NETHERLANDS RAILWAYS
ERASMUS UNIVERSITY ROTTERDAM

MASTER THESIS

Railway Disruption Management in (Near) Out-of-Control Situations

Author:
Rolf van Lieshout
Student Number:
400376

Supervisor:
Prof. Dr. Dennis Huisman
Co-reader:
Dr. Paul Bouman

Econometrics and Management Science
Operations Research and Quantitative Logistics
Erasmus School of Economics

August 20, 2017

Abstract

This thesis investigates new models and methods for disruption management for railway systems in (near) out-of-control situations. On the one hand, we address the problem of finding an adapted line system that can be operated in the affected part of the railway network. On the other hand, we consider the problem of operating such an adapted line system in these situations, during which global coordination is impossible due to a lack of accurate and up-to-date information.

We develop a Benders'-like algorithm to generate profitable and passenger oriented line plans. The underlying mathematical model partially integrates line planning with timetabling and rolling stock scheduling as the line plan should be feasible with respect to infrastructural and resource restrictions. To operate the line plans, we propose several local train dispatching strategies, requiring varying degrees of flexibility and coordination.

Computational experiments based on disruptions in the Dutch railway network indicate that the algorithm performs well. For both a small and a large disrupted region the algorithm finds workable line plans within a couple of minutes. Simulation experiments also demonstrate that the produced line plans can be operated smoothly by applying the proper dispatching strategies.

Contents

Chapter 1

Introduction

Smooth functioning of a railway system depends on the dynamic and synchronized interaction between the infrastructure network, the timetable, resource schedules and information systems. When one or more elements of the system experiences a defect, a disruption occurs that may frustrate operations for some time. In order to maintain a feasible schedule and reduce the inconvenience of passengers and cargo operators, it is often necessary to adjust the timetable, rolling stock circulation and/or crew schedule by performing rescheduling. In recent years, many models and algorithms have been suggested and implemented to provide computerized support for solving rescheduling problems. However, from time to time disruptions continue to accumulate or an extreme incident occurs, causing the system to get into a state of (near) *out-of-control*, in which barely any traffic is possible in a part of the network. Because of the large number of affected trips and resources and the absence of detailed and accurate information, currently developed methods cannot be applied in these situations.

This thesis explores a newly proposed strategy to cope with (near) out-of-control situations. The core idea of this strategy is to completely decouple the operations in the disrupted region from the rest of the railway network, with the aim to isolate the disruption. In the disrupted region, a simplified line system is required, allowing for smooth operations even when the information about the resources in the system degrades and centralized scheduling is impossible. This research addresses (i) how such a simplified line system can be determined and (ii) how such a line system can be operated in a (near) out-of-control situation.

Since existing line planning models are solely used in strategic contexts and do not ensure feasibility in later planning stages, novel mathematical models are developed in this thesis to find optimal adjusted line plans in (near) out-of-control situations. To examine how these line plans should be operated, several local train dispatching strategies are proposed. These strategies require only local coordination, such that they are robust against the lack of overview and information that characterizes out-of-control situations. The effectiveness of the strategies is determined by simulating railway traffic in (near) out-of-control situations.

1.1 Disruption Management and Out-of-Control Situations

Railway operators continuously monitor the status of all involved resources and need to react to unexpected deviations from the plan. The process of how to continue operations after one or more incidents prevent the original schedule from being executed is commonly referred to as *disruption management* (Yu & Qi, 2004). In railway systems specifically, disruption management involves finding a new timetable by rerouting, retiming (delaying) and canceling trains services and rescheduling the rolling stock and crew such that the adapted timetable is compatible with the adapted resource schedules (Jespersen-Groth et al., 2009).

Naturally, whether a certain rescheduling action is appropriate depends on the nature of the disruption. Minor incidents or *disturbances*, for example when the boarding of passengers takes longer than expected, can usually be handled manually as they do not require significant changes. Major incidents or *disruptions*, for example accidents or rolling stock breakdowns, often constitute large combinatorial optimization problems that require specialized algorithms to find good solutions.

This thesis addresses disruption management in (near) out-of-control situations. Typically, control of the railway system is being lost after a combination of several disruptions or a single extreme incident. This leads to an immensely complex problem for the dispatchers, especially since the affected number of resources can be very large and the duration of the disruption is often uncertain. As a result, disruptions continue to accumulate and decisions made by the dispatchers are based on information that is out-of-date, rendering the decision unworkable. In the end, dispatchers are confronted with a lack of accurate and up-to-date information on the current state of the system, preventing them from making viable rescheduling decisions. This causes the disruption to further propagate through the network, making hardly any traffic possible in the affected region.

In the Dutch railway network, out-of-control situations happened about ten times during the period 2009-2012 because of bad weather conditions. For this reason, Netherlands Railways (NS), the largest railway operator in the Netherlands, decided to anticipate on such events by operating a reduced timetable if bad weather is expected. This reduces the probability of losing control, but the downside is that a worse product is offered to the passengers. Moreover, there are also examples of out-of-control situations with completely unexpected causes, such as the power outages in Amsterdam in 2015 and 2017 and the interruption of train services due to a potential terrorist attack in Rotterdam in 2015.

1.2 Organization of the Dutch Railway Sector

The methods developed in this thesis are tested on the railway system in the Netherlands. Multiple railway operators make use of the dense Dutch railway network, which consists of about 7000 kilometers of tracks. ProRail is responsible for maintaining and managing the railway infrastructure. Next to that, ProRail controls the timetable on the day of operations and acts as a referee if the timetables of operators are conflicting in the planning phase. The methods developed in this thesis are tested using the network of NS, which handles over one million passenger trips each day. Two types of trains are operated by NS. *Intercity* trains connect major cities in the Netherlands and do not dwell at any in-between stations. *Regional* trains usually perform shorter trips and do stop at every station. As operating a simplified line system has direct consequences for the timetable, the methods developed in this thesis can only be implemented in practice by close cooperation between NS and ProRail.

The planning at NS can be broken down into several stages:

- *line planing*; determining between which stations direct trains are operated, the stopping patterns and with which frequency the lines are operated,
- *timetabling*; specifying the exact departure and arrival times of trains,
- *rolling stock scheduling*; determining the train units that are used for each trip and
- *crew scheduling*; assigning all tasks to the conductors and drivers.

The planning at NS is done sequentially, meaning that the result of one planning phase is used as input for the following planning phases (Kroon et al., 2009). Line planning is a strategic (long term) planning problem and based on aggregated (expected) passenger flows. When the line plan is known, a timetable is constructed. NS operates a periodic timetable with a period of 1 hour, meaning that the timetable of every hour is (approximately) the same. Both the line plan and timetable remain fairly fixed over the years. The last major change has been implemented in 2016, but the major change before that was in 2006. Rolling stock and crew scheduling are operational planning problems and solved around six times a year. These problems are also handled sequentially, since allowed crew duties depend on the types of trains that are used. The rolling stock circulation is determined based on short term predictions of the number of passengers on every train. The crew duties must adhere to numerous labor requirements.

In the Dutch railway network, traffic controllers and dispatchers at the Operational Control Center Rail (OCCR), five regional rescheduling centers (RBCs) and 13 traffic control centers (VL posts) are responsible for carrying out disruption management. The OCCR tries to keep an overview of the entire system, such that a disruption at a certain location can be solved without introducing a problem elsewhere. Usually, a disruption is handled by sequentially adapting the timetable, rolling stock schedule and finally the crew schedule. To accelerate and improve the rescheduling process, ProRail and NS have developed *emergency scenarios* that specify how the timetable is adjusted when a certain disruption occurs at a certain location. For combinations of disruptions, emergency scenarios do not exist. As for algorithmic support, the dispatchers have access to a solver that reschedules the drivers and conductors. This solver is based on the algorithm developed in Potthoff, Huisman, and Desaulniers (2010). However, this algorithm can only be applied in specific types of disruptions. The RBCs are responsible for solving local rescheduling problems considering the rolling stock schedules and crew schedules. Next to that, the RBCs communicate rescheduling decisions to the train crews and coordinate shunting movements. The traffic controllers and signallers at the VL posts monitor the trains and operate the signal in their respective regions and perform local timetable rescheduling.

1.3 Strategy for Handling Out-of-Control Situations

Evidently, conventional disruption management approaches are not viable in (near) out-of-control situations. For this reason, NS and ProRail have initiated a research project together with Erasmus University Rotterdam, Utrecht University and Delft University of Technology with the aim to improve the resilience of railway systems. One of the sub-projects involves determining effective measures when the system is out-of-control or an early warning signal is received that control is about to be lost.

The global strategy that is proposed to deal with out-of-control situations is described in the research proposal (Panja, 2016). It consists of the following steps:

1. Identify and isolate the disrupted region.
2. Adjust the timetable, rolling stock and crew for the non-disrupted region according to existing disruption management techniques.
3. Determine a simplified line system and timetable to operate in the disrupted region.
4. Schedule rolling stock and crew within the disrupted region according to *self-organising, local* principles.
5. Manage the passenger flows.

The first step aims to prevent the disruption from spreading even further. Once the disrupted region is identified, no resources are allowed to move from the disrupted region to the non-disrupted region or vice versa. One of the sub-projects of the research project will entail finding the optimal way to separate the network in a disrupted region and a non-disrupted region. In this thesis, it is assumed this split is given.

In the second step, the schedules in the non-disrupted region are adjusted in such a way that the traffic in this region runs (approximately) as it normally would. To this end, the emergency scenarios are used that specify how trains should be turned when confronted with a blockage of certain tracks. From the perspective of the non-disrupted region, the separation of the two regions can be dealt with the same way as with (combinations of) track blockages. The rescheduling of the drivers and conductors is more complicated as crew duties must end at a certain location there and have to conform to labor restrictions.

The third step concerns finding a line system and compatible timetable that can be operated in the disrupted region, such that a reasonably good product can still be offered to the passengers. The simplified line plan will specify which lines to operate at which frequency. Most likely, no detailed timetable will be calculated. However, it is necessary for the adapted line plan to be feasible, meaning that a feasible timetable exists with enough time between trains and where the station capacity is respected. Furthermore, there must be enough resources available in the disrupted region to operate the line plan.

The fourth step involves scheduling the resources in the disrupted region. Since out-of-control situations are characterized with great uncertainty regarding the exact whereabouts of the rolling stock and crew, it is not possible to communicate detailed instructions to the crew. Instead, the idea is to provide a strategy on what task to do next. Given that a simple line plan is operated in the disrupted region, it should be possible to find appropriate strategies that lead to workable solutions.

In the fifth step, the passenger flows are managed. Since the line plan is adjusted, passengers also have to be routed differently through the network. Furthermore, since the disrupted region might still be subject to some irregularities in the operations, it will be a challenge to provide the passengers with proper information on how to travel to their destination.

This thesis investigates the third and a part of the fourth step of the described strategy for dealing with out-of-control situations. However, it must be noted that while all steps may look like a sequence of independent problems, they are actually all interdependent. Decisions made in an earlier stage strongly impact the problems that are faced in later stages. Therefore, it is required that methods for performing one of the steps take the other steps into account as well.

1.4 Contribution of the Thesis

This thesis is a follow-up on the work of Schouten (2017), who also addresses line planning in (near) out-of-control situations. However, the suggested solution approach comes with long computation times and results in line plans that strongly deviate from the regular line plan and contain many oddly shaped lines. Next to that, it is unlikely that the produced line plans have feasible timetables and rolling stock schedules. Even more, the used algorithm rules out many potentially good line plans.

The contribution of this thesis is threefold. The first contribution is a novel algorithm for line (re-)planning. This algorithm is designed to generate profitable, passenger oriented line plans that can be operated in (near) out-of-control situation. This is achieved by (i) including timetabling and rolling stock scheduling restrictions in the mathematical formulation for the line planning problem and (ii) iterating between a line planning problem and multiple smaller timetabling problems. In other words, this algorithm partially integrates line planning with timetabling and rolling stock scheduling. The second contribution of this research is that it proposes several local dispatching strategies that can be applied in (near) out-of-control situations. These strategies instantiate a self-organizing railway system that is analyzed with respect to several performance measures. The final contribution is the evaluation of the produced line plans and suggested dispatching strategies by simulating (near) out-of-control situations. In particular, this thesis illustrates that by applying the proper dispatching strategies, the line plans indeed lead to feasible operations, as intended.

1.5 Structure of the Thesis

The structure of the thesis is as follows. In Chapter 2, the problem is described in detail. In Chapter 3, relevant literature is discussed. Chapter 4 addresses the solution approach. Chapter 5 describes the local dispatching strategies and discusses how to test the performance of line plans and dispatching strategies in an out-of-control situation by means of simulation. In Chapter 6, the results of both the line planning algorithm and the simulations are presented. The thesis is concluded in Chapter 7.

Chapter 2

Problem Description

In short, this thesis considers the problem of finding a simplified, practicable line plan for a certain disrupted region, a part of the railway network that is in a state of (near) out-of-control. We refer to this problem as the line planning problem in out-of-control situations (LPOC). This chapter is dedicated to providing a more thorough definition of the LPOC and explaining the different aspects that play a role in this problem. First, we describe how the disrupted region can be identified. Next, we provide the definitions of lines and line plans and discuss when a line plan is considered to be feasible. Finally, we discuss the objectives of the LPOC that should be considered.

