicate the end of 1 h intervals.

The grey shaded background (4) presents the spatial and temporal characteristics of the disruption, i.e. which part of the corridor is confined within the disrupted area, and the scheduling horizon.

Train information

For each train running in the disruption timetable, its train path is displayed with a specific colour. IC and international (Thalys and ICE) trains are displayed in black (4) and red (10) respectively. All other types, i.e. L, P and S trains, got the characteristics of local trains and are shown in blue (5). Blocking time stairways for all trains are included with the same colour coding. Elements (8) and (9) indicate respectively the start and end of the reservation time of the last block section for a train.

Measures

The time-distance diagrams also contain valuable information on the taken measures. Dots (7) indicate the release times, i.e. original arrival times of trains at either end of the displayed (part of a) corridor, and have the same colour coding as before. This can be used to identify delays: this specific train (7) is the next one entering the corridor in direction 0, indicated by (14). If the train switched tracks for the third AIP (Chapter 8), these dots have not been filled. An exception are the red dots followed by a red dotted train path without any blocking times (6). The latter group represents train cancellations.

Dots within the disrupted area display either stops (considered in Chapters 7 and 8) being performed in blue and black (11) or skipped stops in red (13). As international trains do not stop on the corridor between two larger stations, mixing up these two possibilities does not come into play. Dots representing stopping are located within the middle of the block section, and midway in-between the entry and departure times for that block section. Note that for performed stops, the train path does not represent the actual process of braking, dwelling and acceleration.

Appendix B

Computation Times

Chapters 6, 7 and 8 briefly touch upon the problems with computation times for several scenarios. The aim of this appendix is threefold. One, it provides an illustration of why computation times increase dramatically and the relation with the solver settings. Two, it provides arguments for the three computational settings employed, based on the context of disruption management. The focus on this thesis is on modelling the problems and analyzing the effects of several parameters, not on finding the optimal solution quickly for one specific problem. Hence, a third part presents ideas on how to reduce computation times in further research.

B.1 Computational complexity and its impact on solutions

Throughout this thesis, several variants of machine scheduling problems have been used. All of them have been proven to be NP-hard. Omitting cancellation of trains, the first case study in Chapter 5 comes down to minimizing total (weighted) delay. As due dates are set to the first possible departure times, the delay (or lateness) of the train will always be positive. Hence, this is equivalent to minimizing the (weighted) sum of completion times, while satisfying minimum release times and sequence-dependent set-up times. Nogueira et al. [41] report these problems to be NP-hard, and the number of possible solutions increases exponentially with problem size, i.e. the number of trains. Hence, computation times may increase significantly. Also single-machine scheduling problems with rejection, i.e. cancellation, and controllable process times, i.e. in case of stop-skipping, have been proven NP-hard by respectively Zhang et al. [61] and Chen et al. [10].

The high frequency timetable for the first AIP (Chapter 6) presented the first instances for which problems with computation times appeared. The scenarios for increasing cancellation penalty w^{cancel} , i.e. 3,600 (A8), 5,400 (A9) and 7,200 s (A10), were solved in CPLEX with a time limit of 900 s. The instance with $w^{cancel} = 3,600$ could not be solved to optimality within this limit, the second and third instance required 647.3 and 210.6 s respectively. Hence, when cancelling trains becomes less interesting, the problem is solved faster. Solving the problem without cancellation, i.e. $x_t = 0 \ \forall t \in T$, required only 14.3 s.

Figure B.1 presents CPLEX' output log for scenario A10. The "best bound" (BB) column presents the lower bound for relaxing integrality constraints, i.e. integer and binary variables are not necessarily restricted to integral values. The "best integer" (BI) column reports the objective function value for the best feasible solution found. To prove optimality, both have to be equal and the "gap" represents how far the BI is from BB in terms of objective function value: $\left(gap = \frac{BI-BB}{BI}\right)$.

Figure B.1 shows that the optimal solution is found within less than 73.45 s of computation time. However, the gap decreases at a slow rate and proving optimality of this solution requires more than 140 s of additional computation time. One could state that the solver is able to quickly find the optimal solution, but has difficulties to prove that it has actually found the optimal one.

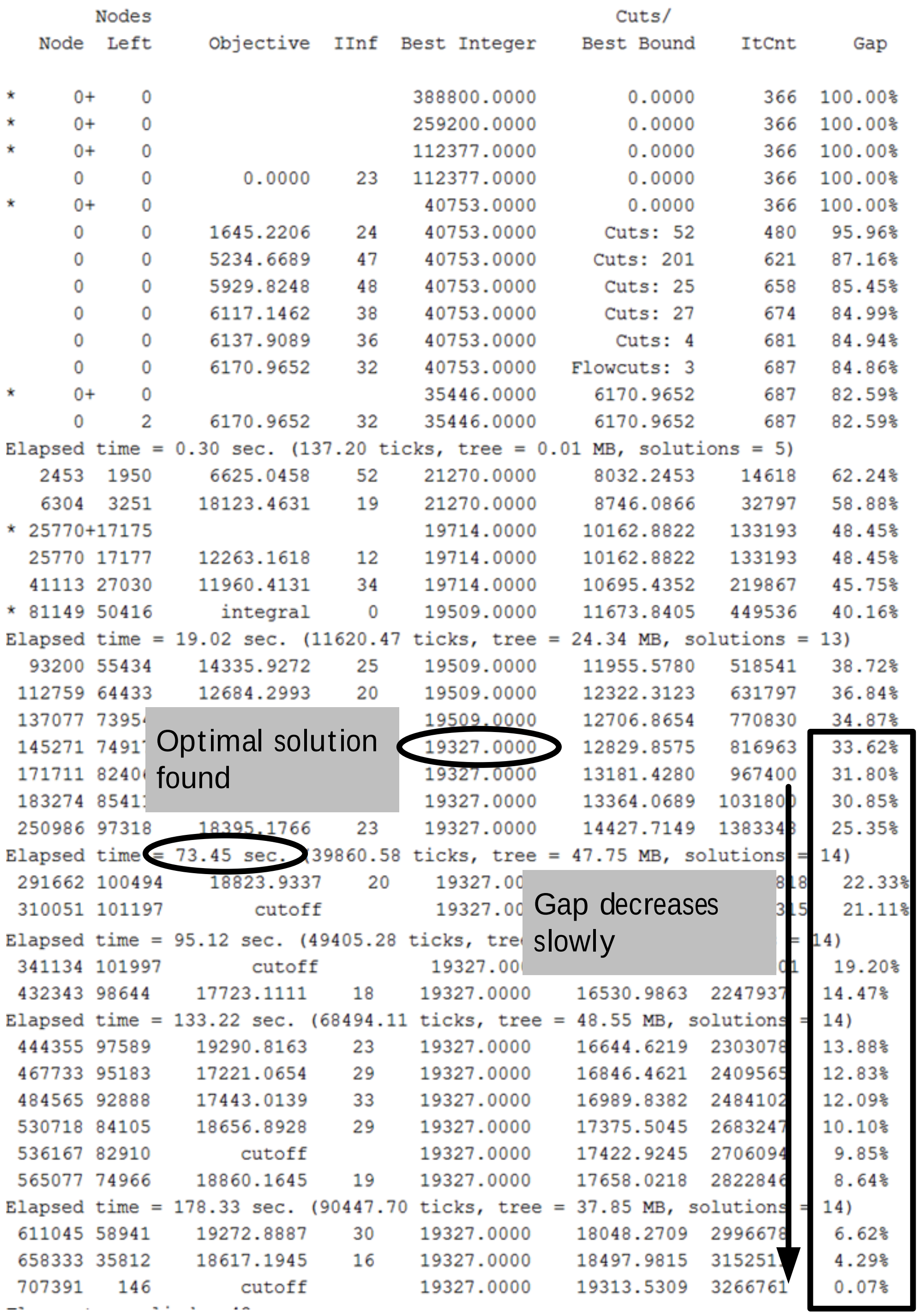
Although the objective function value for scenario A8 ($w^{cancel} = 3,600s$) is lower than those for higher w^{cancel}, it is not proven to be the optimal solution. Comparing results (see Figure 6.5, p.71) shows that inserting the cancelled train leads to strong differences in timetable structure. Optimal solutions for cancelling the train ($w^{cancel} = 3600$) and inserting it in the timetable ($w^{cancel} = 7200$, scenario A10) only differ in one cancellation variable, but in 44 out of 1,260 order variables. However, objective function values only differ by 100 or less than 1%.

B.2 Identification of possible causes and argumentation for computational settings

One of the possible causes lies within the branching strategy of the default solver settings. Basically[1], it comes down to the path illustrated in Figure B.2. Nodes are characterized by objective function values, denoted LB ("lower bound") if it is not a feasible solution, i.e. not all binary variables $\in \{0, 1\}$, or OF if it is a feasible one. The green (filled) and red (unfilled) dots represent two feasible solutions, the former being the optimal one.

Starting at the top node with all integrality constraints relaxed, the problem is solved. Next, one of the binary variables is fixed to 0 and 1 respectively to calculate the next level of nodes whose solutions have objective function values LB_1 and LB_2. As $LB_2 > LB_1$, branching continues on the left node. This depth-first search leads to the solution with OF_1. Optimality cannot be proven as the other node originating from the last branching operation has a lower bound $LB_3 < OF_1$. Consider LB_3 the lowest lower bound, the gap is determined as $gap = \frac{OF_1 - LB_3}{OF_1}$. As long as $gap > 0$, optimality could not be proven. Due to the large number of binary variables, searching the whole tree requires a long time. Following the strategy illustrated in Figure B.2, the optimal solution (green filled node) may not be found within short computation time limits. Also degeneracy of the problem, especially in Chapter 8 may come into play.