2.1 Disrupted Region

The first step in dealing with an out-of-control situation is identifying and isolating the disrupted region. The disrupted region is the part of the railway network that is isolated and for which a new simplified line plan needs to be determined. The methods developed in this thesis are applicable to arbitrary disrupted regions, but we test the methods on instances where the disrupted region was determined using the method of Schouten (2017).

Central in disruption management in the Dutch railway network are *decoupling stations*. At these stations, trains are allowed to be turned when the regular path of a train is blocked. In the network of NS, there are multiple types of decoupling stations. The most important stations are *intercity decoupling stations*. At these stations both intercity and regional trains are allowed to turn. At *regional decoupling stations*, only regional trains are allowed to turn. The remaining stations are referred to as *basic stations*.

As in (near) out-of-control situations trains can only turn at decoupling stations, it makes sense to take this into account when determining the disrupted region. Hence, Schouten (2017) assumes that the location of the disruption is known and defines the disrupted region to be the smallest part of the railway network that contains the disruption and is enclosed by intercity decoupling stations. That is, the boundary stations must be intercity decoupling points. In order to isolate the disrupted region, trains heading towards the boundary stations, either from inside or outside the disrupted region, must turn at the boundary station or at an earlier decoupling station.

Outside the disrupted region, the traffic should run as usual. The exact turning patterns, and possible cancellations of trains can be derived from the predefined emergency scenarios developed by ProRail and NS. From these scenarios, we can also derive which platforms at the boundary stations are used to serve the undisrupted region. The remaining platforms are available for serving the disrupted region.

2.2 Lines and Line Plans

A line is a direct train service between two stations (called the terminal stations) with a certain frequency, route and dwelling pattern specifying at which in-between stations trains dwell. If the frequency of a line is x, this means in every period (60 minutes at NS) x trains are operated in both directions. Ideally, these trains run in a synchronized pattern. For instance, a train with a frequency of 4 should depart approximately every 15 minutes.

In the network of NS, lines can only start and end at decoupling stations. If both terminal stations are intercity decoupling points, both an intercity train and a regional train line can be operated between these stations. Intercity trains only dwell at intercity stations, whereas regional trains dwell at all stations. If one of the terminal stations is a regional decoupling point, only regional train lines can be operated between these stations.

A *line plan* or *line system* is the set of all lines that are operated in a railway network. Not all line plans are feasible in the sense that they can be operated. A line plan is only feasible if (i) there exists a timetable without conflicts and (ii) there exist feasible resource schedules.

The first condition ensures that all trains can be operated without interfering with other trains. This condition is particularly important in out-of-control situations as an inoperable timetable will cause trains queuing up at bottlenecks in the network, potentially resulting in repeated loss of control over the system. A key concept in timetabling is that a certain *headway time* needs to separate two trains that make use of the same track. Next to that, minimum dwell times at stations must be respected. Since the number of available platforms is limited at the stations at the boundary of the disrupted region, it is expected that the station capacity is a strongly limiting factor in the LPOC.

The second condition makes certain that there are enough trains available in the disrupted region, and that there are enough drivers that can drive the rolling stock. A complicating factor is that the exact number of trains required to operate a line plan depends on the timetable, which is of course not available. Furthermore, the current positions of the resources must be taken into account. In theory, it is possible to perform shunting movements to retrieve trains from shunting yards or decouple a unit from a train, thereby increasing the number of trains that can perform trips. However, we assume this is not possible in (near) out-of-control situations.

In this thesis, we do not fully integrate the line planning problem with the timetabling and resource scheduling problems, as this gives an intractable problem. However, we do try to increase the likelihood the produced line plans are feasible with respect to the timetable and the number of rolling stock units that are present in the disrupted region. We assume that the crew that currently operates a train is able to continue operating that train.

2.3 Objectives

The main objective of the LPOC is to minimize the passenger inconvenience as caused by the (near) out-of-control situation. Given the severity of out-of-control situation, it will not be possible to offer the same service as in the regular schedule, but the discomfort should be minimized and distributed approximately uniformly over the passengers. The experiences of passengers are mostly dependent on whether the passenger has a seat, the travel time and how the expectation of the passenger compares with the actual outcome. To take these aspects into account, the adapted line plan (i) should be similar to the regular line plan in terms of the number of trains, (ii) should not require passengers to transfer much more often than usual and (ii) should be easy to understand and communicate.

Chapter 3

Literature Review

This chapter addresses relevant literature for line planning in out-of-control situation. First, we consider literature on disruption management techniques for railway rescheduling. Next, we consider literature on the strategic variant of the line planning problem. Finally, we consider earlier work on out-of-control situations in railway systems. For literature on other planning problems in passenger railway systems, we refer to Caprara, Kroon, and Toth (2011).

3.1 Real-Time Railway Rescheduling

When a disruption occurs, the timetable, rolling stock circulation and crew schedule need to be adjusted to obtain a new feasible schedule. Since solving this problem in an integrated manner leads to unacceptably long computation times, the problem is often decomposed and solved sequentially. First, the timetable is adjusted. The modified timetable then serves as input for the rolling stock rescheduling problem. Finally, both the adjusted timetable and rolling stock schedule are input for the crew rescheduling problem. An overview of proposed methods and algorithms is presented in Cacchiani et al. (2014).

3.1.1 Timetable Rescheduling

Timetable rescheduling deals with finding a new feasible timetable by canceling, retiming, rerouting or reordering trains services. Of the three rescheduling phases, timetable rescheduling has received the most attention in the literature. Approaches differ in the type of incident that has occurred (either a disturbance or a more serious disruption), in the level of detail the railway infrastructure is considered (either macroscopic or microscopic) and in the extent the inconvenience of passengers is taken into account. Objectives are usually to stay close to the regular timetable and minimize the total or maximum delay.

Many microscopic approaches formulate timetable rescheduling problems as job scheduling problems, in which a number of operations (the passing of trains) with certain operation times (running times) have to be scheduled on machines (block sections), see e.g. D'Ariano, Pacciarelli, and Pranzo (2007). In case of small delays, such models can be solved within a reasonable amount of time by utilizing implication rules of job scheduling theory. Macroscopic approaches use a higher level representation of the railway network, which has as an advantage that additional aspects can be incorporated. For example, Schöbel (2007) introduces the problem of delay management, where one decides whether trains depart on time or should wait for delayed feeder trains. The objective in delay management is usually to minimize the total delay of all passengers combined. Recently, this problem has been extended with the routing of passengers (Dollevoet, Huisman, Schmidt, & Schöbel, 2012) and the capacities of stations (Dollevoet, Huisman, Kroon, Schmidt, & Schöbel, 2014). Only a few contributions consider timetable rescheduling after disruptions. Louwerse and Huisman (2014) introduce the problem of finding a new timetable in case of partial or complete blockades. Additional constraints are added to increase the probability that a feasible rolling stock schedule exists for the modified timetable. Veelenturf, Kidd, Cacchiani, Kroon, and Toth (2015) further extend this model by considering a larger part of the network, allowing rerouting of trains and incorporating the transition from the regular timetable to the modified timetable and back.

3.1.2 Rolling Stock Rescheduling

The rescheduling of rolling stock calls for adapting the rolling stock circulation to the modified timetable by changing the compositions of certain trains. Sometimes, this implies that shunting movements are canceled or that new shunting movements are introduced. In case no train units are available, train services must be canceled. Hence, the goal is usually to minimize a combination of the number of canceled trains, the number of changed shunting movements and the difference with the planned end-of-day inventory at the stations.

Nielsen, Kroon, and Maróti (2012) present a rolling horizon approach for rescheduling rolling stock. In this approach, the rolling stock is rescheduled periodically, as information about the disruption is updated. The used model is based on a mixed integer programming formulation of the rolling stock scheduling problem proposed in Fioole, Kroon, Maróti, and Schrijver (2006). Kroon, Maróti, and Nielsen (2014) use the same model but also take passenger flows into account when rescheduling the rolling stock. Since disruptions can cause passengers to take different paths, their model tries to facilitate this change in demand by adapting the rolling stock schedule. To solve the problem, the authors iteratively compute a rolling stock schedule and simulate the corresponding passenger flows, until a satisfactory overall solution is found. In Veelenturf, Kroon, and Maróti (2017) this model is extended by also allowing small timetable adjustments, namely introducing stops of trains at stations where they would normally not call. Haahr, Wagenaar, Veelenturf, and Kroon (2016) compare the composition model used by Nielsen et al. (2012) and Kroon et al. (2014) with a path based model and conclude that both models are fast enough to be used in rescheduling contexts.

3.1.3 Crew Rescheduling

When the timetable and rolling stock schedule are updated, it is known which tasks need to be executed by the train drivers and conductors. Crew rescheduling involves assigning these tasks to the crew members. Often, many changes are necessary to the crew schedules as disruptions cause many duties to become infeasible. For example, a train driver on a delayed train might arrive too late for his next task, such that this task must be performed by a different train driver. Many (labor) restrictions need to be respected when reassigning tasks, the most important one being that a crew duty must always end at the planned crew base. If a task cannot be assigned to any crew member, it must be canceled. This is especially undesired for driving tasks, as this requires the rolling stock schedule to be updated once more. Therefore, the objective in crew rescheduling is usually minimizing the number of canceled tasks and changes to duties.

Huisman (2007) addresses crew rescheduling in the context of scheduled maintenance operations. As the number of possible duties is very large, the problem is solved using a combination of column generation and Lagrangian relaxation. Potthoff et al. (2010) consider the crew rescheduling problem when a disruption has occurred that causes a blockage of a route. To keep the problem size tractable, first a core problem with a limited number of tasks is solved. In case the solution contains canceled tasks, tasks that are in some sense close to canceled tasks are added to the core problem. This process is repeated until all tasks are covered or a time limit is exceeded. Veelenturf, Potthoff, Huisman, and Kroon (2012) extend the crew rescheduling problem by also allowing retiming of trips. This increases scheduling flexibility, such that more tasks can be covered. A final paper in this stream of research is Veelenturf et al. (2014). Here, uncertainty with respect to the length of the disruption is taken into account by requiring that duties have feasible completions in a number of different scenarios. A completely different approach to crew rescheduling is taken by Abbink et al. (2010). In this paper, train drivers are represented by driver-agents. In case the duties of some drivers have become infeasible, the driver-agents try to solve this by swapping tasks between drivers.

3.2 Line Planning

There is a vast amount of research on line planning in public transportation networks. Models can be classified as being either cost- or passenger-oriented and differ in objectives and the decision that is considered (selection of lines and/or setting frequencies). Very few models integrate aspects from other planning phases into the line planning decision, because line planning is always considered in a strategic context. Therefore, it is assumed that restrictions with respect to the infrastructure or rolling stock can be dealt with later. We refer to Schöbel (2012) for a more extensive survey.

3.2.1 Line Pools

Given that not all paths in a public transportation network constitute a feasible line, most line planning models assume that lines have to be selected from a predefined line pool, consisting of a set of feasible and 'reasonable' lines. Route generation is performed in a preprocessing phase and usually done using shortest path based procedures (Kepaptsoglou & Karlaftis, 2009). In a more recent paper, Gattermann, Harbering, and Schöbel (2017) generate line pools using minimum spanning trees while taking the objective of the line planning model into account. Exceptions to using predefined line pools are some heuristic approaches, and the model from Borndörfer, Grötschel, and Pfetsch (2007), who generate profitable lines dynamically in a column generation algorithm. Park, Seo, Hong, and Rho (2013) extend this model to allow for different dwelling patterns, as opposed to stopping at every station.

3.2.2 Passenger Routing

Passenger oriented approaches to line planning make use of an origin-destination matrix with demand data. This allows for more sophisticated objectives to be considered, such as the number of direct travelers or the total (perceived) travel time. For the sake of simplicity, earlier line planning models assume that the routes that passengers take through the network are fixed. However, in reality different line plans may constitute different passenger flows.[1] More recently, researchers have started to integrate the routing of passengers in line planning models. Borndörfer et al. (2007), Nachtigall and Jerosch (2008) and Schöbel and Scholl (2006) integrate route assignment with line planning. However, in the proposed models it is assumed that passengers can be steered by the operator. Goerigk and Schmidt (2017) relax this assumption and let passengers travel on shortest path, while ensuring enough capacity is provided to transport all passengers. As this does result in a complex bilevel model, the authors propose a genetic algorithm for large instances.

[1] The same holds for timetables, see Schmidt and Schöbel (2015)

3.2.3 Integrated Approaches

As the goal of this thesis is to construct line plans that can readily be operated, we wish to integrate line planning with timetabling and resource scheduling (at least to some extent). Relatively few line planning models take aspects from subsequent planning stages into account, because line planning has exclusively been considered in strategic contexts.

Kaspi and Raviv (2013) develop a metaheuristic to solve the integrated line planning and timetabling problem. In every iteration of their heuristic, trains are randomly scheduled based on certain distribution parameters. The resulting operator costs and passenger inconvenience is used to change the parameters for the next iteration. Burggraeve, Bull, Vansteenwegen, and Lusby (2016) also iterate between the line planning and timetable phase, but use feedback from the timetable to make deterministic changes to the line plan. In the line planning problem, constraints are included that increase the likelihood a timetable exists with large enough buffer times between trains. With this approach they are able to construct timetables with larger minimum buffer times, thereby increasing the robustness.

Schöbel (2015) gives two formulations for a complete integration of line planning and timetabling. The first formulation is a linear integer program, where it is assumed to be known on which paths passengers will travel. In the second formulation, passengers choose their shortest path based on the line plan and timetable, but this does result in a quadratic program. The Periodic Event Scheduling Problem (PESP) is used to account for the timetable aspect of the problem. PESP-constraints are activated when the associated lines are selected. No computational results or solution approaches are presented.