[1] This explanation makes abstraction of a number of improvements by the solver developers, but is used to present the key notions.




```
        Nodes                                         Cuts/
   Node  Left     Objective  IInf  Best Integer    Best Bound    ItCnt     Gap

*     0+    0                      388800.0000        0.0000      366  100.00%
*     0+    0                      259200.0000        0.0000      366  100.00%
*     0+    0                      112377.0000        0.0000      366  100.00%
      0     0        0.0000    23  112377.0000        0.0000      366  100.00%
*     0+    0                       40753.0000        0.0000      366  100.00%
      0     0     1645.2206    24   40753.0000      Cuts: 52      480   95.96%
      0     0     5234.6689    47   40753.0000     Cuts: 201      621   87.16%
      0     0     5929.8248    48   40753.0000      Cuts: 25      658   85.45%
      0     0     6117.1462    38   40753.0000      Cuts: 27      674   84.99%
      0     0     6137.9089    36   40753.0000       Cuts: 4      681   84.94%
      0     0     6170.9652    32   40753.0000   Flowcuts: 3      687   84.86%
*     0+    0                       35446.0000     6170.9652      687   82.59%
      0     2     6170.9652    32   35446.0000     6170.9652      687   82.59%
Elapsed time = 0.30 sec. (137.20 ticks, tree = 0.01 MB, solutions = 5)
   2453  1950     6625.0458    52   21270.0000     8032.2453    14618   62.24%
   6304  3251    18123.4631    19   21270.0000     8746.0866    32797   58.88%
* 25770+17175                       19714.0000    10162.8822   133193   48.45%
  25770 17177    12263.1618    12   19714.0000    10162.8822   133193   48.45%
  41113 27030    11960.4131    34   19714.0000    10695.4352   219867   45.75%
* 81149 50416      integral     0   19509.0000    11673.8405   449536   40.16%
Elapsed time = 19.02 sec. (11620.47 ticks, tree = 24.34 MB, solutions = 13)
  93200 55434    14335.9272    25   19509.0000    11955.5780   518541   38.72%
 112759 64433    12684.2993    20   19509.0000    12322.3123   631797   36.84%
 137077 7395                        19509.0000    12706.8654   770830   34.87%
 145271 7491                        19327.0000    12829.8575   816963   33.62%
 171711 8240                        19327.0000    13181.4280   967400   31.80%
 183274 8541                        19327.0000    13364.0689  103180    30.85%
 250986 97318    18395.1766    23   19327.0000    14427.7149  138334    25.35%
Elapsed time = 73.45 sec. (39860.58 ticks, tree = 47.75 MB, solutions = 14)
 291662 100494    18823.9337    20   19327.00                     818   22.33%
 310051 101197        cutoff        19327.00                     315   21.11%
Elapsed time = 95.12 sec. (49405.28 ticks, tre                    = 14)
 341134 101997        cutoff        19327.00                      01   19.20%
 432343 98644    17723.1111    18   19327.0000    16530.9863  2247937  14.47%
Elapsed time = 133.22 sec. (68494.11 ticks, tree = 48.55 MB, solutions = 14)
 444355 97589    19290.8163    23   19327.0000    16644.6219  2303078  13.88%
 467733 95183    17221.0654    29   19327.0000    16846.4621  2409565  12.83%
 484565 92888    17443.0139    33   19327.0000    16989.8382  2484102  12.09%
 530718 84105    18656.8928    29   19327.0000    17375.5045  2683247  10.10%
 536167 82910        cutoff         19327.0000    17422.9245  2706094   9.85%
 565077 74966    18860.1645    19   19327.0000    17658.0218  2822846   8.64%
Elapsed time = 178.33 sec. (90447.70 ticks, tree = 37.85 MB, solutions = 14)
 611045 58941    19272.8887    30   19327.0000    18048.2709  2996678   6.62%
 658333 35812    18617.1945    16   19327.0000    18497.9815  315251    4.29%
 707391   146        cutoff         19327.0000    19313.5309  3266761   0.07%
```

Figure B.1: CPLEX' output log when solving scenario A10 of Section 6.2 with a 900 s time limit.

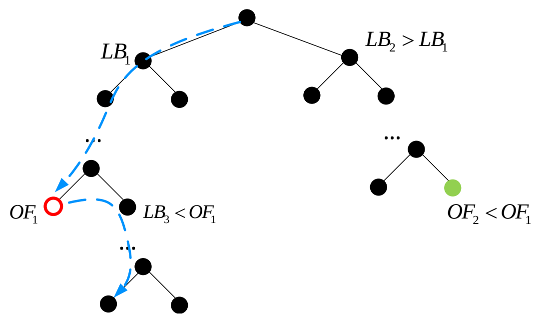


Figure B.2: Illustration of the branching strategy when applying the default settings in CPLEX. Dots represent nodes, LB and OF their associated objective functions values for infeasible and feasible solutions respectively. For the former group, integrality constraints are not all satisfied.

Table B.1 reports the objective function values of the solutions returned after 120 and 900 s computations for the scenarios which include buffer times (Section 6.7). Clearly, for scenarios E7, E8 and E10, better solutions are obtained, although the difference is only very minor. Therefore, one could accept these solutions already based on the following argumentation:

> Dispatchers are better supported by a good solution within short computation times, than an optimal one of which they do not know how long to wait for it.

When only a limited time is available, one could advocate it does not make sense to focus on proving optimality. Rather, it is better to focus the search on finding more feasible solutions and returning the best one found. In CPLEX, this can be done by setting the MIPEmphasis parameter to 4, meaning "Emphasize finding hidden feasible solutions" [23]. Throughout this thesis, it will be referred to as *MIPfocus*. As Table B.1 shows, it may generate (much) better solutions within 120

Table B.1: Objective function values of the solutions returned by three different computational settings for scenarios E6 to E10 on the first AIP (Section 6.7). The last two columns present the values relative to those obtained after 900 s.

#	Objective function values			Relative increases reference = 900 s	
	Default 120 s	Default 900 s	MIPfocus 120 s	120 s	MIPfocus
E6	19,327	19,327	19,327	0%	0%
E7	41,544	39,900	39,223	4%	-2%
E8	68,430	67,338	58,530	2%	-13%
E9	32,024	32,024	32,024	0%	0%
E10	38,597	38,333	38,333	1%	0%

s than default settings do within 900 s. This statement holds true for the major portion of all scenarios reported within this thesis.

To conclude, scenarios are run with three computational settings:

1. Default settings with time limit 120 s.
2. Default settings with time limit 900 s.
3. MIPfocus settings with time limit 120 s.

The first one represents the basic settings one would use without a further search for improvements, the second one is used as a reference to check the MIPfocus' performance[2]. Results are always reported for the latter, as these would be presented to dispatchers.

B.3 Improving the computational performance

Following the observations of the previous sections, several improvements to come up with better solutions faster have been considered. Recall that the focus of this thesis is on developing the models and analysing them, not on their computation times. However, several of them have been considered throughout this thesis research with the MIPfocus settings being the result of this.

One possibility would be to define an acceptable relative gap. When the solver reaches a point at which *gap* is smaller than this predefined value, computations are aborted and the current solution is returned. A few of the cited works in Chapter 4 do this, but the issue lies within defining the threshold. This would significantly depend on aspects among which penalty values as experiments showed.

Next, limiting the search space could be an option. For most scenarios, the order between trains in the same direction is fixed, which is a valid assumption. However, when relaxing the order between them, Constraints (5.20) may still fix some of these values. Further limitations on search space would require imposing restrictive maximum delay values such as in [57]. Reducing problem size could also be achieved by considering shorter disruption durations (as in Section 7.4.3), but this may hamper identification of certain interactions when analysing results. Care has to be taken when limiting search space, as it may also hamper the search for better solutions, as illustrated by the computation time increases when imposing service constraints (Section 5.5.5).

Finally, two heuristics have been presented in this thesis (Section 5.6), generating solutions within less than 1 s. Note that the (max,wait) heuristic requires tailoring of the parameters in order to generate solutions of acceptable quality. Hence, within an operational context, parameter selection is key to achieve good results. On the other hand, this heuristic did not manage to thoroughly balance between delaying trains and cancellation.

[2] For example, for scenarios A7 to A9 in Section 8.5.1 it seemed desirable to increase the MIPfocus time limit from 120 to 180 s.

Future research may look into the development of new heuristic approach based on the analyses presented throughout the thesis. Another pathway is to alter the branching strategy of the solver, which has already been done for the alternative graph approach [16]. Using constraint-specific M values as Nogueira et al. [41] describe, may improve performance as well, as this parameter may impact on computations.

Appendix C

Limiting Batch Size: Constraint Derivation

Optimally exploiting the remaining capacity of a segment D may lead to large batch sizes, not allowing trains to run in the direction opposite to the batch's direction for a long time. Limiting the number of trains running after each other in the same direction, i.e. batch size, becomes a plausible additional constraint in these situations. This appendix develops such constraints for the first AIP (Chapter 5), although they could be applied to other AIPs too.

Limiting the maximum batch size is accomplished by exploiting the order variables q_{ij}. Recall that $q_{ij} = 1$ if train j is scheduled after train i on the machine M^D. In case of cancellation, the train are moved to the start of the timetable, and q_{ij} can be fixed as follows:

$$q_{ij} \geq x_i + dev_i \qquad \forall i, j \in T, i > j \tag{C.1}$$

$$1 - q_{ij} \geq (x_j + dev_j) - (x_i + dev_i) \qquad \forall i, j \in T, i > j \tag{C.2}$$

Constraints (C.1) ensure that q_{ij} is set to 1 if train i is cancelled, regardless of train j being cancelled. By exploiting symmetry, i.e. $q_{ij} = 1 - q_{ji}$, Constraints (C.2) do the same for $i < j$. In case only train j is cancelled, this constraint reduces to $q_{ij} \leq 0$.

Using the q_{ij} values, one can count the number of trains operated before a specific one. Assuming train orders are fixed for trains in the same direction, n_i and n_j represent the ordinal numbers of trains i and j in their own direction, e.g. $n_i = 3$ if i is the third train. Constraints to limit the batch size in direction 0 in-between two subsequent trains i and j, i.e. $n_j = n_i + 1$, in direction 1 by a maximum batch size N^{max}_{batch} are formulated as:

$$\begin{aligned} \sum_{k \in T_1} q_{kj} - \sum_{k \in T_1} q_{ki} \leq \left(n_j - n_i - X_i^j\right) N^{max}_{batch} \\ + M(x_i + dev_i + x_j + dev_j) \qquad \forall i, j \in T_0, n_i < n_j \end{aligned} \tag{C.3}$$

The left-hand side terms in Constraints (C.3) count the number of trains that have in the other direction before trains j and i respectively. By enforcing $q_{kj} = q_{ki} = 0$ if train k is cancelled, it is accounted for in both terms. Hence, the difference between them equals the number of trains between both. However, cancellation of either train i or j relaxes Constraints (C.3).

Imagine the situation in Figure C.1a, which satisfies Constraints (C.3) with $N_{batch}^{max} = 2$: both between trains i and k, and k and j, only two trains run in the other direction. If train k is cancelled, both the constraints between i and k and k and j are relaxed and the batch size between j and the predecessor of k in direction 0, may become larger than N_{batch}^{max} (see Figure C.1b). Therefore, batch size constraints have to be added between any pair of trains in the same direction as follows:

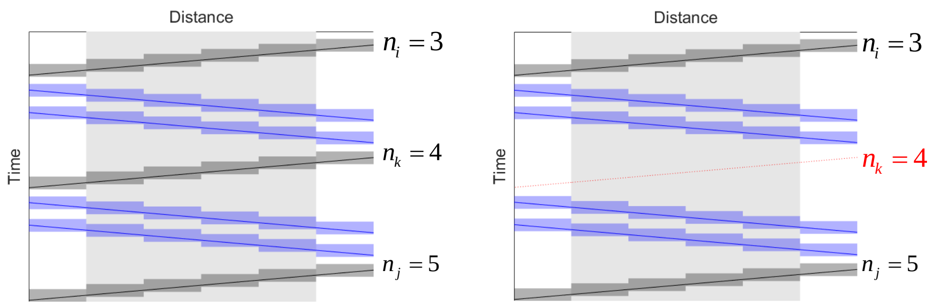
$$\begin{aligned}\sum_{k\in T_1} q_{kj} - \sum_{k\in T_1} q_{ki} \leq \left(n_j - n_i - X_i^j\right) N_{batch}^{max} \\ + M(x_i + dev_i + x_j + dev_j) \qquad \forall i,j \in T_0, n_i < n_j\end{aligned} \tag{C.4}$$

In Constraints (C.4), X_i^j is the cancellation term of all trains running after train i and before train j in the same direction, i.e. $X_i^j = \sum_{n_i<n_k<n_j \in T_0} (x_k + dev_k)$. For the example in Figure C.1, $X_i^j = x_k + dev_k$, meaning that if k is not cancelled or rerouted, the number of trains between i and j can be up to $2N_{batch}^{max}$ (Figure C.1a). On the other hand, cancellation of train k reduces the right-hand side to N_{batch}^{max} and the situation in Figure C.1b does no longer satisfy all constraints.

Limiting the batch size before the first train in a certain direction, which is not cancelled, is done in a similar way:

$$\begin{aligned}\sum_{k\in T_1} q_{kj} - \sum_{k\in T_1} (x_k + dev_k) \leq \left(n_j - \sum_{n_k<n_j} (x_k + dev_k)\right) N_{batch}^{max} \\ + M(x_j + dev_j) \qquad \forall j \in T_0\end{aligned} \tag{C.5}$$

The second term of the left-hand side in Constraints (C.5) only includes the cancelled and deviated trains. Constraints (C.4) and (C.5) can be generated for the other direction by reversing indices 0 and 1.



(a) (b)

Figure C.1: (a) Illustration how Constraints (C.3) successfully allow to limit the batch size ($N_{batch}^{max} = 2$) for a set of trains in absence of cancellation. (b) However, they fail in case the fourth train in direction 0 ($n_k = 4$) is cancelled.

Appendix D

Adjustments for the Second Archetypical Infrastructure Piece

This appendix elaborates on the stopping time, running time, headway time, and delay calculations for the second AIP, as an extension of Section 7.2. All of these aspects should be provided to the model as input, i.e. originating from a timetable. Infrabel either has these available, or uses its own calculation methods. In this thesis, the following assumptions are made:

1. A train skips all of its stops on the disrupted area or none of them, easing communication towards all actors and passengers. Moreover, it is in line with current practice at Infrabel: a train is either classified as IC or L [30].
2. Only trains t for which a stop at o_s was originally scheduled can either perform a stop or not. Hence, switching stopping patterns (as in [60]) between trains is not considered as a measure.
3. Performing a stop at $o_s \in O$ requires a minimum dwell time $t^{min}_{dwell,s}$. Minimizing total delay means that no train stop lasts longer than this value.

Sections D.1, D.2 and D.3 discuss the adjustments to running time, set-up time and delay calculation respectively. Section 7.2.1 summarizes the aspects which are most relevant to the model.

D.1 Adjusting process times to include stopping

To perform the commercial stop, trains have to brake and re-accelerate, resulting in time supplements on top of only running through the segment. Hence, dwell time becomes a second constituent for a train's process time. The exact location of the stop within the block section influences the speed profile. However, to keep the model as general as possible, two simplifications are made. One, trains enter and leave a block section at the same speed as they would do without stopping (see

Section 5.5.2): braking and acceleration are performed completely within the block section. In case of high train speeds and short block sections, this assumption may be violated. However, disruption speed is in general lower, decreasing the braking and acceleration distances. Two, the head of the train stops at a location within the block section, allowing it to accelerate to the exit speed. This simplification does not influence running times over the full segment D, which is merely a sum of the running times over all block sections. For example, if a stop is located at the end of a block section, the train's acceleration process would extend into the next one.

Figure D.1 sketches the speed profile of a train running from left to right, and performing a commercial stop at o_s within the block section b. Assume the bold signal is not present. The train enters the segment at the speed of the previous block section v_{b-1} and immediately decelerates to 40 km/h, which allows the train driver to stop-on-sight at the platform. After a certain dwell time, the train re-accelerates and leaves the block section at speed v_{b+1}, which may be higher, lower, or equal to v_{b-1}. Using this speed profile and constant acceleration and braking rates a_t and b_t, the running time over b of train t are determined as follows:

$$t_b = \frac{v_{b-1}}{b_t} \qquad x_b = \frac{v_{b-1}^2}{2b_t} \tag{D.1}$$

$$t_a = \frac{v_{b+1}}{a_t} \qquad x_a = \frac{v_{b+1}^2}{2a_t} \tag{D.2}$$

$$R_{tb}^{stopping} = 1.05 \left(\frac{l_b - x_b - x_a}{v_{sight}} + t_b + t_a + t_{dwell,s}^{min} \right) \tag{D.3}$$

Equations (D.1) and (D.2) calculate the time and distances required for deceleration and acceleration respectively. Note that deceleration is a two-step process with $v_{sight} = 40\ km/h$. Assuming all block sections have a speed limit of 80 km/h and $a_t = 0.6\ m/s^2$ and $b_t = 0.375\ m/s^2$, $x_b = 658\ m$ and $x_a = 412\ m$, justifying the assumption of both processes being conducted completely within the block section b. Using these values, Equation (D.3) calculates the running time required to perform the stop in b. The first term determines the travel time over the distance not needed for acceleration or deceleration, l_b representing the block section's length. As in Section 5.5.2, a 5% buffer is added to the technical minimum running time to absorb small deviations.

Differences exist for cases with stops in segment S_1. For these, an additional signal (bold in Figure D.1) is located right at the end of the platform, assumed to be midway the segment S_1. Braking and acceleration are performed within the first and second half of segment S_1 respectively. Equation (D.3) can be used with l_b being half of the original value and omitting t_a for the first half. For the second one, t_b and $t_{dwell,s}^{min}$ have to be removed from the first factor.