Next to deciding the lines and frequencies to operate, Claessens, van Dijk, and Zwaneveld (1998) and Goossens, Van Hoesel, and Kroon (2004) also decide with how many carriages each line should be operated. Based on the length of lines, turnaround times and the selected frequencies, this can be used to compute a lower bound on the number of train units that are necessary to operate the line plan (the exact number depends on the timetable). Goossens, van Hoesel, and Kroon (2006) extend this model by considering multiple types of lines and rolling stock.

Liebchen and Möhring (2007) present modeling techniques to include both line planning and rolling stock considerations into the PESP that is used for timetabling. To partially incorporate line planning, they assume that line segments are already fixed, but these segments need to be matched at a certain station to constitute lines. Next to that, the authors show that the available amount of rolling stock for a line can be incorporated within the PESP framework.

3.3 (Near) Out-of-Control Situations

Railway systems getting into a state of (near) out-of-control is still an unexplored field of study, with only two relevant academic works. On the fundamental side, Ball, Panja, and Barkema (2016) use a simplified model of a railway network and investigate the stability properties of this system. They find that even modest fluctuations in the supply of crew can lead to a collapse of the system, where almost all trains queue up at a small number of stations. Extending this model with more features of railway networks will be one of the sub-projects of the upcoming research project on out-of-control situations.

Schouten (2017) is the first research addressing effective measures in (near) out-of-control situations. Similar to this thesis, it investigates how simplified line plans can be determined when railway systems get into a state of out-of-control. The author first proposes a basic model for line planning. This model aims to produce a feasible line plan by including an upper bound on the number of trains passing through a station based on halting, turning and headway times. Later, an extended model is developed that imposes that a line plan should allow for cyclic traffic on all edges and that has high probability a feasible platform allocation exists. Rolling stock restrictions are not taken into account. Both models have a quadratic objective in order to model that the penalty for canceling trains on an edge progressively becomes larger. The basic model is solved in a fraction of a second. On the other hand, computation times for the extended model reach up to 30 hours, clearly infeasible for use in actual out-of-control situations. Another disadvantage of the used approach is that it leads to solutions with a large number and oddly shaped lines. In this thesis, we aim to resolve the issues highlighted by Schouten (2017) and also further integrate the line planning with timetabling and rolling stock scheduling.

Chapter 4

Line Planning Algorithm

The objective of the LPOC is to find the optimal line plan, according to the criteria specified in Chapter 2, for which a feasible timetable and rolling stock schedule exists. It is possible to provide formulations for the complete integration of these problems, but this results in a very large integer program that is too time consuming to solve. Therefore, in this chapter we propose a new method to solve the LPOC.

4.1 Decomposition

Our solution approach rests on a decomposition of the LPOC in a *master* problem and a *slave* problem. The master problem corresponds to the line planning aspect of the LPOC and amounts to finding the optimal line plan subject to certain restrictions. The line plan produced by the master problem serves as input for the the slave problem, which is intended to capture the timetable aspect of the LPOC. The slave problem hence is a feasibility problem, where it is checked whether the line plan is timetable feasible. If it is feasible, we have found the optimal line plan for which a timetable exists and the algorithm terminates. If not, we can identify a combinatorial cut in terms of the variables of the master problem. The cut is then added to the master, after which the process iterates.

The purpose of the master problem is to suggest good line plans that can be fed to the slave problem. To limit the number of iterations of the algorithm, the solutions of the master problem should already have a high probability of having a feasible timetable and rolling stock schedule. Therefore, we derive several necessary conditions for the existence of a timetable and rolling stock schedule, in terms of the line planning variables. Whether incorporating more aspects in the master problem reduces the overall computation time is one of the aspects analyzed in this thesis.

The role of the slave problem is to quickly evaluate the feasibility of the line plan produced by the master problem and identify one or more violated inequalities or cuts when confronted with an infeasible solution. A straightforward way to solve the slave problem is to consider the timetabling problem induced by the proposed line plan and check whether a feasible solution is found. However, besides the large computational effort involved in solving a large timetabling problem, this approach has the disadvantage that only a single line plan is ruled out by the generated cut if the line plan is found to be infeasible. A better approach is to use the slave problem to derive minimal sets of inconsistencies. To this end, we do not try to compute a timetable for the entire network, but rather we consider every station independently and try to compute partial timetables. As only a few lines attend every station, this gives timetabling instances that can be solved quickly and allows us to identify small sets of inconsistencies, resulting in strong cuts that can be added to the master.

Evidently, the existence of feasible partial timetables for all stations is only a necessary and not a sufficient condition for the existence of a feasible timetable for the entire disrupted region. However, since the capacity of the boundary stations of the disrupted region is the major limiting factor in (near) out-of-control situations, it is expected that this usually suffices, which is also confirmed by preliminary experiments.

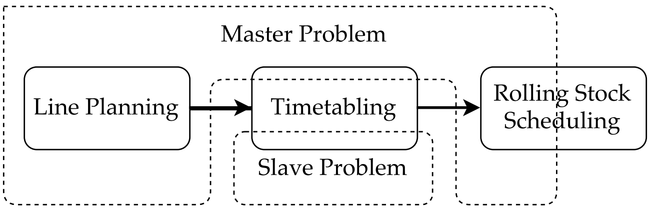


FIGURE 4.1: Decomposition of the planning problems into a master problem and a slave problem.

Figure 4.1 summarizes the decomposition. The master problem contains the entire line planning problem, but also includes parts of the timetabling and rolling stock scheduling problems. The slave problem captures a different side of the timetabling problem, but since we do not compute a timetable for the whole network, not the entire timetabling problem is integrated in the algorithm.

The above described decomposition of a problem into a master and a slave problem can be described as the integer or combinatorial variant of Benders' decomposition (Codato & Fischetti, 2006; Vanderbeck & Wolsey, 2010). It can be particularly useful for problems in railway transportation, as it provides an effective way to deal with the microscopic level of detail that ultimately determines whether a solution is feasible. In particular, in the master problem the railway network is considered at an highly aggregated level, in which every station is a node and the connections between stations form the edges. In the slave problem, the produced solution can be evaluated in a much more detailed representation of the railway network, that considers all non-shareable resources. Examples of successful applications of this macro/micro approach to railway problems are presented in Schlechte, Borndörfer, Erol, Graffagnino, and Swarat (2011) and Lamorgese, Mannino, and Piacentini (2016).

4.2 Solving the Master Problem

In this section, a mathematical formulation for the master problem is presented. We start by describing how we generate the set of candidate lines. Next, we discuss the basic line planning model proposed by Schouten (2017). In the remainder of the section, we show how the model can be extended with timetabling and rolling stock scheduling restrictions.

4.2.1 Line Pool Generation

To generate the set of allowed lines, we use a connection network $G = (S, E)$. The set S contains a node for every station and the set E contains an edge for every pair of stations between which a train can run without dwelling at an in-between station. In particular, we have *basic edges* connecting all pairs of stations that have no other station in-between, and *intercity edges* connecting all pairs of adjacent intercity stations. This network is similar to the type graph used by Goossens et al. (2004).

In Figure 4.2 the connection network corresponding to the Dutch railway network between Schiphol (Shl) and Utrecht (Ut) is depicted. The network contains four intercity stations and six basic stations. We say that every intercity edge *covers* multiple basic edges. For example, the intercity edge Asdz-Asb covers the basic edges Asdz-Rai and Rai-Asb.

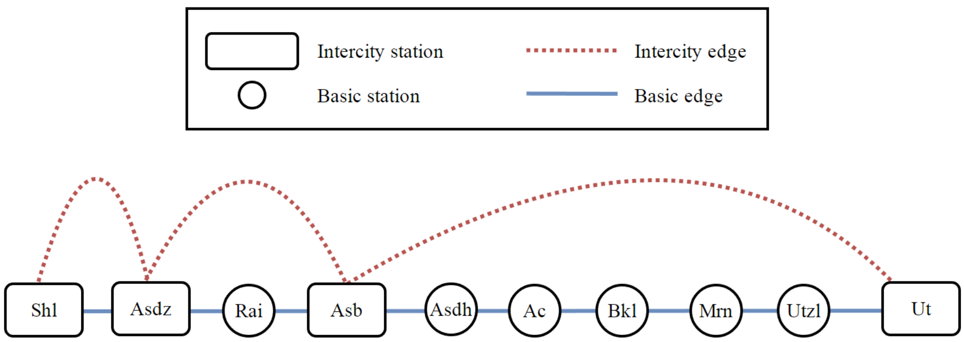


FIGURE 4.2: Connection graph of a part of the Dutch railway network.

A line l is defined by a frequency f_l and a set of connected edges representing the route $E_l = (e_1, e_2, ..., e_m)$. If l is a regional line, all edges must be basic edges and if l is an intercity line, all edges must be intercity edges. We denote the set of stations that line l attends (either crosses or dwells at) as S_l. The terminal stations of a line must be decoupling points compatible with the type of the line. For regional lines the set S_l is equivalent to the set of stations that appear in some edge in E_l. On the other hand, for intercity lines the set S_l also contains stations that l crosses without dwelling, and these are not included in the edges in E_l. The relevant stations can be found by looking at the basic edges that are covered by the intercity edges in E_l.

A final restriction on a line is that the railway infrastructure allows a train to travel along the sequence of edges in E_l in both directions.

To define the line pool we let g_e denote the frequency at which edge e is served in the regular timetable. Then, the line pool is defined as $\mathcal{L} = \{(E_l, f_l)\}$, where E_l respects the restrictions stated above and where $f_l \in \{1, 2, ..., \min_{e \in E_l} g_e\}$, since we do not want to operate more trains on an edge than in the regular timetable. Furthermore, we have the requirement that a line must be a shortest path between the terminal stations, as this promotes lines to have logical shapes. Next to that, most lines in the regular line plan of NS are shortest paths between the terminal stations, hence by imposing this restriction we are more likely to find line plans similar to the regular line plan operated by NS, making it easier to communicate the plan to passengers and crew.

4.2.2 Basic Model

To give a mathematical formulation of the basic line planning model, we first introduce some more notation. We let P^s denote the set of platforms at station s and let τ_l^s denotes the time a platform at station s is blocked when a train from line l attends station $s \in S_l$. This time includes the headway time that needs to separate two trains using the same track, and dwelling time if l stops at s. The parameter T is the period of the timetable. Next to that, the parameter m_{ls} equals 1 if s is a terminal station of line l and 2 otherwise. We introduce the decision variables x_l, indicating whether line l is selected and z_e, representing the number of trains canceled on edge e. The basic model given in Schouten (2017) for the LPOC can then be formulated as

$$
\begin{aligned}
\text{minimize} \quad & \sum_{e \in E} (\frac{z_e}{g_e})^2 + w \sum_{l \in \mathcal{L}} x_l && && (4.1)\\
\text{s.t.} \quad & \sum_{l \in \mathcal{L} | e \in E_l} f_l x_l + z_e = g_e && \forall e \in E, && (4.2)\\
& \sum_{l \in \mathcal{L} | s \in S_l} \tau_l^s m_{ls} f_l x_l \leq T|P^s| && \forall s \in S, && (4.3)\\
& z_e \geq 0 \text{ and integer} && \forall l \in \mathcal{L}, \quad \forall e \in E, && (4.4)\\
& x_l \in \{0, 1\} && \forall l \in \mathcal{L}. && (4.5)
\end{aligned}
$$

The objective function is discussed in the next section. Constraints (4.2) ensure that the z-variables obtain their correct values. Constraints (4.3) guarantee that the total time platforms are used at a station is not larger than the total available time. Note that the multiplier m_{ls} is necessary as at in-between stations twice as much trains belonging to a certain line attend the station than at terminal stations. As constraints (4.4) impose that z_e is nonnegative, it is not possible to operate a frequency on an edge that is higher than the frequency in the original schedule.

4.2.3 Objective Function

The primal objective of the model used in Schouten (2017) is to minimize the decrease in frequency on all edges. We refer to this objective as the *edge objective*. To prevent finding solutions with many short lines, the objective also includes a term penalizing the number of lines. The parameter w is used to weigh the two objectives. A quadratic objective is used for the decrease in edge frequency (i) to distribute cancellations evenly over the edges and (ii) to penalize larger relative decreases more strongly. That is, if two edges have the same original frequency, we prefer to cancel one train on both edges instead of canceling two trains on one of the edges. Moreover, if two edges have a different frequency we prefer to cancel a train on the edge with the higher frequency, as this results in a lower relative decrease.

Although the edge objective used by Schouten (2017) is intuitively appealing, it has two disadvantages. Firstly, the objective is a quadratic function of the decision variables. In general, quadratic programs are more difficult to solve than linear programs. Schouten (2017) indeed reported very long computation times for this model, unworkable for use in disruption management. Secondly, the passenger flows are not taken into account. Every edge has the same weight, despite that the number of passengers on some edges are much larger than on others. In this section, we propose methods to resolve both issues.

Linearizing the Quadratic Program
Here we provide an equivalent linear formulation of the quadratic objective. To this end, we introduce the auxiliary variables $u_{e1}, u_{e2}, ..., u_{eg_e}$ For every edge e with the following definition:

$$u_{ei} = \begin{cases} 1, & \text{if at least } i \text{ trains are canceled on edge } e, \\ 0, & \text{otherwise.} \end{cases}$$

This can be modeled by replacing constraints (4.2) by the following constraints:

$$\sum_{l \in \mathcal{L} | e \in E_l} f_l x_l + \sum_{i=1}^{g_e} u_{ei} = g_e \quad \forall e \in E, \tag{4.6}$$

$$u_{e1} \geq u_{e2} \geq ... \geq u_{eg_e} \quad \forall e \in E, \tag{4.7}$$

$$u_{ei} \in \{0, 1\} \quad \forall e \in E, \quad i \in \{1, ..., g_e\}. \tag{4.8}$$

Next, we replace the objective (4.1) by

$$\text{minimize} \quad w \sum_{l \in \mathcal{L}} x_l + \sum_{e \in E} \sum_{i=1}^{g_e} c_{ei} u_{ei} \tag{4.9}$$

where the cost coefficients c_{ei} are defined as follows:

$$c_{ei} = \left(\frac{i}{g_e}\right)^2 - \left(\frac{i-1}{g_e}\right)^2. \tag{4.10}$$

To see the equivalence between the two formulations, consider an edge e where z trains are canceled. In the quadratic model, $z_e = z$, such that the contribution in the objective equals $\left(\frac{z}{g_e}\right)^2$. In the linearized model, this gives $u_{e1} = u_{e2} = ... = u_{ez} = 1$ and $u_{e(z+1)} = u_{e(z+2)} = ... = u_{eg_e} = 0$, resulting in an objective contribution of $\sum_{i=1}^{z} c_{ei} = \sum_{i=1}^{z} \left(\frac{i}{g_e}\right)^2 - \left(\frac{i-1}{g_e}\right)^2 = \left(\frac{z}{g_e}\right)^2 - \left(\frac{z-1}{g_e}\right)^2 + \left(\frac{z-1}{g_e}\right)^2 - ... - \left(\frac{1}{g_e}\right)^2 + \left(\frac{1}{g_e}\right)^2 = \left(\frac{z}{g_e}\right)^2$. As the contribution to the objective value is the same for both formulations, they are equivalent.

Finally, note that since the cost coefficients are increasing in i and we are minimizing, we can omit constraints (4.7).

The OD objective

To take passenger flows into account in the master problem, we introduce the *OD objective*. Where the original objective aims to maintain edge frequencies, the OD objective aims to maintain frequencies for origin-destination pairs, weighted with the number of passengers. The frequency of an OD pair is defined as the minimum frequency of the edges in the shortest path between the origin and destination. Transfers are not taken into account, as this further increases complexity.

For the mathematical formulation, we let $\rho_{s,s'}$ denote the shortest path between stations s and s'. The shortest paths are computed using the running times on all edges. The actual shortest paths of passengers depend on the line plan and timetable, but this is a reasonably good approximation. The original frequency of OD (s, s') is then given by $g_{s,s'} = \min_{e \in \rho_{s,s'}} g_e$. We now introduce the decision variables $\phi_{s,s'}$, representing the frequency of OD pair (s, s') in the adapted line plan and $z_{s,s'}$, representing the decrease in frequency of OD (s, s'). These relations are modeled using the following constraints:

$$\phi_{s,s'} + z_{s,s} = g_{s,s'} \qquad \forall (s,s') \in S \times S, \tag{4.11}$$

$$\phi_{s,s'} \leq \sum_{l \in \mathcal{L} | e \in E_l} f_l x_l \qquad \forall (s,s') \in S \times S, \quad \forall e \in \rho_{s,s'}, \tag{4.12}$$

$$\phi_{s,s'}, z_{s,s} \geq 0 \text{ and integer} \qquad \forall (s,s') \in S \times S. \tag{4.13}$$

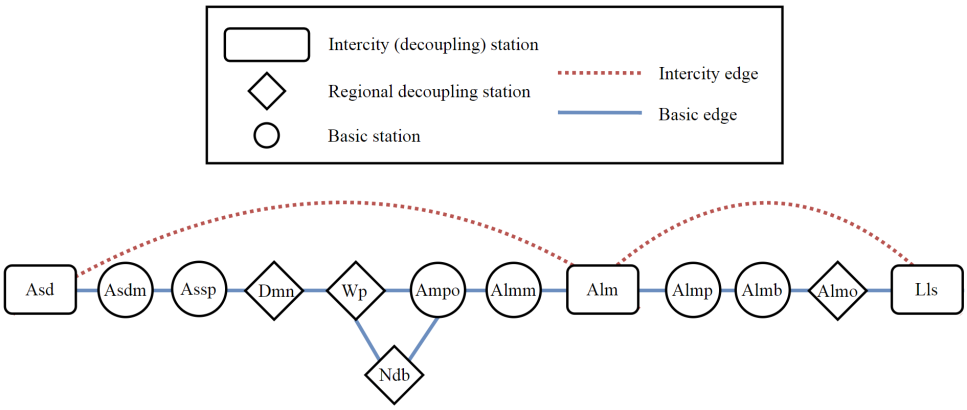
Constraints (4.11) ascertain that the z-variables obtain their correct value and constraints (4.12) make sure that the frequency of every OD pair is equal to the minimum of the edge frequencies in the shortest path of the OD. The OD objective is now given by

$$\text{minimize} \quad \sum_{(s,s') \in S \times S} n_{s,s'} (\frac{z_{s,s'}}{g_{s,s'}})^2 + w \sum_{l \in \mathcal{L}} x_l, \tag{4.14}$$

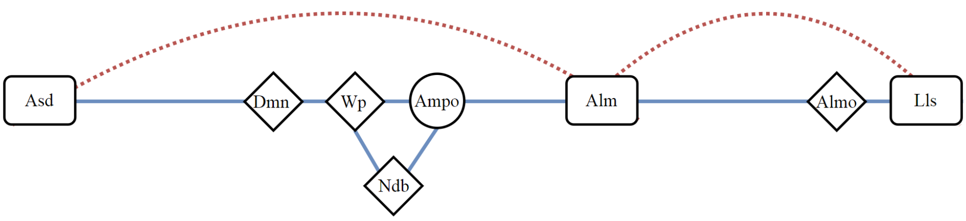
where $n_{s,s'}$ equals weight of OD pair (s, s'). We let the weights equal the daily number of passengers that travel between s and s' divided by the total daily number of passengers. The OD objective is again quadratic, but can be linearized in the same way as the edge objective.

The disadvantage of the OD objective compared to the edge objective is that the number of OD pairs is quadratic in the number of stations, whereas the number of edges is roughly linear in the number of stations for railway networks. This implies that for larger networks it might be more computationally expensive to minimize the OD objective. Therefore, we now describe an aggregation technique that reduces the number of OD pairs that are required for the minimization.

The aggregation exploits the fact that a relatively small subset of the stations are decoupling stations or located at junctions in the railway network, such that the edge frequency can change only at a limited number of stations. Edges that will necessarily have the same frequency can be replaced by an aggregated edge. OD pairs whose shortest path uses the same aggregated edges can then be aggregated into an aggregated OD pair. This implies that aggregated ODs are of the form (e, e') instead of (s, s'). The mathematical formulation is very similar and therefore not provided.



(A) Before aggregation.



(B) After aggregation.

FIGURE 4.3: Example of a connection network before and after aggregation.

We illustrate the aggregation method with Figure 4.3. In this figure the connection network corresponding to the railway network between Amsterdam Centraal (Asd) and Lelystad Centrum (Lls) is visualized before and after aggregation. In the aggregated network, the stations adjacent to edges that will always have the same frequency are removed. Ampo (Almere Poort) is the only remaining basic station because it is located at a junction. ODs are aggregated based on the first and last aggregated edge that appear in the shortest path. For example, the ODs Asd-Ndb and Assp-Ndb both count towards the aggregated OD (Asd-Dmn,Wp-Ndb). The first and last aggregated edge can also be the same: the OD Asd-Alm counts towards the aggregated OD (Asd-Alm,Asd-Alm).

Before aggregation, the total number of ODs in the example equals $(|S| \times (|S|-1)/2 = 72$. Letting E^a denotes the number of aggregated edges, in the aggregated network the number of ODs equals $(|E^a| \times |E^a|)/2 = 50$. However, many of these OD pairs can be omitted because they cannot appear in a shortest path. For example, if the first edge is Almo-Lls, the only possible last edge is Alm-Almo. The reason is that passengers with origin (destination) Lls will only use the basic edge between Almo and Lls if their destination (origin) is Almp or Almb. When all infeasible OD pairs are omitted, we are left with 32 ODs in the aggregated network.

4.2.4 Timetabling Constraints

The integrated line planning and timetabling model of Schöbel (2015) extends a standard line planning model with PESP constraints. Using the Big-M technique, PESP events and constraints are activated if the associated lines are selected. Since the number of lines can be very large and timetabling in itself is already a very hard problem, this results in an intractable model (even for strategic planning purposes).

Therefore, in our approach we only derive necessary conditions for the existence of a timetable. The idea behind these conditions is that every line 'eats up' a certain amount of the capacity at stations and a timetable can only exist if not all capacity is utilized. Constraints (4.3) in the basic model are already an example of such a necessary condition, but we will show that by declaring some auxiliary variables we can provide much more precise and restrictive conditions.

Capacity Per Platform
Constraints (4.3) implicitly assume that all trains are spread uniformly over all platforms. However, even if all platforms are allowed for all lines, a capacity restriction per station can be inaccurate. For example, consider a station with 2 platforms where all trains need to turn. The minimum time a platform is blocked when a train is turned is 7 minutes. Therefore, in the original model $\lfloor (60 \times 2)/7 \rfloor = \lfloor 17.1 \rfloor = 17$ trains per hour are allowed to be scheduled at this station. However, maximally $\lfloor 60/7 \rfloor = \lfloor 8.6 \rfloor = 8$ trains can be scheduled per platform. Hence, the upper bound of 17 can be strengthened.

Furthermore, constraints (4.3) neglect the railway infrastructure at stations, which can bring about restrictions to the platforms that can be used by trains. Consider for example the infrastructure around station Weesp in Figure 4.4. Intercity trains will always use the two tracks in the middle where there is no physical platform (we do however consider these as platforms in the model). Regional trains coming from Amsterdam heading towards Almere can use both platform 1 and 2 (counting from the bottom upwards). On the other hand, regional trains heading towards Hilversum can only use platform 2. By incorporating this information in the model, we can impose a more accurate restriction on the number of trains that can be operated.

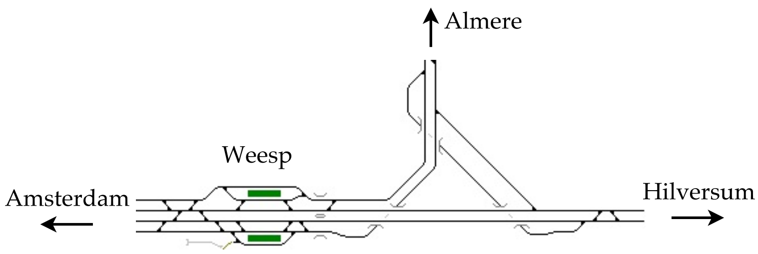


FIGURE 4.4: The railway infrastructure around station Weesp.

As for the mathematical formulation, we introduce the variables y_{lpd} that represent the number of trains of line l in a certain direction d that are assigned to some platform p. The parameter a_{lpd} indicates whether this assignment is allowed. We let D_l^s denote the set of directions of a line l at station s. If s is a terminal station on line l, this set only contains one element. Otherwise, it contains two elements. Next, we let P^s denote the set of platforms at station s. Then, the platform capacity can be modeled by the following constraints:

$$\sum_{p \in P^s} a_{lpd} y_{lpd} = f_l x_l \qquad \forall l \in \mathcal{L}, \quad \forall s \in S_l, \quad \forall d \in D_l^s, \tag{4.15}$$

$$\sum_{l \in \mathcal{L} | s \in S_l} \sum_{d \in D_l^s} \tau_l^s y_{lpd} \leq T \quad \forall s \in S, \quad \forall p \in P^s, \tag{4.16}$$

$$y_{lpd} \geq 0 \text{ and integer} \qquad \forall l \in \mathcal{L}, \quad \forall s \in S, \quad \forall p \in P, \quad \forall d \in D_l^s. \tag{4.17}$$

Constraints (4.15) assure that if a line is selected, in all directions all trains are assigned to platforms and constraints (4.16) impose that the total time a platform is blocked is smaller than the cycle time.

Dwell Times

In constraints (4.3) and (4.16), the values τ_l^s are equal to the minimum amount of time a platform needs to be blocked when a train of line l attends station s. However, in periodic timetables trains often have longer stops and turnaround times to enforce the periodic pattern. It follows that constraints (4.3) and (4.16) are actually not restrictive enough with regards to the time trains spend at stations.

To illustrate this, consider a line between stations A and B with a frequency of 2 and a travel time between A and B of 31 minutes and assume all trains turn on themselves. This means that when a train arrives at its terminal station, the train turns and starts performing the reverse trip of the trip the train just finished. In a periodic timetable with a period of 60 minutes, this means that without loss of generality trains depart at station A at minutes 0, 30, 60 and so on. Figure 4.5 depicts an example of a periodic timetable for this line. The train leaving A at 0 arrives at B at minute 31. Assuming a minimal turning time of 5 minutes, the soonest the train can again arrive at A is at minute 67. Likewise, the soonest the train is ready to depart again from A is at minute 72. However the earliest next trip it can perform starts at minute 90. This implies that on top of the minimum dwell times, we are certain the train has to dwell an additional 18 minutes at the stations on the line. In the figure, the additional dwell time is spent entirely at station A, but it is of course possible to spread this time over the stations where the train dwells.