Trains either perform all of their stops or none, resulting in two possible process times on machine M^D for those trains with scheduled stops. Let binary decision variables σ_t represent the execution of all commercial stops by train t within the disrupted area. Stop-skipping reduces the running time over block section b from the

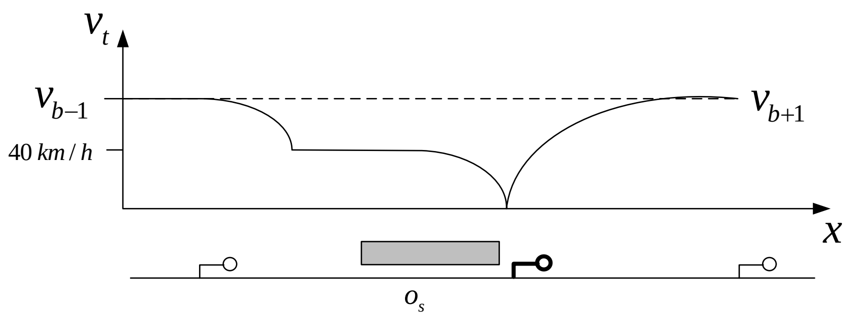


Figure D.1: Speed profile of a train performing a stop at o_s within block section b. The bold signal is only present in case o_s is located within segment S_1 or S_2. After entering the block section, the train slows down to 40 km/h, allowing it to stop-on-sight at the platform. After dwelling, it accelerates to the maximum speed in the next block section v_{b-1} if it does not exceed the speed restriction of block section b.

running time including a stop ($R_{tb}^{stopping}$) to the standard value R_{tb} as determined by the method in Section 5.5.2. However, a penalty $w_{t,b}^{stop-skip}$ has to be paid, equivalent to machine scheduling with discrete controllable process times (see Section 4.4.3). Let S represent the set of stopping operations (t, b) with train t performing a stop at stop o_s located in block section b. Process time constraints are adapted as follows:

$$C_t \geq t_t + p_t(1 - x_t - dev_t) + \sum_{b:(t,b) \in S} \left(\Delta r_{tb}^{stop} \right) \sigma_t \qquad \forall t \in T \tag{D.4}$$

$$\sigma_t \leq x_t + dev_t \qquad \forall t \in T \tag{D.5}$$

Constraints (7.1) express that performing the stops ($\sigma_t = 1$) increases the process time without stops (p_t) with a *stopping time supplement* $\Delta r_{tb}^{stop} = R_{tb}^{stopping} - R_{tb}$ per stop location. When stops are located in segments S_1 or S_2, incorporation of time required to decelerate, depends on whether the stop is located within the disrupted area (Figure 7.2, top), or not (Figure 7.2, bottom). Regardless of this aspect, the acceleration part always has to be included in the additional running time over the disrupted area.

Note that including the cancellation factor in the last term of Equation (D.4) would lead to non-linear constraints. Hence, Constraints (D.5) state that cancelling or rerouting a train, results in the stop being skipped.

D.2 Influence on set-up times

Performing a stop on one of the block sections fundamentally changes the blocking time stairways of the involved train. Each blocking time still consists of all components listed in Section 2.2.2, except for the block section right after the stop: a stopping train only reserves the next block section a time $t_{stop}^{approach}$ before its dwell ends.

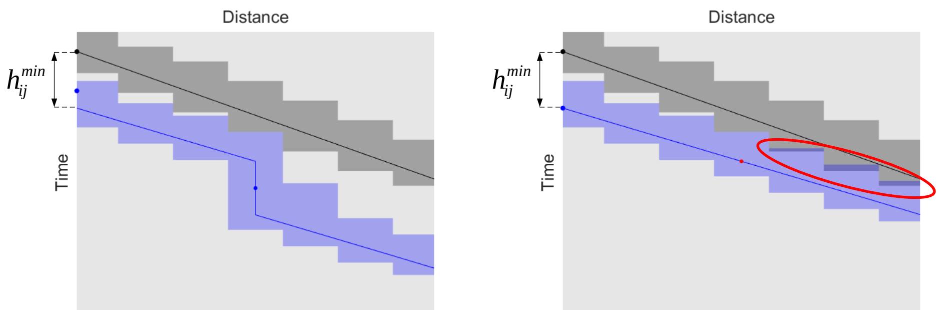
As a result, minimum headway and set-up times may depend on performing a stop or not. Assume these would be calculated in the same way as in Section 5.5.2, based on the blocking time stairways for stopping trains and resulting in minimum headway time h_{ij}. Conducting the stopping operation as planned, results in the time-distance diagram shown in Figure D.2a. However, if the second train skips its stop, it may catch up with its predecessor as shown in Figure D.2b causing a conflict (encircled in red). One can see that both set-up times and the critical block section may differ.

Assume only train i has a scheduled stop. Two levels for set-up times with its non-stopping successor j arise: $s_{ij}^{stopping}$ and $s_{ij}^{disrupted}$ for performing and skipping the stops respectively. To calculate these, Equation (5.1) can be used as before. Let $\Delta s_{ij} = s_{ij}^{stopping} - s_{ij}^{disrupted}$ represent the (negative) increase in set-up times. Constraints (5.7) and (5.8) are replaced with:

$$t_j \geq C_i + s_{ij}^{disrupted} + \Delta s_{ij}\ \sigma_i - M(1 - q_{ij} + x_i + x_j + dev_i + dev_j) \qquad \forall i, j \in T \tag{D.6}$$

$$t_i \geq C_j + s_{ij}^{disrupted} + \Delta s_{ij}\ \sigma_i - M(q_{ij} + x_i + x_j + dev_i + dev_j) \qquad \forall i, j \in T \tag{D.7}$$

Constraints (D.6) and (D.7) ensure that set-up times between trains are adjusted depending on whether train i stops or not. They can be employed for cases in which j is the stopping one by replacing σ_i with σ_j.



(a) (b)

Figure D.2: Time-distance diagrams for a stopping trains following a non-stopping one. (a) Calculating h_{ij}^{min} as before results in sufficient time-separation if the stop is not skipped. (b) However, skipping it may lead to conflicts further on the corridor.

Difficulties arise when two trains following each other have scheduled stops, resulting in four different possible set-up times depending on which of trains i and j perform their stops or not. These set-up times do not show an additive nature, as the critical block section may differ for each combination (as is also the case in Figure D.2). Moreover, they may be either higher or lower than the base values of both trains skipping their stops, i.e. $s_{ij}^{disrupted}$. Modelling all possible combinations would result in multiple and complex pairs for Constraints (D.6) and (D.7), of which most will be abundant.

Hence, a simplification is made. Let $s_{ij}^{(\sigma_i,\sigma_j)}$ represent the set-up time after train i with operation pattern σ_i, i.e. stopping (1) or skipping (0), before train j can be operated. Constraints (D.6) can be used with $s_{ij}^{disrupted} = \max\left(s_{ij}^{(0,0)}, s_{ij}^{(0,1)}\right)$ and $s_{ij}^{stopping} = \max\left(s_{ij}^{(1,0)}, s_{ij}^{(1,1)}\right)$, ensuring sufficient train separation regardless of the exact operation of train j. Although this may result in abundant train separation, the resulting buffer times are expected to be rather small, and it is better to have these buffer times than risking conflicts between trains.

Finally, closures of tracks in which only part of segment S_1 belongs to the disrupted area (Figure 7.2, bottom), with trains dwelling outside of the disrupted area, have to be considered. Trains in opposite direction cannot have a conflict in the first half of segment S_1, as they operate on different tracks. The method explained by Figure 5.8 has to be applied while disregarding this block section. However, for trains in the same direction, conflicts may arise in this block section, resulting in the need to incorporate it in the set-up time calculation.

D.3 Delay calculation

Question remains how the delay D_t of a stopping train has to be calculated. For a non-stopping train, d_t in Constraints (5.5) referred to the adapted departure time, taking into account disruption speed v_D. For a train with a stop, this could be the first possible departure time either with or without stop-skipping.

Assume D_t is calculated relative to the earliest departure without a stop, meaning that skipping the stop leads to a decreased delay, at the expense of the stop-skipping penalty. On the other hand, performing it increases the process time and results in a delay, whereas passengers may still arrive on time. The alternative is to let d_t represent the earliest departure including all stops, thereby avoiding the latter effect. Yet, by skipping a stop, train t can still arrive early ($C_t \leq d_t$). Constraints (5.6) ensure such situations do not result in $D_t < 0$, which would promote stop-skipping.

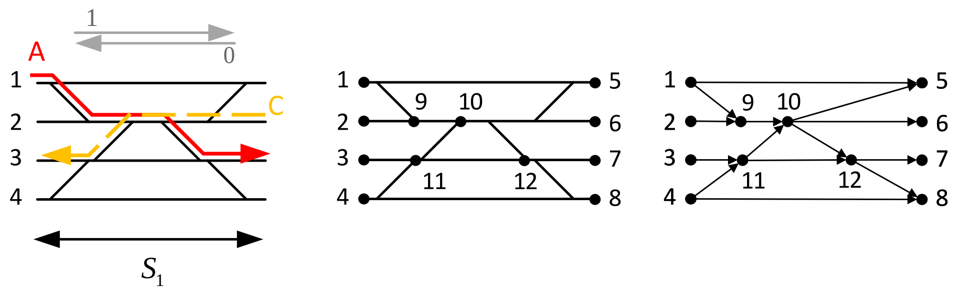
Appendix E

Conflict Identification on the Third Archetypical Infrastructure Piece

This appendix elaborates on a methodology to determine pairs of track assignments with conflicting paths on segments S_1 and S_2, so-called *overlapping track assignments*, for the third AIP in Section 8.6. Recall that exact configurations for these segments vary from case to case as opposed to their standardized configurations for the first and second AIP. Figure E.1a presents one possible configuration for segment S_1 on a four-track line[1], which is used to illustrate the methodology.