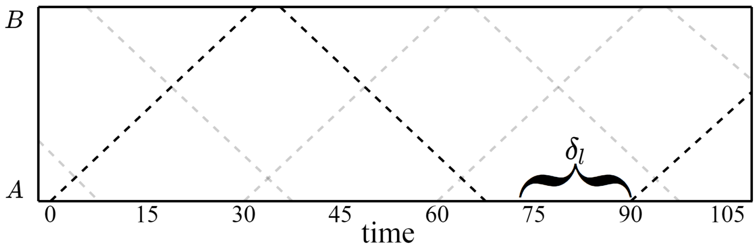


FIGURE 4.5: Time space diagram depicting the timetable of a line between two stations with frequency 2. The additional dwell time is denoted by δ_l.

In general, the additional dwell time δ_l of a line with minimum time between two trips t_l equals

$$\delta_l = \min_k \{ \frac{T}{f_l} k - t_l : \frac{T}{f_l} k - t_l \geq 0, k \in \mathbb{N} \}. \tag{4.18}$$

To incorporate this in the master problem, we declare variables and constraints that describe how the additional dwell time is divided over the stations on the lines. We refer to these constraints as the *dwell constraints*. First we show how this can be incorporated at station level. Thereafter, we show how it can be incorporated at platform level.

For every station on a line we define the variables t_{lsd} representing the additional dwell time of line l at station s in direction d. In a periodic timetable, this value is the same for every train of the line in the same direction. Then, the following set of constraints model how the additional dwell time is divided over the stations:

$$\sum_{s \in S_l} \sum_{d \in D_l^s} t_{lsd} = \delta_l x_l \qquad \forall l \in \mathcal{L}, \tag{4.19}$$

$$\sum_{l \in \mathcal{L}} \sum_{d \in D_l^s} f_l \tau_l^s x_l + f_l t_{lsd} \leq T|P^s| \quad \forall s \in S, \tag{4.20}$$

$$0 \leq t_{lsd} \leq \sigma_l^s \qquad \forall l \in \mathcal{L}, \quad \forall s \in S_l, \quad d \in D_l^s. \tag{4.21}$$

Constraints (4.19) ensure that if a line is selected, the additional dwell time is divided over the stations. Constraints (4.20) impose that the total time platforms are blocked at a station is less than the total available platform time. Constraints (4.21) guarantee that the additional dwelling time at a station is nonnegative and less than a specified upper bound.

The case with the restrictions on platform level is more complex as both the number of trains that dwell at a platform and the additional dwelling times are variables. To model this, the y_{lpd} variables are replaced by the more detailed binary variables y_{l_ipd}, which are equal to one if the i'th train of line l in direction d is assigned to platform p, where $i = 1, 2, ..., f_l$. Constraints (4.15) must then be replaced by the constraints:

$$\sum_{p \in P^s} a_{lpd} y_{l_ipd} = x_l \quad \forall l \in \mathcal{L}, \quad \forall i \in \{1, ..., f_l\}, \quad \forall s \in S_l, \quad \forall d \in D_l^s. \tag{4.22}$$

Now, the division of the additional dwell time over the stations can be modeled by replacing constraints (4.20) by the constraints:

$$\sum_{l \in \mathcal{L}} \sum_{d \in D_l^s} \sum_{i=1}^{f_l} (\tau_l^s + t_{lsd}) y_{l_ipd} \le T \quad \forall s \in S, \quad \forall p \in P^s. \tag{4.23}$$

To linearize this set of constraints, we introduce new variables $t_{l_ipd} = t_{lsd} y_{l_ipd}$. This relation is enforced by adding the following linear constraints:

$$t_{l_ipd} \le \sigma_l^s y_{l_ipd} \qquad \forall l \in \mathcal{L}, \quad \forall i \in \{1, ..., f_l\}, \quad \forall s \in S_l, \quad \forall p \in P^s, \quad \forall d \in D_l^s, \tag{4.24}$$

$$t_{l_ipd} \le t_{lsd} \qquad \forall l \in \mathcal{L}, \quad \forall i \in \{1, ..., f_l\}, \quad \forall s \in S_l, \quad \forall p \in P^s, \quad \forall d \in D_l^s, \tag{4.25}$$

$$t_{l_ipd} \ge t_{lsd} - \sigma_l^s (1 - y_{l_ipd}) \quad \forall l \in \mathcal{L}, \quad \forall i \in \{1, ..., f_l\}, \quad \forall s \in S_l, \quad \forall p \in P^s, \quad \forall d \in D_l^s. \tag{4.26}$$

It might be attractive to set the upper bounds at 0 for all stations except terminal stations, such that the travel times between the terminal stations are no longer than the minimum travel times. However, this could be too restrictive, especially since we might have limited capacity at the terminal stations. An alternative is too put an upper bound on the total additional dwell time during a trip. By doing so, we decrease the likelihood of the the scenario that in order to realize a feasible timetable, travel times must be far larger then usual. This constraint can be formulated as follows:

$$\sum_{s \in S_l | s \notin \mathrm{TRM}(l)} t_{lsd} \le \delta_l^{\max} x_l \qquad d = 1, 2 \quad \forall l \in \mathcal{L}, \tag{4.27}$$

where $\mathrm{TRM}(l)$ is the set containing the two terminal stations of line l and $\delta_l^{\max}$ denotes the imposed maximum.

4.2.5 Rolling Stock Scheduling Constraints

In the planning context, rolling stock scheduling is performed once the timetable is known, since the the number of trains required to operate lines depends on the exact trip times and therefore on the timetable. In (near) out-of-control situations on the other hand, it is essential to take rolling stock restrictions into account when solving the line planning problem as there are only a limited number of trains available in the disrupted region. Furthermore, in such situations it might be very hard to perform shunting movements to increase the number of operating trains. Claessens et al. (1998) and Goossens et al. (2006) do include rolling stock costs in the objective of their line planning model based on a lower bound on the number of necessary trains (rolling stock compositions). Such a lower bound is easily computed if we assume all lines are operated with fixed rolling stock circulations, in which all trains turn on themselves. We first demonstrate how to use this lower bound for a necessary condition of the existence of a rolling stock schedule with fixed circulations. Thereafter, we present a new model that allows for flexible rolling stock circulations.

Fixed Rolling Stock Circulations
The case with fixed rolling stock circulations can be formulated using assignment constraints. We first define some notation. We let R denote the set of trains located in the disrupted region. Here, a train is a rolling stock composition that is currently operating. We assume it is not possible to split or combine trains or to retrieve additional trains from shunting yards. The parameter b_{rl} indicates whether train r can be assigned to line l. This depends on the current location of the train and on whether the type of the train is compatible with the type of the line (intercity trains cannot be used for regional lines and vice versa). Next, we let n_l denote how many trains are at least necessary to operate line l. This value can be computed as follows:

$$n_l = \left\lceil \frac{t_l}{T} f_l \right\rceil, \tag{4.28}$$

where t_l is the minimum time it takes for a train to perform a full circulation of line l. Now, we introduce decision variables v_{rl} that are equal to 1 if train r is assigned to line l. The allocation of rolling stock can be included in the master problem by adding the following constraints:

$$\sum_{r \in R} b_{rl} v_{rl} \geq n_l x_l \quad \forall l \in \mathcal{L}, \tag{4.29}$$

$$\sum_{l \in \mathcal{L}} v_{rl} \leq 1 \quad \forall r \in R, \tag{4.30}$$

$$v_{rl} \in \{0, 1\} \quad \forall l \in \mathcal{L}, \quad \forall r \in R. \tag{4.31}$$

This set of constraints states that if a line is selected, at least n_l trains should be assigned to the line and that every train can only be assigned to a single line.

Flexible Rolling Stock Circulations
When a railway system is operated with flexible rolling stock circulations trains can switch lines when they reach their terminal station. This leads to shorter turning times, which allows more trains to be operated as (i) trains spend less time dwelling and more time running and (ii) pressure is released at the turning stations, such that there is capacity for additional trains. Moreover, flexible circulations increase flexibility during operations, since if a train is delayed, a different train may take over its next trip.

Nielsen (2011) describes how flexible turning patterns can be incorporated in rolling stock rescheduling models. However, it is nontrivial how flexible rolling stock circulations can be allowed for in line planning. Therefore, in this section we propose a new mathematical formulation for modeling flexible rolling stock circulations.

It is possible to give a more general definition of circulations, but to keep the number of circulations limited we define a rolling stock circulation c as a sequence of lines $L(c) = l_1, l_2, ..., l_{|c|}$, such that all consecutive lines have a shared terminal station and the lines are either all regional or intercity lines. A train performing this circulation continuously traverses the sequence from left to right.

Now, let t_c denote the minimum time between two round trips of circulation c of the same train, taking into account the frequencies of the lines in the circulation. For example, if a line in the circulation has frequency 1, the possible values of t_c are $T, 2T, 3T, ...$ et cetera. In general, if we let f_{min} the minimum frequency of a line in the circulation, t_c must be an integer multiple of T/f_{min}. The crucial observation is that every period T, a train running on circulation c performs T/t_c trips of all lines in $L(c)$.

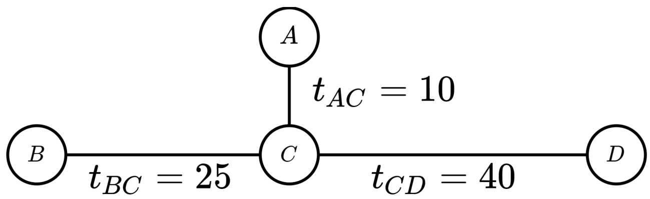


FIGURE 4.6: Simple network to illustrate flexible circulations.

In Figure 4.6 a simple network is depicted to illustrate the above stated concepts. Assume all lines only consist of a single edge and that the maximum frequency is 2. We can then denote the set of lines as $\mathcal{L} = \{l^1_{AC}, l^2_{AC}, l^1_{BC}, l^2_{BC}, l^1_{CD}, l^2_{CD}\}$. An example of a circulation is $c_1 = l^2_{AC}, l^2_{BC}$. Assuming a turning time of 5 minutes, a train performing this circulation starting at station C at time 0 returns at C at minute 10+5+10+5+25+5+25=85. As a consequence, the train can start its next circulation at minute 90, such that $t_{c_1} = 90$. Every 60 minutes, this train performs 2/3 of the trips of these lines every 60 minutes. This implies that if we select this circulation, we have to complement the rolling stock schedule with other circulations to exactly cover all trips.

To provide a mathematical formulation of flexible rolling stock circulations, we let $\mathcal{C}$ denote the set of all circulations. Furthermore, we introduce the decision variables γ_c, indicating whether circulation c is selected, and θ_c, representing how many trains perform circulation c. We also have the assignment variables v_{rc}, indicating whether train $r \in R$ is assigned to circulation c. We let the parameter b_{rc} indicate whether such an assignment is possible. The formulation now reads as follows:

$$\sum_{c \in \mathcal{C} | l \in L(c)} \frac{T}{t_c} \theta_c = f_l x_l \quad \forall l \in \mathcal{L}, \tag{4.32}$$

$$l_c \gamma_c \leq \theta_c \leq u_c \gamma_c \quad \forall c \in \mathcal{C} \tag{4.33}$$

$$\sum_{r \in R} b_{rc} v_{rc} = \theta_c \quad \forall c \in \mathcal{C}, \tag{4.34}$$

$$\sum_{c \in \mathcal{C}} v_{rc} \leq 1 \quad \forall r \in R, \tag{4.35}$$

$$\theta_c \geq 0 \text{ and integer} \quad \forall c \in \mathcal{C}, \tag{4.36}$$

$$\gamma_c, v_{rc} \in \{0, 1\} \quad \forall c \in \mathcal{C}, \quad \forall r \in R. \tag{4.37}$$

Constraints (4.32) guarantee that if a line is selected, we also select sufficient circulations that cover the line. Constraints (4.33) assure that that if a circulation is selected, the number of trains assigned to the circulation is between a certain upper and lower bound. These bounds are given by

$$l_c = \left\lceil \frac{t_c}{T} \right\rceil \quad \text{and} \quad u_c = \frac{t_c}{T} f_{\min}. \tag{4.38}$$

The lower bound ensures that every selected circulation accounts for at least one train service of each line in every hour. It is possible to further restrict the values of the θ-variables to for example impose that the turning patterns at stations repeat itself every hour. The upper bound is derived from constraints (4.32).

The assignment part of the formulation is covered by constraints (4.34) and (4.35). These constraints make certain that the number of trains assigned to a circulation equals the number of times the circulation is selected and that every train is assigned to at most one circulation.

To strengthen the linear programming relaxation of the above formulation, we also add the following constraints:

$$\gamma_c \leq x_l, \quad \forall c \in C, \quad \forall l \in L(c). \tag{4.39}$$

Finally, note that the dwell constraints defined in Section 4.2.4 are valid under the assumption of fixed circulations. It is possible to generalize these constraints, such that the additional dwell time is not distributed over the stations of all selected lines, but over the stations of the selected circulations. However, preliminary experiments showed that this constraint is rarely violated. The reason for this is that in order to use the trains as efficiently as possible, the model selects circulations with short turning times, such that the additional dwell time is very small. Therefore, we do not include this constraint in the master problem if we allow for flexible circulations.

4.3 Solving the Slave Problem

In this section we discuss how to solve the slave problem. First, we provide a general explanation of the Periodic Event Scheduling Problem that is used to define periodic timetabling problems. Thereafter, we describe how to identify violated cuts using the slave problem.

4.3.1 Periodic Event Scheduling Problem

Most models for finding a periodic timetable in railway networks are based on the Periodic Event Scheduling Problem (PESP). This problem is originally introduced by Serafini and Ukovich (1989), who also prove that the PESP is NP-complete. The PESP aims to schedule periodically recurring events, such that periodic restrictions between pairs of events are respected. In the railway context, events are arrivals and departures at stations and crossings at relevant points in the network such as junctions or bridges. An example of periodic restrictions are headway constraints, which state that the times between two departures or arrivals at a station are larger than the headway time.

Formulation

The PESP can be formulated concisely using an event-activity network $\mathcal{N} = (\mathcal{E}, \mathcal{A})$. The arrival of the i'th train of line l at station s is represented by an arrival node $(l, s, \text{arr}, i) \in \mathcal{E}^{\text{arr}}$. Similarly, departures are represented by nodes $(l, s, \text{dep}, i) \in \mathcal{E}^{\text{dep}}$. Junctions and bridges can also be treated as stations. In the most basic form, there are four types of activities (arcs) linking two nodes:

- Dwelling activities link arrival nodes (l, s, arr, i) to departure nodes (l, s, dep, i).
- Driving activities link the departure node (l, s_1, dep, i) at station s_1 to the arrival node (l, s_2, arr, i) at the next station s_2.
- Safety activities link departure nodes (l, s, dep, i) or arrival nodes (l, s, arr, i) with departure nodes (l', s, dep, j) or arrival nodes (l', s, arr, j).
- Synchronizing activities link departure nodes (l, s, dep, i) with departure nodes (l, s, dep, j), with $i \neq j$.

When timetabling is considered in the strategic context, transfer activities are also included in the event-activity network to create attractive connections between trains. However, these constraints are only used to create better timetables whereas our focus lies purely on feasibility. Hence, we do not include these activities in the network.

Every activity corresponds to a constraint stating that the duration of the activity should be in a certain interval. For the dwell activities, this is used to require that the dwelling times of trains are not too short and also not too long. For driving activities, the time between departure and arrival should be at least the minimum driving time. For safety activities, this allows us to model headway constraints by for example imposing that the times between departures is large enough. When properly implemented, this will also ensure that the headway constraints are respected at the arrivals of the trains. Finally, the synchronizing activities are used to impose synchronized departure times of trains belonging to the same line.

For the mathematical formulation of the PESP, we let $\pi_i \in [0, T-1]$ denote the decision variable representing the time instant assigned to node i. The periodic constraint for an activity $(i,j) \in \mathcal{A}$ is then given by

$$l_{ij} \leq \pi_j - \pi_i + Tp_{ij} \leq u_{ij}, \quad p_{ij} \in \{0,1\}, \tag{4.40}$$

where l_{ij} is the lower bound of the duration of activity (i,j) and u_{ij} the upper bound. The decision variable p_{ij} is introduced to compute the duration of the activity correctly when $\pi_j < \pi_i$. For instance, if a train arrives at a station at $t = 58$ and leaves the station at $t = 2$, the duration associated to the dwell arc between these events equals $2 - 58 + 60 = 4$ minutes. The p-variables are also referred to as the *modulo* parameters. When $u_{ij} \geq T$, the correct domain for the modulo parameters are the nonnegative integers. However, such a PESP can be transformed in an equivalent PESP where all $u_{ij} < T$.

For a more detailed exposition of the PESP and periodic railway timetabling we refer to Peeters (2003).

Station Capacity in the PESP

Since the PESP only contains constraints between pairs of events and the number of trains in a station at a certain time depends on all events at that station, it is not possible to formulate a station capacity constraint within the PESP framework. As demonstrated by Peeters (2003), it is however possible to formulate this constraint using the modulo parameters. To this end, we use the following interpretation of the modulo parameters:

$$p_{ij} = \begin{cases} 1, & \text{if event } j \text{ takes place before event } i, \\ 0, & \text{if event } j \text{ takes place after or at the same time as event } i. \end{cases} \tag{4.41}$$

Here, *before* or *after* refers to the sequence of the events on the linear axis $[0, T-1]$.

Now assume the event-activity network of a station $\mathcal{N}_s = (\mathcal{E}_s, \mathcal{A}_s)$ contains all arcs

$$(i, j), \text{ where } i \in \mathcal{E}_s^{\text{arr}} \text{ and } j \in \mathcal{E}_s^{\text{dep}}, \tag{4.42}$$

$$(i, j), \text{ where } i \in \mathcal{E}_s^{\text{arr}} \text{ and } j \in \mathcal{E}_s^{\text{arr}} \text{ and } i < j. \tag{4.43}$$

That is, the network contains all activities from all arrivals to all departures and from lower indexed arrivals to higher indexed arrivals. The reason it is not possible to formulate the station capacity constraint for an arbitrary network is that we need to know the sequence of all events. However, this poses no restriction since activities that are not included in the original network can be added with the trivial time window $[0, T-1]$.

We will now illustrate how the number of trains in a station can be counted using the modulo parameters using an example. In Figure 4.7 the arrival and departure times of four trains at a certain station are given and in Table 4.1 the values of the corresponding modulo parameters are presented.

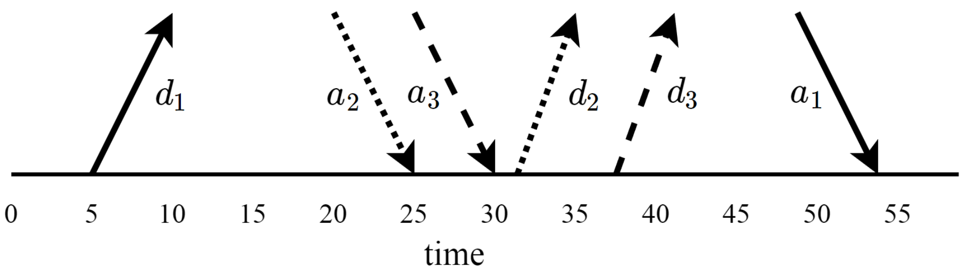


FIGURE 4.7: Example of arrival and departure times at a station.

TABLE 4.1: The modulo parameters corresponding to the arrival and departure times in Figure 4.7. The circled numbers are used in the example.

		Event j					
	p_{ij}	a_1	a_2	a_3	d_1	d_2	d_3
Event i	a_1	-	1	1	1	1	1
	a_2	-	-	0	1	0	0
	a_3	-	-	-	1	0	0

Suppose we are interested in the number of trains in the station right after the arrival of train 3. Intuitively, this can be done by starting with the number of trains at $t = 0$, increasing this number with all arrivals that take place before train 3, and decreasing this number with all departures before train 3. Finally, we need to add one to the result since train 3 has just arrived.

This can be computed using the modulo parameters as follows. The number of trains at the station at $t = 0$ is equal to the sum of all modulo parameters corresponding to dwell activities, $p_{a_1d_1} + p_{a_2d_2} + p_{a_3d_3}$. In the example only the dwell activity of the first train has a modulo parameter equal to one, so this sum equals 1. The number of trains arriving before train 3 equals $(1 - p_{a_1a_3}) + (1 - p_{a_2a_3})$ in terms of the modulo parameters. In the example, train 1 arrives after train 3 and that train 2 arrives before train 3, hence the sum equals 1. The number of trains departing before train 3 arrives can be derived from the row of a_3 in the table and is equal to $p_{a_3d_1} + p_{a_3d_2} + p_{a_3d_3}$, which is in this case 1.

We can conclude that the number of trains in the station when train 3 arrives equals

$$\begin{aligned}
&p_{a_1d_1} + p_{a_2d_2} + p_{a_3d_3} && \text{(trains dwelling at } t = 0\text{)} && (4.44)\\
&+ (1 - p_{a_1a_3}) + (1 - p_{a_2a_3}) && \text{(trains arriving before train 3)} && (4.45)\\
&- p_{a_3d_1} - p_{a_3d_2} - p_{a_3d_3} && \text{(trains departing before train 3)} && (4.46)\\
&+ 1 && \text{(train 3 itself).} && (4.47)
\end{aligned}$$

The station capacity constraint can be formulated by applying the same counting method as above. We let $\mathcal{A}_s^{\text{dwell}}$ denote the set of dwell activities at station s, let $\mathcal{A}_i^a$ denote the set of outgoing activities to arrivals from event i and let $\mathcal{A}_i^d$ denote the set of outgoing activities to departures from event i. Then, we can enforce that the capacity of station s is never violated by adding the following constraints:

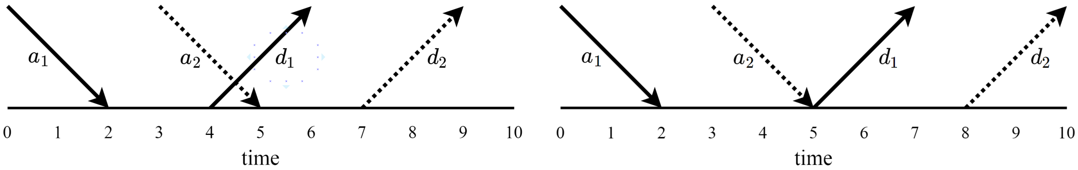
$$1 + \sum_{(k,l)\in\mathcal{A}_s^{\text{dwell}}} p_{kl} + \sum_{(j,i)\in\mathcal{A}_j^a} (1 - p_{ji}) + \sum_{(i,j)\in\mathcal{A}_i^a} p_{ij} - \sum_{(i,j)\in\mathcal{A}_i^d} p_{ij} \leq |P^s| \quad \forall i \in \mathcal{A}_s^{\text{arr}}. \tag{4.48}$$

Usually, the platforms at a station are subdivided into groups that are assigned to the different lines and directions. In such cases, the capacity constraint can be included for every group of platforms. We refer to a PESP extended with capacity constraints as a C-PESP.

Constraints (4.48) contain two slight inaccuracies. The first inaccuracy is that when events occur concurrently, this is not dealt with consistently. When two events are assigned the same time instant, the modulo parameter corresponding to this pair of events equals zero. This means that in the third and fifth term of the left hand side, trains arriving or departing at the same time as arrival i are counted as being in the station. However, the fourth term does not include concurrencies, so these arrivals are counted as being outside the station. This issue can be resolved for by introducing a second modulo parameter for every pair of events. We refer to Peeters (2003) for details.

The second inaccuracy of the capacity constraint has to do with the headway time between two trains that needs to be respected. Figure 4.8a depicts the arrival and departure times of two trains at a platform. It can be seen that the capacity constraint at this platform is not violated, since the second train arrives after the first train departs. However, assuming a headway time of 2 minutes, this solution violates the headway constraint between d_1 and a_2. This flaw in the capacity constraint can be solved using the fact that a train dwelling 2 minutes at a station actually consumes

4 minutes of platform time, whereas it only consumes 3 minutes of platform time according to the capacity constraint (recall that concurrencies are always counted). Hence, the headway time can be accounted for by increasing the dwell times with 1 minute (the headway time minus one) when solving the PESP. To compensate, the trip times need to be decreased with the same amount. In other words, we pretend that trains spend more time at a station and less time driving between stations. The actual departure times can be recovered from the solution by applying the reverse transformation.



(A) Before (B) After

FIGURE 4.8: Departure and arrival times of two trains at a platform before and after the transformation of the dwell arcs.

In Figure 4.8b the same situation as in Figure 4.8a is depicted, but now the dwell times are increased with 1 minute. In this case, the model will recognize that at time instant 5 there are two trains at the platform, violating the capacity restriction. Note that for groups of platforms that only contain a single platform, the headway constraint can also be accounted for by the inclusion of safety constraints between arrivals and departures. However, for larger groups of platforms, it is really necessary to transform the dwell and trip arcs.

Pre-processing

The downside of the outlined solution approach is that we are not guaranteed to find a line plan that has a feasible timetable, as it can be impossible to extend the partial timetables at the sub-stations to a feasible timetable for the whole network. However, we can increase the likelihood that this is possible by pre-processing the C-PESPs at the sub-stations before they are solved, using constraint propagation.

To illustrate this, consider the event activity network in Figure 4.9. In this example we have two trains and two stations. We assume both trains must use the same track between the two stations and the same platform at station B. As can be seen, both trains dwell 2 minutes at station A. On the other hand, at station B train t_1 dwells 5 minutes and train t_2 dwells 2 minutes. Train t_1 needs 4 minutes to travel from station A to B. Train t_2 is a bit faster and only takes 3 minutes. Furthermore, as both trains use the same platform at station B, the network also contains a headway activity between the departure of train t_2 and the arrival of train t_1 at station B (i.e. train t_2 must depart at least 9 minutes after and at least 2 minutes before train t_1 arrives).