Consider the following problem: trains t have an originally assigned track k_t^{orig} on which they have to run before and after the segment D. However, during the

[1]This example exists near the junction "Ruisbroek" on line 96, south of Brussels, as can be seen on a technical map of the Belgian network [26].



(a) (b) (c)

Figure E.1: Methodology to identify overlapping track assignments for a particular example of a segment S_1 configuration. (a) Considered configuration of segment S_1, with an overlapping track assignment for trains A and C. (b) Identification of nodes with merging incoming tracks. (c) Unidirectional graph representation of the infrastructure and its links between nodes.

disruption, they may get assigned to a different track (k_t). Let trains A, B, C and D denote trains originally scheduled on tracks 1, 2, 3 and 4 respectively. For example, train A in Figure E.1a (red full line) runs in direction 1 and is originally scheduled on track 1, and gets assigned to track 3. Hence, it uses the segment S_1 to move from track 1 to track 3 and segment S_2 vice versa. On the other hand, train C (orange dashed line) has been assigned to track 2, and has to revert to track 3 after exiting from segment D. If both are scheduled too close in time, a conflict arises due to overlapping paths.

In a first step, nodes on the infrastructure of Figure E.1a are identified for trains running from left to right[2]. Nodes 1 to 4 represent the entrances for trains originally scheduled on tracks 1 to 4, i.e. trains A to D, and nodes 5 to 8 represent the exits, i.e. the start of segment D. Locations were tracks merge represent points where new conflicts may arise. These are marked for trains running from left to right, resulting in nodes 9 to 12 in Figure E.1b. In theory, nodes 5 and 8 also represent such a merging of tracks.

Next, a unidirectional graph is constructed by adding arcs between nodes directly connected by the physical infrastructure in Figure E.1c. The path of train A in Figure E.1a can be characterized by the nodes it passes: 1 - 9 - 10 - 11 - 12 - 7. In a bidirectional graph, also the path 1 - 9 - 10 - 12 - 7 would be a possibility, which it is clearly not in reality (Figure E.1a). Similarly, the path of train C can be determined as 3 - 11 - 10 - 6 (or reversed). Both paths of A and C contain the node 12, meaning they (partially) overlap. Hence, assigning trains A and C to tracks 3 and 2 respectively, i.e. $k_A = 3$ and $k_C = 2$, represents a conflicting track assignment on segment S_1.

Finally, all possible overlapping track assignments are identified by solving a shortest path problem for any pair of entry, i.e. nodes 1 to 4, and exit nodes, i.e. 5 to 8. For this specific configuration, Tables E.1a to E.1f present all overlapping track assignments for a pair of trains, characterized by their original track. Rows represent the track assignment for the first train, columns the one for the second train. Conflicts are indicated by a value 1. For example, the shaded entry in Table E.1b highlights a conflict when trains A and C are assigned to tracks 3 and 2 respectively, as in Figure E.1. Constraints (8.7) and (8.8) for sequence-dependent set-up times of trains assigned to the same track, already account for the diagonal entries. Pairs of trains originally assigned to the same track, e.g. trains D and D, will have overlapping paths regardless of their assigned tracks.

[2] Note that trains running from right to left can use exactly the same paths due to symmetry as all tracks can be used in both directions. For this rather simple configuration, this method is satisfactory as trains enter and exit at different ends, e.g. none run from node 5 to 6. For more complex infrastructures, one should use the colon graph method described by Radtke [47].

Table E.1: Identification of overlapping track assignments for the configuration presented in Figure E.1. Letters represent trains and are representative for the originally scheduled track k_t^{orig}, i.e. $k_A^{orig} = 1$, $k_B^{orig} = 2$, $k_C^{orig} = 3$, and $k_D^{orig} = 4$. Each cell indicates whether assigning the trains to the numbered tracks, results in overlapping track assignments (1) or not (0). For example, the shaded entry in (b) denotes that assigning trains originally scheduled on tracks 1 and 3, to respectively tracks 3 and 2, results in overlapping paths.

(a) Trains A & B

		Train B			
		1	2	3	4
Train A	1	-	0	0	0
	2	1	-	1	1
	3	1	1	-	1
	4	1	1	1	-

(b) Trains A & C

		Train C			
		1	2	3	4
Train A	1	-	0	0	0
	2	1	-	0	0
	3	1	1	-	1
	4	1	1	1	-

(c) Trains A & D

		Train D			
		1	2	3	4
Train A	1	-	0	0	0
	2	1	-	0	0
	3	1	1	-	0
	4	1	1	1	-

(d) Trains B & C

		Train C			
		1	2	3	4
Train B	1	-	1	0	0
	2	1	-	0	0
	3	1	1	-	1
	4	1	1	1	-

(e) Trains B & D

		Train D			
		1	2	3	4
Train B	1	-	1	0	0
	2	1	-	0	0
	3	1	1	-	0
	4	1	1	1	-

(f) Trains C & D

		Train D			
		1	2	3	4
Train C	1	-	1	1	0
	2	1	-	1	0
	3	1	1	-	0
	4	1	1	1	-

Appendix F

Timetable Data Case Study Tienen - Landen

This appendix provides an overview of the timetable data for the practical case study Tienen-Landen of the first AIP (Chapter 5). Tables G.1a and G.1b present train type and entry time for all trains running between 5:54 and 8:54 in directions 1 and 0 respectively. Data is based on public information provided in the folder of line 36 [38]. Trains running in directions 1 and 0 enter the corridor at their departure times from stations Tienen and Landen respectively. The stations are located at the extreme ends of the corridor in Figure 5.5. Stopping locations are neglected for this case study.

Time-distance diagrams in Figures F.1a and F.1b visualize the original timetable for trains running in directions 1 and 0 respectively. The applied speed equals the most common disruption speed of 80 km/h. Superimposing them in Figure F.2 results in a large number of conflicts and crossing trains. Hence, there is a need for rescheduling during the disruption. Here, the time-distance diagrams are shown for the complete corridor. To improve readability, those in Chapter 5 only show the disrupted area, i.e. segments S_1, D and S_2.

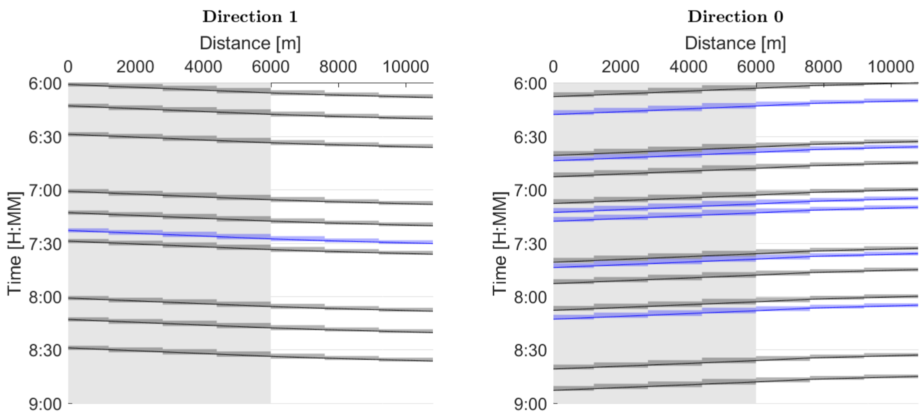
Table F.1: Timetable data for the practical case study of the first AIP (Tienen-Landen). Information concerns the train types and entry times for those in direction (a) 1, and (b) 0 respectively.

(a) Direction 1

Train type	Entry time (H:MM)
IC	5:55
IC	6:07
IC	6:23
IC	6:55
IC	7:07
P	7:17
IC	7:23
IC	7:55
IC	8:07
IC	8:23

(b) Direction 0

Train type	Entry time (H:MM)
IC	5:54
P	6:04
IC	6:27
P	6:30
IC	6:39
IC	6:54
P	6:59
P	7:04
IC	7:27
P	7:30
IC	7:39
IC	7:54
P	7:59
IC	8:27
IC	8:39



(a) (b)

Figure F.1: Time-distance diagrams for the Tienen-Landen case, including the disruption speed restriction, for trains running in directions (a) 1 and (b) 0. The grey area indicates the disrupted area.

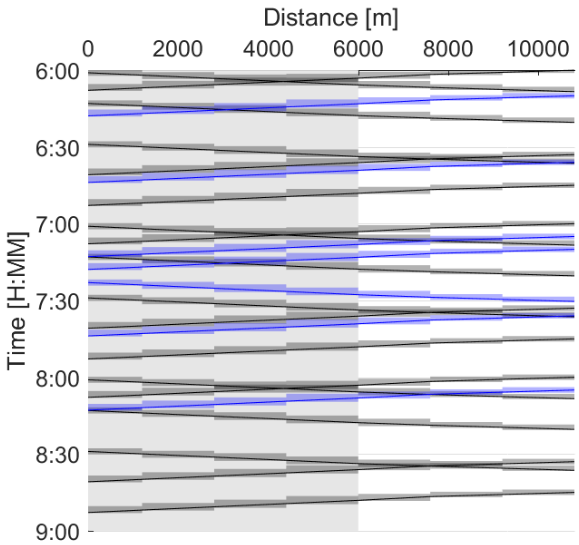


Figure F.2: Time-distance diagrams for the Tienen-Landen case study, including the disruption speed restriction, for trains running in both directions. The grey area indicates the disrupted area.

Appendix G

Timetable Data Case Study Oostkamp - Aalter

This appendix provides an overview of the timetable data for the practical case study Oostkamp-Aalter of the second AIP (Chapter 7). Tables G.1a and G.1b present train type and entry time for all trains running between 6:00 and 9:00 in directions 1 and 0 respectively. The last columns indicate whether the train stops at respectively Oostkamp (within segment A), Beernem (within segment S_1) and Maria-Aalter (within segment D). Section 7.1.1 mentions how these were derived from the information in the simulation tool mentioned by Van Thielen et al. [56].

This original timetable is visualized in the time-distance diagrams shown in Figures G.1a and G.1b for trains running in directions 1 and 0 respectively. The applied speed equals the most common disruption speed of 80 km/h. Figure G.2 superimposes them on each other, resulting in a large number of conflicts and crossing trains. Hence, there is a need for rescheduling during the disruption. Here, the time-distance diagrams are shown for the complete corridor. To improve readability, those in Chapter 7 only show the more interesting segments S_1, D and S_2.

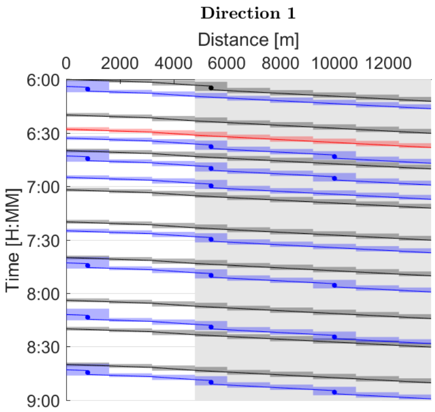
Table G.1: Timetable data the practical case study (Oostkamp-Aalter) of the second AIP. Information concerns the train types, entry times and whether stops are performed at Oostkamp (O), Beernem (B) and Maria-Aalter (MA) (x) for those in direction (a) 1, and (b) 0 respectively.

(a) Direction 1

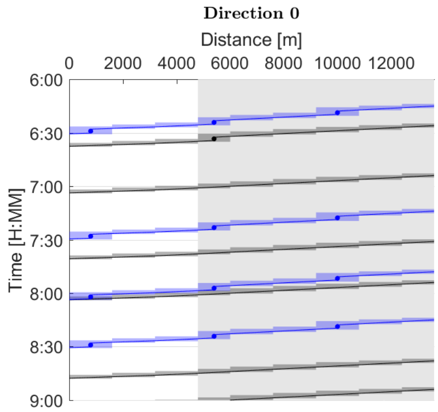
Train type	Entry time (H:MM)	Stops at		
		O	B	MA
IC	6:01		x	
P	6:05	x		
IC	6:21			
THA	6:29			
P	6:34		x	x
IC	6:41			
L	6:44	x	x	x
P	6:56		x	
IC	7:03			
IC	7:21			
P	7:26		x	
IC	7:41			
L	7:44	x	x	x
IC	8:05			
P	8:13	x	x	x
IC	8:21			
IC	8:41			
L	8:44	x	x	x

(b) Direction 0

Train type	Entry time (H:MM)	Stops at		
		O	B	MA
L	6:16	x	x	x
IC	6:27		x	
IC	6:55			
L	7:15	x	x	x
IC	7:32			
P	7:49	x	x	x
IC	7:55			
L	8:16	x	x	x
IC	8:39			
IC	8:55			



(a)



(b)

Figure G.1: Time-distance diagrams for the Oostkamp-Aalter case study, including the disruption speed restriction, for trains running in directions (a) 1 and (b) 0. The grey area indicates the disrupted area in case of closure of the track in direction 1.

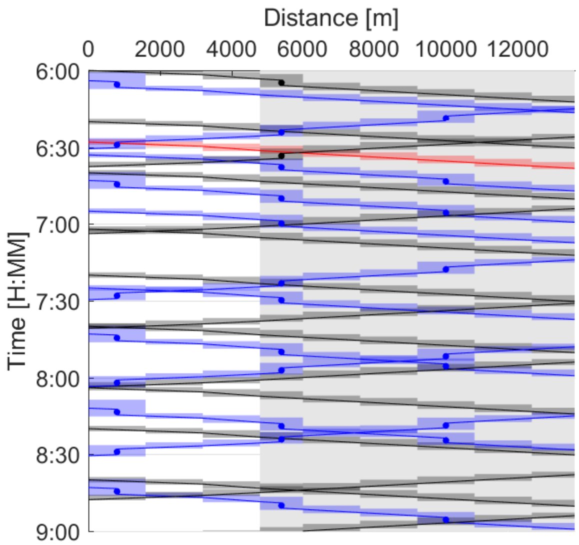


Figure G.2: Time-distance diagrams for the Oostkamp-Aalter case study, including the disruption speed restriction, for trains running in both directions. The grey area indicates the disrupted area in close of closure of the track in direction 1.

Appendix H

Timetable Data Case Study Schaarbeek - Diegem

This appendix provides an overview of the timetable data for the practical case study Schaarbeek-Diegem of the third AIP (Chapter 8). Tables H.1a and H.1b present train type, entry time and assigned track for all trains running between 6:29 and 9:29 in directions 1 and 0 respectively. The last column indicates whether the train stops at Haren-Zuid or not. Section 8.1.1 describes how these have been derived from public information in the folder of line 36 [38].

This original timetable is visualized in the time-distance diagrams shown in Figures H.1 and H.2 for tracks in directions 1 and 0 respectively. The applied speed equals the most common disruption speed of 80 km/h.

Table H.1: Timetable data for the practical case study (Schaarbeek-Diegem) of the third AIP. Information concerns the train types, entry times, assigned tracks, and whether a stop is performed at Haren-Zuid (x) for those in direction (a) 1, and (b) 0 respectively.

(a) Direction 1

Train type	Entry time (H:MM)	Track	Stops at Haren-Zuid
IC	6:29	1	
ICE	6:37	2	
IC	6:41	1	
S	6:45	1	x
IC	6:53	2	
IC	7:06	1	
IC	7:11	2	
S	7:15	1	x
IC	7:26	2	
IC	7:29	1	
IC	7:41	1	
S	7:45	1	x
IC	7:53	2	
IC	8:06	1	
IC	8:11	2	
S	8:15	1	x
IC	8:26	2	
IC	8:29	1	
IC	8:41	1	
S	8:45	1	x
IC	8:53	2	
IC	9:06	1	
IC	9:11	2	
S	9:15	1	x
IC	9:26	2	

(b) Direction 0

Train type	Entry time (H:MM)	Track	Stops at Haren-Zuid
P	6:36	3	
S	6:39	4	x
IC	6:43	3	
IC	6:51	4	
IC	7:01	4	
P	7:06	4	
S	7:09	4	x
IC	7:09	3	
P	7:21	4	
IC	7:26	3	
IC	7:27	4	
P	7:34	4	
P	7:37	3	
S	7:39	4	x
IC	7:44	3	
IC	7:51	4	
P	7:54	3	
IC	8:02	4	
P	8:06	3	
S	8:09	4	x
IC	8:13	4	
THA	8:22	3	
IC	8:26	3	
IC	8:27	4	
P	8:33	4	
S	8:39	4	x
IC	8:44	3	
IC	8:51	4	
IC	9:02	4	
S	9:09	4	x
IC	9:09	3	
ICE	9:16	3	
IC	9:28	3	

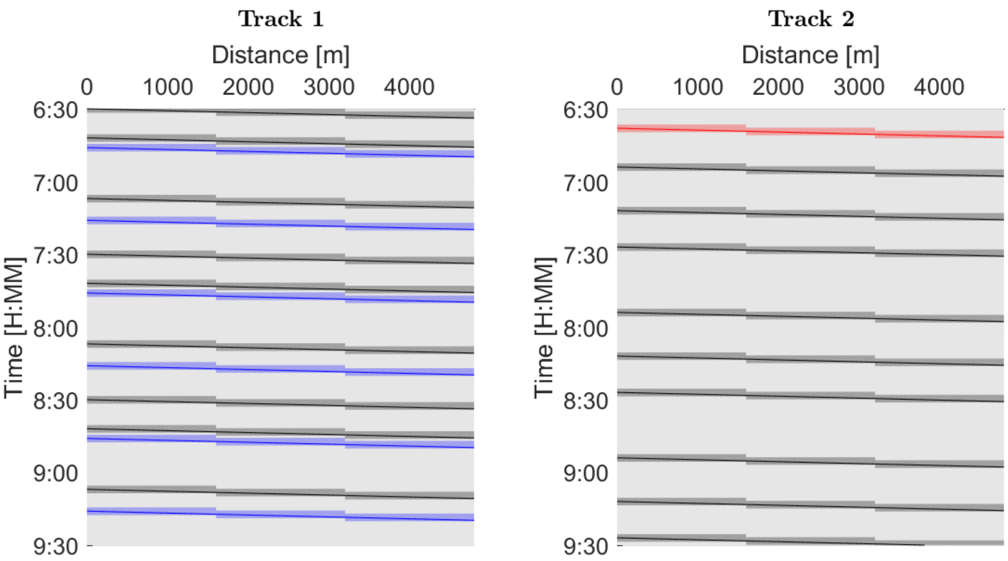


Figure H.1: Time-distance diagrams for the Schaarbeek-Diegem case, including the disruption speed restriction, for trains running in direction 1 on tracks 1 (left) and 2 (right).