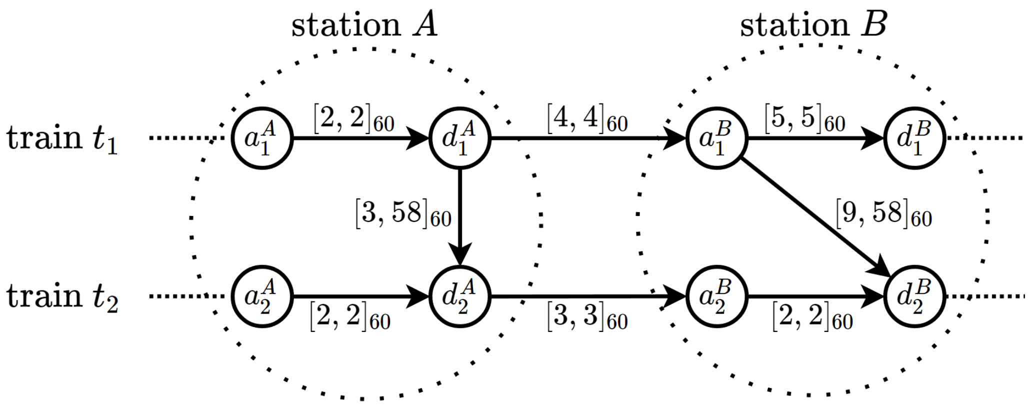


FIGURE 4.9: An event activity network with two trains and two stations.

We now show how the headway constraint between the two trains at station A can be strengthened. Before pre-processing, this constraint reads

$$d_2^A - d_1^A \in [3, 58]_{60}. \tag{4.49}$$

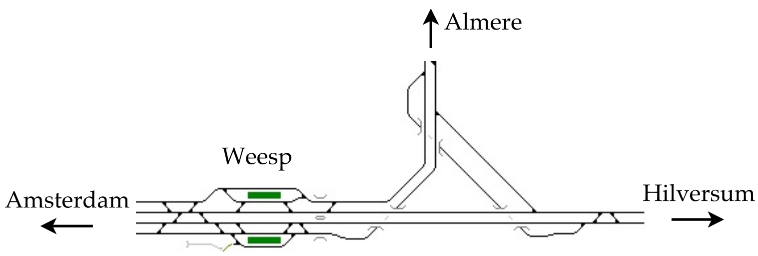
In words: train t_2 should depart at least 3 minutes after train t_1 and train t_1 should depart at least 2 minutes after train t_2. The reason that the constraint is asymmetric is that train t_2 takes one minute less to drive from A to B. Hence, by imposing it should depart at least three minutes after train t_1, the headway constraint just before entering station B is automatically satisfied.

We can strengthen this constraint by combining some of the constraints. The path $d_1^A \to a_1^B \to d_2^B \to a_2^B \to d_2^A$ implies a constraint between d_1^A and d_2^A with the time window $[4+9-2-3, 4+58-2-3]_{60} = [8, 57]_{60}$. As is clear, this path provides a tighter bound between d_1^A and d_2^A, hence we have now strengthened the constraint.

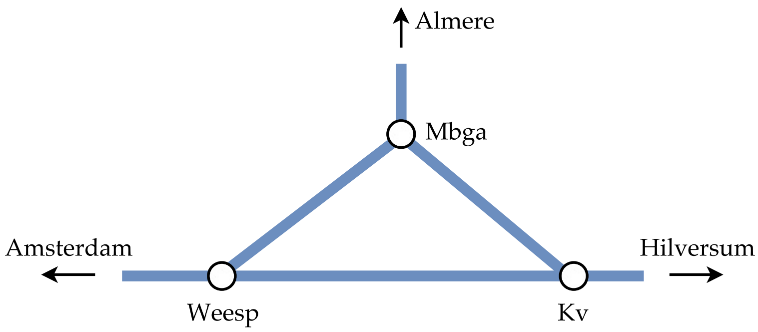
This procedure can be formalized by means of constraint propagation techniques. A description of how constraint propagation is used in CADANS, the periodic timetable solver of NS, is provided in Odijk, Lentink, and Steenbeek (2002). As is clear, by applying pre-processing, the C-PESPs at the sub-stations are stronger as they also take into account the constraints at other parts of the network. Therefore, it is more likely that solutions of the algorithm have a feasible timetable.

Implementation
In our implementation of the C-PESP, we include dwell constraints, trip time constraints, synchronizing constraints, headway constraints and capacity constraints. To make sure the paths of trains do not interfere with each other, dummy stations are created for junctions between stations and for points where the number of tracks changes. An example of this representation is visualized in Figure 4.10. The dummy station Mbga (Muiderberg aansluiting, the name used by NS and ProRail for this timetabling point) is added at the location where the track from Almere splits in a track to Weesp and a track to Hilversum. Similarly, the dummy station Kv (Keverdijk aansluiting) is added at the location where the track from Hilversum splits in a track

to Weesp and a track to Almere. Both dummy stations have two platforms. Doing so prevents trains from for example Weesp and Hilversum headed to Almere to interfere when the tracks come together at the junction.



(A) Railway infrastructure



(B) Representation used for timetabling.

FIGURE 4.10: The railway infrastructure around station Weesp and the corresponding representation used for timetabling.

As for the dwell constraints, the dwell time is allowed to vary between a minimum dwell time (dependent on whether the train is turning) and a maximum (dependent on whether the station is a terminal station or an in-between station). For the maximum we use the sum of the minimum dwell time and σ_l^s, which is the maximum allowed additional dwell time at a station (see constraints (4.23)). Then, both the minimum and maximum are increased with the headway time minus one to make the capacity constraint more accurate. At terminal stations, it is not known beforehand to which departing train the arriving train connects. In Figure 4.11, a part of the event activity network is visualized to illustrate this. This example considers a train with frequency two, resulting in two trains, t_1 and t_2, arriving at the terminal station s_2 and two trains, t_1' and t_2' departing the terminal station. The two types of dashed arrows illustrate the two possible turning patterns. Fortunately, it is possible to let the model select the turning pattern within the PESP framework, by including disjunctive constraints stating that t_1 and t_2 should turn on either t_1' or t_2'. These constraints can easily be transformed into regular PESP-constraints, see Peeters (2003).

In our implementation, we fix the trip times for all trains. The trip time depends on the type of train (regional or intercity) and is equal to the minimum possible driving time plus a margin of 5 percent to absorb small delays. The trip times are reduced with the headway time minus one, again for the sake of the capacity constraint.

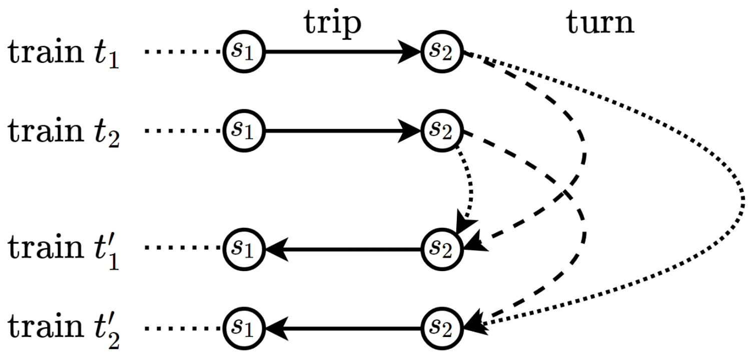


FIGURE 4.11: Part of the event-activity network corresponding to the turnings of a train with frequency 2. The two types of dashed arrows represent the two possible turning patterns.

Synchronizing constraints are only added between trains of the same line to reduce complexity. To limit the solution space, we impose perfect synchronization. This means that the departure times of a train with frequency f should be exactly T/f minutes apart.

We add headway constraints to avoid multiple type of conflicts. There should be sufficient time between departing trains such that the headway constraint is respected both at the departure at the station and the arrival at the next station. The required time between trains can be calculated using the headway time and the difference in trip time between trains. By doing so, these constraints also prevent faster trains to overtake slower trains where that is not allowed. If two trains must be assigned the same platform at a station (e.g. when there is only one platform), we also add a constraint stating that the arrival of one train should not be too close to the departure of the other train. For parts of the network with a single track, safety constraints are also added for trains going in opposite directions, to prevent that the trains meet each other on this track.

Before we add the station capacity constraint, we divide the platforms at every station in groups, based on the solution of the master problem. Two platforms belong to the same group if there exists at least one line for which the trains can be assigned to both platforms in one of its directions. Then, the capacity constraint is added for every group of platforms. Consider for example the station Utrecht Overvecht in Figure 4.12. If none of the lines in the line plan turns at this station, platform 1 is exclusively used for trains in the direction of Blauwkapel and platform 2 is exclusively used for trains in the direction of Utrecht Centraal. In such a case, the capacity constraint can be added for the two platforms separately. On the other hand, if the line plan contains a line that enters Overvecht from the side of Blauwkapel and then turns, the two platforms belong the same platform group as the turning trains can be assigned to both platform 1 and platform 2.

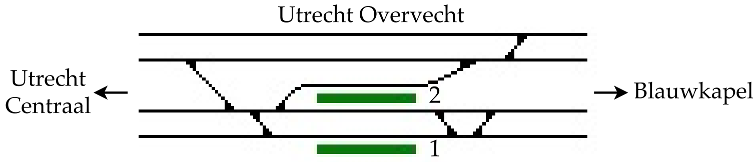


FIGURE 4.12: Lay-out of station Utrecht Overvecht (source: sporenplan.nl).

4.3.2 Cut Generation

In order to find inequalities that cut of a large part of the feasible region, instead of solving a large timetabling problem for the entire network, we solve multiple small timetabling problems for small parts of the network. When encountered with infeasibilities, this allows us to pinpoint exactly which combinations of lines lead to inconsistencies, resulting in stronger cuts than can be added to the master problem.

More specifically, in the slave problem we solve a timetabling problem for every *sub-station*, which is a group of platforms for which the arrival and departure times only occur in constraints with arrival and departure times of the same sub-station. That is, the C-PESP for a station, containing all dwell, synchronizing, safety and capacity constraints for that station, can be decomposed in the C-PESPs for its sub-stations.

The decomposition of a station into sub-stations depends on the solution of the master problem. Consider again the lay-out of station Utrecht Overvecht depicted in Figure 4.12. In case none of the lines in the solution contains a turn at this station, platforms 1 and 2 belong to different sub-stations, as the trains using platform 1 (in the direction Blauwkapel) do not share any constraints with trains using platform 2 (in the direction Utrecht Centraal). However, if one of the lines coming from Utrecht Centraal turns at Utrecht Overvecht, the C-PESP contains safety constraints ensuring that the turning trains departing from platform 1 do not interfere with the departing trains on platform 2. Hence, the platforms now belong to the same sub-station. Note that the capacity constraint can still be added for platforms 1 and 2 separately.

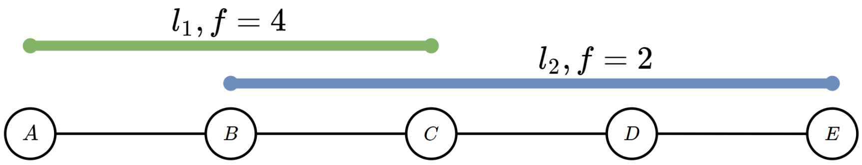
When we detect a station infeasibility at a sub-station, we first check whether the set of lines attending the sub-station is a minimal inconsistent set by iteratively removing the lines in the set and checking whether the C-PESP now has become feasible. If removing one of the lines still results in an infeasible C-PESP, the line is removed from the set. This process is repeated until the set of lines corresponds to a minimal inconsistency.

To give a mathematical formulation of the cut, we let $\mathscr{L}^s$ denote the minimal inconsistent set of lines attending sub-station s in the current solution of the master problem. Then, when the C-PESP of s does not have a feasible solution, the following cut can be added to the master:

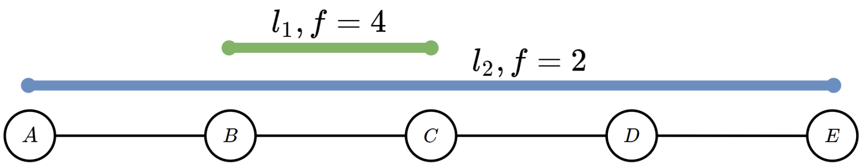
$$\sum_{l \in \mathscr{L}^s} x_l \leq |\mathscr{L}^s| - 1. \tag{4.50}$$

This cut rules out the combination of lines that attend sub-station s in the current solution. However, this is not the only cut we can derive from the discovered infeasibility, since we know all line plans that generate a C-PESP that is at least as difficult as the C-PESP at s do not have a feasible timetable.

To illustrate this, consider the line plan visualized in Figure 4.13a, containing the lines l_1 and l_2 with frequencies 4 and 2, respectively. Now assume this line plan leads to an infeasible C-PESP at station C. After adding the corresponding cut to the master and resolving, we might find the line plan in Figure 4.13b in the next iteration. However, this line plan must also result in an infeasibility at station C, as from the perspective of this station, the solution has not changed (assuming that the minimum dwell time only depends on whether trains turn or not). In other words, the C-PESPs generated for station C are equivalent. Therefore, we could have excluded both line plans upon finding the infeasibility at C.



(A) Solution of the master problem.



(B) Alternative line plan.

FIGURE 4.13: Two line plans that generate the same C-PESP at station C.