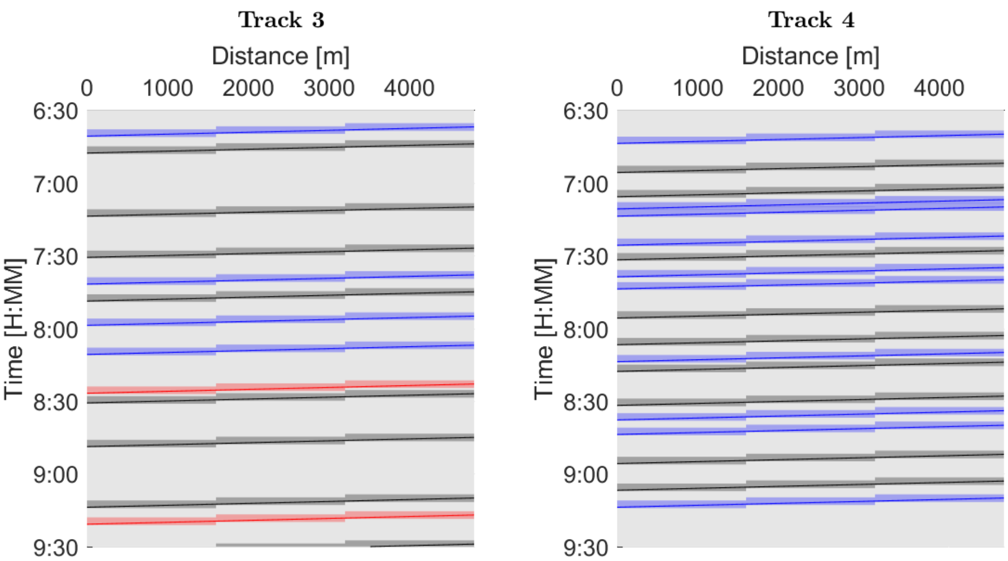


Figure H.2: Time-distance diagrams for the Schaarbeek-Diegem case, including the disruption speed restriction, for trains running in direction 0 on tracks 3 (left) and 4 (right).

Bibliography

[1] M. Abril, F. Barber, L. Ingolotti, M. A. Salido, P. Tormos, and A. Lova. "An assessment of railway capacity". In: *Transportation Research Part E: Logistics and Transportation Review* 44.5 (2008), pp. 774–806.

[2] K. R. Baker, G. D. Scudder, D. Trietsch, and S. T. Webster. *Elements of Sequencing and Scheduling*. Michigan, USA: University of Michigan, 1995, p. 242.

[3] Y. Bartal, S. Leonardi, A. Marchetti-Spaccamela, J. Sgall, and L. Stougie. "Multiprocessor scheduling with rejection". In: *SIAM Journal on Discrete Mathematics* 13.1 (2000), pp. 64–78.

[4] J. E. Beasley, M. Krishnamoorthy, Y. M. Sharaiha, and D. Abramson. "Scheduling aircraft landings – the static case". In: *Transportation science* 34.2 (2000), pp. 180–197.

[5] J. A. Bennell, M. Mesgarpour, and C. N. Potts. "Airport runway scheduling". In: *A Quarterly Journal of Operations Research, 4OR* 9.2 (2011), pp. 115–138.

[6] N. Besinovic, V. Cacchiani, T. Dollevoet, R. M. P. Goverde, D. Huisman, M. P. Kidd, L. G. Kroon, E. Quaglietta, J. Rodriguez, and P. Toth. "Integrated Decision Support Tools for Disruption Management". In: *Proceedings of the 6th International Conference on Railway Operations Modelling and Analysis (IAROR): RailTokyo2015, March 23-26*. Narashino, Japan, 2015.

[7] B. Bošković, M. Ivić, and A. Marković. "Organizing transportation on a double-track line under conditions of major overhaul on one track". In: *Yugoslav Journal of Operations Research* 16.1 (2006), pp. 97–105.

[8] O. Brünger and E. Dahlhaus. "Running Time Estimation". In: *Railway Timetabling & Operations: Analysis - Modelling - Optimisation - Simulation - Performance Evaluation*. Ed. by I. A. Hansen and J. Pachl. Hamburg, Germany: Eurailpress, 2014, pp. 65–89.

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

[10] Z.-L. Chen, Q. Lu, and G. Tang. “Single machine scheduling with discretely controllable processing times”. In: *Operations Research Letters* 21.2 (1997), pp. 69–76.

[11] F. Chu and A. Oetting. “Modeling capacity consumption considering disruption program characteristics and the transition phase to steady operations during disruptions”. In: *Journal of Rail Transport Planning & Management* 3.3 (2013), pp. 54–67.

[12] F. Corman, A. D’Ariano, and I. A. Hansen. “Disruption handling in large railway networks”. In: *WIT Transactions on The Built Environment* 114 (2010), pp. 629–640.

[13] F. Corman, A. D’Ariano, I. A. Hansen, D. Pacciarelli, and M. Pranzo. “Dispatching trains during seriously disrupted traffic situations”. In: *2011 IEEE International Conference on Networking, Sensing and Control (ICNSC), April 11-13*. Delft, the Netherlands, 2011, pp. 323–328.

[14] F. Corman, A. D’Ariano, D. Pacciarelli, and M. Pranzo. “A tabu search algorithm for rerouting trains during rail operations”. In: *Transportation Research Part B: Methodological* 44.1 (2010), pp. 175–192.

[15] A. D’Ariano, P. D’Urgolo, D. Pacciarelli, and M. Pranzo. “Optimal sequencing of aircrafts take-off and landing at a busy airport”. In: *2010 13th International IEEE Conference on Intelligent Transportation Systems (ITSC), September 19-22*. Madeira Island, Portugal, 2010, pp. 1569–1574.

[16] A. D’Ariano, D. Pacciarelli, and M. Pranzo. “A branch and bound algorithm for scheduling trains in a railway network”. In: *European Journal of Operational Research* 183.2 (2007), pp. 643–657.

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

[18] W. Eggermont. *An Introduction to Belgian Signalling - SNCB, Student Manual, Version 2.0*. Brussels, Belgium: NMBS, n.d.

[19] N. Ghaemi and R. M. P. Goverde. “Review of railway disruption management practice and literature”. In: *Proceedings of the 6th International Conference on Railway Operations Modelling and Analysis (IAROR): RailTokyo2015, March 23-26*. Narashino, Japan, 2015.

[20] M. van Hagen. “Waiting experience at train stations”. PhD thesis. Twente, the Netherlands: Universiteit Twente, 2011.

[21] I. A. Hansen and J. Pachl. *Railway Timetabling & Operations: Analysis - Modelling - Optimisation - Simulation - Performance Evaluation*. Hamburg, Germany: Eurailpress, 2014, p. 330.

[22] IBM. *IBM Knowledge Center - IBM ILOG CPLEX Optimization Studio V12.6.3 documentation*. [Last accessed 25/12/2016]. 2015. URL: http://www.ibm.com/support/knowledgecenter/SSSA5P_12.6.2/ilog.odms.studio.help/Optimization_Studio/topics/COS_home.html.

[23] IBM. *IBM Knowledge Center - MIP emphasis switch*. [Last accessed 10/10/2016]. 2016. URL: http://www.ibm.com/support/knowledgecenter/SS9UKU_12.4.0/com.ibm.cplex.zos.help/Parameters/topics/MIPEmphasis.html.

[24] Infrabel. *Punctuality figures*. [Last accessed 17/12/2016]. 2016. URL: https://www.infrabel.be/en/about/punctuality-figures.

[25] Infrabel. *Safety and Signalling Plans L36, zone Leuven, block 9 (1-FLV-002/99, pp. 14–17)*. Map. 2012.

[26] Infrabel. *The Network Statement - C.4 Technical Map of the Network*. [Last accessed 24/12/2016]. Jan. 2016. URL: http://www.infrabel.be/en/professionals/rail-operators/network-statement.

[27] J. Jespersen-Groth, D. Potthoff, J. Clausen, D. Huisman, L. Kroon, G. Maróti, and M. N. Nielsen. "Disruption Management in Passenger Railway Transportation". In: *Robust and Online Large-Scale Optimization: Models and Techniques for Transportation Systems*. Ed. by R. K. Ahuja, R. H. Möhring, and C. D. Zaroliagis. Berlin, Germany: Springer, 2009, pp. 399–421.

[28] K. Kerckaert. "Vision and strategic considerations of the current and future transportation plans of SNCB". In: *Presented during the 2nd Railway Operations Research Seminar "Put Passengers First", May 3*. Leuven, Belgium, 2016.

[29] L. Kroon and D. Huisman. *Algorithmic Support for Disruption Management at Netherlands Railways*. Econometric Institute Research Papers EI 2011-06. Rotterdam, the Netherlands: Erasmus University Rotterdam, Erasmus School of Economics (ESE), Econometric Institute, 2011. URL: http://EconPapers.repec.org/RePEc:ems:eureir:22456.

[30] M. Lambrechts. *Visit to Infrabel's local signal box in Leuven*. Visit. Leuven, Belgium, Oct. 2016.

[31] J. Löfberg. "YALMIP : A Toolbox for Modeling and Optimization in MATLAB". In: *Proceedings of the CACSD Conference, September 2-4*. Taipei, Taiwan, 2004.

[32] I. Louwerse and D. Huisman. "Adjusting a railway timetable in case of partial or complete blockades". In: *European Journal of Operational Research* 235.3 (2014), pp. 583–593.

[33] A. S. Manne. "On the job-shop scheduling problem". In: *Operations Research* 8.2 (1960), pp. 219–223.

[34] A. Mascis and D. Pacciarelli. "Job-shop scheduling with blocking and no-wait constraints". In: *European Journal of Operational Research* 143.3 (2002), pp. 498–517.

[35] *MATLAB version 9.0.0.341360 (R2016a)*. The Mathworks, Inc. Natick, Massachusetts, 2016.

[36] T. Nakamura, C. Hirai, and Y. Nishioka. “A practical train rescheduling algorithm using three predetermined factors”. In: *Proceedings of the 4th International Seminar on Railway Operations Modelling and Analysis (IAROR): RailRome2011, February 16-18*. Rome, Italy, 2011.

[37] S. Narayanaswami and N. Rangaraj. “Modelling disruptions and resolving conflicts optimally in a railway schedule”. In: *Computers & Industrial Engineering* 64.1 (2013), pp. 469–481.

[38] NMBS. *Lijnfolders - Lijn 36: Brussel-Zuid - Luik-Guillemins*. [Last accessed 05/11/2016]. 2016. URL: https://www.belgianrail.be/nl/klantendienst/infodiensten-reistools/lijnfolders.aspx.

[39] NMBS. *NMBS Visie en activiteiten 2014*. Report. Brussels, Belgium, 2015.

[40] NMBS. *Reisplanner*. [Last accessed 20/10/2016]. 2016. URL: http://www.belgianrail.be/jp/nmbs-routeplanner/query.exe/nn.

[41] T. H. Nogueira, C. R. V. de Carvalho, and M. G. Ravetti. “Analysis of mixed integer programming formulations for single machine scheduling problems with sequence dependent setup times and release dates”. In: *Optimization Online* (2014).

[42] A. Oetting and F. Chu. “Disruption Programs in Passenger Rail Transport - Ensuring Steady Operations During Disruptions”. In: *13th World Conference on Transport Research (WCTR), July 15-18*. Rio de Janeiro, Brazil, 2013.

[43] J. Pachl. “Timetable Design Principles”. In: *Railway Timetabling & Operations: Analysis - Modelling - Optimisation - Simulation - Performance Evaluation*. Ed. by I. A. Hansen and J. Pachl. Hamburg, Germany: Eurailpress, 2014, pp. 13–46.

[44] W. Passchyn. “Scheduling Locks on Inland Waterways”. PhD thesis. Leuven, Belgium: KU Leuven, 2016.

[45] P. Pellegrini, G. Marlière, and J. Rodriguez. “Optimal train routing and scheduling for managing traffic perturbations in complex junctions”. In: *Transportation Research Part B: Methodological* 59 (2014), pp. 58–80.

[46] E. Quaglietta, P. Pellegrini, R. M. P. Goverde, T. Albrecht, B. Jaekel, G. Marlière, J. Rodriguez, T. Dollevoet, B. Ambrogio, D. Carcasole, M. Giaroli, and G. Nicholson. “The ON-TIME real-time railway traffic management framework: A proof-of-concept using a scalable standardised data communication architecture”. In: *Transportation Research Part C: Emerging Technologies* 63 (2016), pp. 23–50.

[47] A. Radtke. “Infrastructure Modelling”. In: *Railway Timetabling & Operations: Analysis - Modelling - Optimisation - Simulation - Performance Evaluation*. Ed. by I. A. Hansen and J. Pachl. Hamburg, Germany: Eurailpress, 2014, pp. 47–63.

[48] RailNetEurope. *Network Statement Glossary*. [Last accessed 06/11/2016]. 2016. URL: http://www.rne.eu/ns_glossary.

[49] P. L. Rocha, M. G. Ravetti, G. R. Mateus, and P. M. Pardalos. "Exact algorithms for a scheduling problem with unrelated parallel machines and sequence and machine-dependent setup times". In: *Computers & Operations Research* 35.4 (2008), pp. 1250–1264.

[50] J. Rodriguez, P. Pellegrini, G. Marlière, S. Hu, and S. S. Richard. "Improvement of real-time traffic management by using optimization tools". In: *Procedia - Social and Behavioral Sciences* 160 (2014), pp. 465–473.

[51] Y. Segers. "Analyseren en beperken van de impact van tijdelijke snelheidsbeperkingen op het spoorwegnetwerk". MSc. Thesis. KU Leuven, 2015.

[52] P. Sels, T. Dewilde, D. Cattrysse, and P. Vansteenwegen. "Deriving all passenger flows in a railway network from ticket sales data". In: *Proceedings of the 4th International Seminar on Railway Operations Modelling and Analysis (IAROR): RailRome2011, February 16-18*. Rome, Italy, 2011.

[53] *Standaardfiche 36 Brussel-Noord Liège-Guillemins 36*. [Last accessed 05/12/2016]. 2016. URL: http://wayback.archive.org/web/20080516130438/http://users.pandora.be/brail/lijn/lijn36.htm.

[54] J. Törnquist and J. A. Persson. "*N*-tracked railway traffic re-scheduling during disturbances". In: *Transportation Research Part B: Methodological* 41.3 (2007), pp. 342–362.

[55] UIC. *UIC Code 406 - Capacity*. Tech. rep. Paris, France: Union International des Chemins de Fer, 2004.

[56] S. Van Thielen, S. Burggraeve, and P. Vansteenwegen. "Optimal train rescheduling after conflict detection". In: *Conference on Advanced Systems in Public Transport (CASPT2015), July 19-22*. Rotterdam, the Netherlands, 2015.

[57] L. P. Veelenturf, M. P. Kidd, V. Cacchiani, L. G. Kroon, and P. Toth. "A railway timetable rescheduling approach for handling large-scale disruptions". In: *Transportation Science* 50.3 (2015), pp. 841–862.

[58] D. van de Velde, C. Nash, A. Smith, F. Mizutani, S. Uranishi, M. Lijesen, and F. Zschoche. *EVES–Rail - Economic effects of Vertical Separation in the railway sector*. Tech. rep. Amsterdam, the Netherlands: inno-V, 2012.

[59] R. Watson and G. Medeossi. "Simulation". In: *Railway Timetabling & Operations: Analysis - Modelling - Optimisation - Simulation - Performance Evaluation*. Ed. by I. A. Hansen and J. Pachl. Hamburg, Germany: Eurailpress, 2014, pp. 191–215.

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

[61] L. Zhang, L. Lu, and J. Yuan. “Single machine scheduling with release dates and rejection”. In: *European Journal of Operational Research* 198.3 (2009), pp. 975–978.

Fiche masterproef

Student: Sander Van Aken

Titel: Optimal timetables for temporarily unavailable tracks

Nederlandse titel: Optimale dienstregelingen tijdens tijdelijke, gedeeltelijke blokkades van het spoor

UDC: 621.3

Korte inhoud:

Train passengers expect a high level of service under all circumstances, while disruptions occur on a daily basis on the Belgian railway network, and dispatchers at Infrabel still have to rely on their experience to tackle them. This thesis develops mathematical models that can support dispatchers during disruptions resulting in capacity reduction, on three types of frequently encountered parts of the infrastructure (AIPs), outside station areas.

A first AIP consists of a double-track corridor with a double switch on both sides, modelled as a single-machine scheduling problem with rejection, which accounts for train re-timing, reordering, rerouting and cancellation. Next, the second AIP incorporates stops along the corridor. The model is extended with discretely controllable process times to account for stop-skipping as an additional measure. Finally, disruptions on multi-track corridors are modelled as a parallel-machine scheduling problem, thereby accounting for conflicts on the aligning junctions.

Applying these models on different practical case studies allows to formulate service constraints, which dispatchers could consider. Additionally, all models are subjected to a theoretical analysis of their parameters' impact. The results obtained by solving the mathematical models are compared to two heuristics, representative for dispatcher's decision making.

The models outperform both heuristics by effectively balancing between all possible measures. Original timetable structure is identified as the major determinant for the generated disruption timetables. Results also suggest that interactions between measures are much more complex than one could intuitively expect, even in absence of additional service constraints. This clearly illustrates that these models could improve current dispatching practice.

Future research could evaluate the impact of the generated disruption timetables on the complete network, as conflicts might arise outside of the considered AIPs. To deal with the uncertain nature of disruptions and additional delays, the models could be embedded in a rolling horizon approach, rerunning them upon information updates. The AIP concept presents a powerful approach to model a wide range of disruptions, and future work could aim at identifying new ones, thereby developing models for them.

Dispatchers could employ the models by characterizing the disruption and the infrastructure around it with a limited set of parameters. This approach allows to combine the models' merit of efficiently balancing between measures, with the dispatcher's knowledge on the surroundings of the corridor, and passenger expectations.

Thesis voorgedragen tot het behalen van de graad van Master of Science in de ingenieurswetenschappen: verkeer, logistiek en intelligente transportsystemen

Promotor: Prof. dr. ir. Pieter Vansteenwegen

Assessoren: Prof. dr. ir. Herman Bruyninckx
Prof. dr. ir. Greet Vanden Berghe

Begeleider: Sofie Van Thielen