More generally, let $\mathcal{M}_i^s$ denote the set of lines that have the same frequency, the same in- and outbound edge and the same minimum dwell time at station s as the i'th line in $\mathscr{L}^s$. Then, from a station infeasibility at station s, we can derive the following cuts:

$$\sum_{l \in M} x_l \leq |\mathscr{L}^s| - 1, \quad \forall M \in \mathcal{M}_1^s \times \mathcal{M}_2^s \times \ldots \times \mathcal{M}_{|\mathscr{L}^s|}^s. \tag{4.51}$$

The advantage of adding more than one cut in every iteration is that it is likely to reduce the total number of iterations before all C-PESPs are feasible. On the other hand, the time spent solving the master problem might increase, as adding all cuts (4.51) increases the size of the master, especially when the disrupted region and/or the set $\mathscr{L}^s$ is large. To investigate whether the benefits of adding more than one cut outweighs the disadvantages, we test the algorithm with and without adding all cuts (4.51).

It must be noted that when multiple cuts are added with every detected infeasibility, the constraint propagation must be limited to some extent. When constraint propagation is applied, it is possible that an infeasibility at station s is caused by constraints at a different station, say station s' and not all line combinations that are ruled out by cuts (4.51) may generate these specific constraints (other lines may not even attend station s'). To avoid this scenario, we continue propagation from station s only up until decoupling stations. This way, the pre-processing for the original line combination generates the same constraints as the line combinations appearing in cuts (4.51).

Finally, in some cases it is possible to infer even more cuts from an infeasible C-PESP. For example, consider a sub-station with two lines that use the same platform and have the same minimum dwell time. If no feasible timetable exists for this sub-station, a logical implication is that replacing these lines by a single line with as frequency the sum of the frequency of the lines also results in an infeasible C-PESP. The intuition behind this is that due to synchronizing constraints, it is more difficult to find a timetable for a single line with frequency $f_1 + f_2$ than for two lines with frequencies f_1 and f_2. In preliminary experiments, adding these type of cuts did not lead to any improvements with respect to the number of iterations before termination. Therefore, we do not add these cuts in the implementation of the algorithm.

Chapter 5

Train Dispatching Strategies

In this chapter we propose several train dispatching strategies and describe how we evaluate the performance of combinations of line plans and such strategies. We start by discussing the local dispatching strategies. Next, we describe the simulation framework and how the simulation is initialized with the aim to mimic restarting the railway system in an out-of-control situation. Thereafter, we describe the mathematical model that is used to assign the rolling stock in the disrupted region to the lines. We finalize the chapter by outlining the measures that are used to quantify the performance of the line plans and dispatching strategies.

5.1 Strategies

In regular operations the trains are operated according to a timetable that specifies the exact departure and arrival times of trains and the routing through stations. Trains are dispatched according to the timetable and adjustments are made to the timetable in case of disturbances or disruption. Conversely, when operating a simplified line system in out-of-control situations, a timetable is not available. Therefore, radically different train dispatching strategies are required. These strategies should specify simple rules that determine when trains depart. Next to that, as out-of-control situations are characterized by a lack of complete and accurate information, the strategies should be *local*, meaning that only information of the directly surrounding part of the railway network is required to decide what to do next.

The train dispatching strategies that we develop determine what to do next when a train arrives at a station. More specifically, the strategies specify (i) *when* the arrived train will depart and (ii) *where to* the train will depart. The information that is allowed to be used to make these decisions are previous departure times at the station and information from trains directly surrounding the station.

As for the *when* aspect of strategies, we consider three options, referred to as FIFO, SYNC and SYNC+COOR. When trains are operated using the FIFO (first in first out) principle, trains that arrive in a station always leave as soon as possible. This may be reasonable, as we expect to have limited station capacity, hence trains might not occupy platforms longer than necessary. In the initialization, the trains are scheduled to depart as soon as possible.

When trains are operated using the SYNC (synchronize) principle, the departure time is decided in a more sophisticated manner. As the capacity is not limited at all stations, it is not always necessary to depart as soon as possible. Therefore, when a train arrives at a terminal station this principle determines the departure time based on the previous departure time of the line associated to the train in order to promote the regularity of the departure times. For example, if a line has frequency 4 and the previous train of the line departed 7 minutes ago, the train will depart in 8 minutes. In case the previous train departed 14 minutes ago, the train will still leave as soon as possible. Moreover, in the case a train is unable to enter a terminal station because of a waiting train, the waiting train will depart sooner than its desired departure time to make place for the entering train. Finally, when initializing using the SYNC principle, the trains are scheduled to depart as soon as possible unless there are multiple trains of the same line at a station. In such a case, departures are scheduled in a regular departure pattern as indicated by the frequency of the line.

The SYNC + COOR (synchronize and coordinate) principle is even more involved. This principle can be applied only when considering instances with both intercity and regional trains as it tries to coordinate the departure times of these trains. SYNC+ COOR extends SYNC in two ways. Firstly, the principle imposes that if a regional train has departed from a station on a part of the network that has two tracks, intercity trains can only depart when enough time has passed to make sure the faster intercity train does not have to wait for the slower regional train. The minimum in-between time can be computed using the difference in driving times until the first point where the intercity can overtake the regional train. The second rule is that at stations where overtaking is possible, regional trains wait at the station if an intercity train is coming within 3 minutes and the regional train would have otherwise blocked the incoming intercity train. Note that it is also possible to take a different value than 3 minutes or let the maximum waiting time depend on the decrease in travel time of the intercity train.

As for the *where to* aspect of dispatching strategies, we consider two options, STAT and DYN. In the STAT (static) principle, the line assignment that is determined when the system is starting up remains fixed at all times. In the DYN principle, trains can be reassigned when they reach their terminal station. Trains are reassigned based on the type of the train and the previous departure times; the line that needs a departure the earliest gets assigned the first compatible train. The advantage of the DYN principle is that it results in shorter turning times at the terminal stations. Even more, this principle leads to more efficient use of the trains, such that it is possible to operate more trains per hour.

Clearly, the STAT and DYN principles are related to the respectively fixed and flexible rolling stock constraints discussed in Section 4.2.5. However, even if fixed rolling stock circulations are imposed in the line planning algorithm, it might be beneficial to use the DYN principle, as it leads to more flexible operations.

The three principles for the *when* aspect and two principles for the *where to* aspect give rise to six different strategies, with ranging degrees of coordination. For instances without intercity trains we have only four strategies, as the SYNC+COOR principle is only different from the SYNC principle in the way it deals with intercity trains.

5.2 Simulation Framework

We evaluate the line plans by simulating the railway traffic in the disrupted region. For this simulation, we use a macroscopic representation of the railway network where nodes are stations and edges are tracks. Junctions are modeled as dummy stations with the number of tracks as the number of platforms in the same way as described in Section 4.3.1. In the actual operations, tracks are subdivided into blocks and a train is allowed to enter a block if the previous train is no longer occupying the block. Otherwise, the train needs to wait before the red sign placed at the beginning of the block. In the simulation, the blocks are not taken into account and trains therefore only wait upon arriving at or departing from a (dummy) station.

Trains can enter a station if there is a platform available that is compatible with the in- and outgoing track of the train. A platform is available if it has not been used by a train in the last 2 minutes. When confronted with multiple compatible platforms, the simulation picks one at random. If there is no available platform, the train is added to the arrival queue at the station. It is assumed this queue has infinite capacity. Whenever a train departs from a station, the simulation checks whether the newly available platform can be used by one of the trains in the arrival queue and a arrival is scheduled if that is the case.

Dwell times are set at fixed numbers in the simulation. We use 5 minutes if the train turns and 2 minutes otherwise, the same values that are used in the line planning model of Chapter 4. At a dummy station, the minimum dwell time is of course 0 minutes. The minimum dwell times are taken into account when computing the minimum travel time of a line.

Trains can depart from a station if the outgoing track is available for the type of the train. If there are four tracks, regional and intercity trains run on separate tracks. The outgoing track is available if a certain time has passed since the last train of the same type has departed. Usually, this time is again equal to the headway time of 2 minutes. However, if there is only a single platform available at the next station, the minimum time between trains is increased with the minimum dwell time at the next station of the train. Otherwise, we are certain that a train departing only 2 minutes after the previous train will encounter a red signal at the next station.

On single-tracked parts of the railway network, usually one or more stations have two platforms that allow trains going in opposite directions to pass each other. To prevent two trains in opposite directions from meeting each other on a single track we impose that trains can only depart from such stations if another train is present on the other platform.

For the travel time between two stations we use the travel time as specified in the timetable of NS. Moreover, between stations we assume that trains are running at a constant speed. Modeling acceleration and brake curves is outside the scope of this thesis. This implies that trains speed up instantaneously in the simulation.

5.2.1 Initialization

The goal of the simulation is to mimic railway traffic in an actual out-of-control situation. Hence, the simulation takes as input a disrupted region and the time instant control is lost over the system. Using these inputs we can retrieve the current position of the trains in the disrupted region from the regular timetable of NS. We assume that all trains driving when the snapshot is taken first arrive at the next station before the system restarts. Next to the locations, we also retrieve the in- and outbound tracks of trains from the timetable. In this way, we can also determine what the feasible outbound tracks are for every train from their current position.

When the current positions and possible outbound tracks of the trains are retrieved, the trains are assigned to lines in the adapted line plan. Simultaneously, the trains are also assigned to outbound tracks to determine the direction the trains depart initially. In the next section it is explained how we perform this assignment.

Once the trains are assigned, departure times are scheduled for the trains, based on the train dispatching strategy that is applied. After the departure times are scheduled, the simulation commences. Trains that are not assigned to a line are not taken into account in the simulation. In other words, it is assumed that these trains can be routed to a shunt yard without interfering with the other trains.

5.2.2 Line-Direction Assignment

We use a mathematical model to assign the trains in the disrupted regions to lines and to a direction the train will depart in. This model tries to capture the decision making process in out-of-control situations, and is designed to assign the trains in such a way that trains assigned to the same line are sufficiently spread out over the network. We now present a mathematical formulation of the model used for this assignment.

As in Section 4.2.5, R denotes the set of trains located in the disrupted region. An *assignment* a is a tuple (l, e) representing the line the train is assigned to, denoted by l_a and the departing edge (track), denoted by e_a. The parameter b_{ra} indicates whether assignment a is allowed for train r. Such an assignment is possible if (i) the type of train r matches the type of line l_a and (ii) train r can depart on edge e_a from the current position. The parameter n_l denotes the number of trains required to operate line l under fixed circulations. The decision variables are v_{ra}, indicating whether train r is assigned to line-edge a, and z_l, representing the shortage in the number of trains assigned to line l. Next to that, we have the combination variables q_{rapb}, indicating whether assignment a is used for train r and assignment b is used for train p. The assignment model can then be formulated as follows:

$$
\begin{aligned}
\text{minimize} \quad & C \sum_{l \in \mathcal{L}} z_l + \sum_{r,p \in R} \sum_{a,b \in A} w_{rpab} q_{rpab} && && (5.1)\\
\text{s.t.} \quad & \sum_{r \in R} \sum_{a \in A | l_a = l} b_{ra} v_{ra} + z_l = n_l && \forall l \in \mathcal{L}, && (5.2)\\
& \sum_{a \in A} v_{ra} \leq 1 && \forall r \in R, && (5.3)\\
& v_{ra} + v_{pb} - 1 \leq q_{rpab} && \forall r,p \in R, \quad \forall a,b \in A, && (5.4)\\
& v_{ra}, q_{rpab} \in \{0,1\} && \forall r,p \in R, \quad \forall a,b \in A, && (5.5)\\
& z_l \geq 0 \text{ and integer }, && \forall l \in \mathcal{L}. && (5.6)
\end{aligned}
$$

The objective is to minimize the total shortage and the number of unattractive assignment combinations. For C we choose a high number such that this becomes the primary objective. We describe how the weights of the combinations are defined in the next paragraph. Constraints (5.2) assure that for every line l the sum of the assigned trains and the shortage equals n_l trains, the minimum number required for the line under fixed circulations. If rolling stock constraints with fixed circulations are included in the line planning algorithm, it is certain that all lines can be assigned n_l trains. Otherwise, some lines may have a deficit. Constraints (5.3) impose that a train can only be assigned once and Constraints (5.4) make sure the q-variables obtain their correct values.

The combination weights w_{rpab} are defined in order to avoid assigning trains in close proximity to each other to the same line. This leads to smoother resumption of operations, since the trains should be operated in a synchronized pattern, with approximately equal distances between trains of the same line. To make this rigorous, we define the 'spread' between two assignments (r, a) and (p, b), where $l_a = l_b$ as

$$
s_{rpab} = \frac{\min\{t_{re_a \to p}, t_{pe_b \to r}\}}{T/f_{l_a}}, \tag{5.7}
$$

where $t_{re \to p}$ denotes the time it would take train r to reach the position of train p by departing on edge e. Note that the spread can be larger than 1. For example, if two trains that are 20 minutes apart and are both assigned to a line with frequency 4, their spread is 20/15=4/3.

Now, the weight of a combination of two assignments is given by

$$w_{rpab} = \begin{cases} (1 - s_{rpab})(1 + 1/f_{l_a}), & \text{if } l_a = l_b \text{ and } s_{rpab} < 1, \\ 0, & \text{else.} \end{cases} \tag{5.8}$$

The penalty of a combination of assignments is 0 if trains are assigned to different lines or when the spread is sufficiently large. Otherwise, the penalty decreases linearly in the spread and in the frequency. The reason for this is that when two trains have a spread of 0 (the trains are at the same station), it is better to assign them both to a line with a high frequency than to a line with a low frequency.

5.3 Evaluation Criteria

The simulation is used to assess the performance of the line planning algorithm and train dispatching strategies. To provide a complete image of the performance, we use several evaluation criteria. Combined, these criteria help answer the question whether the line plans and dispatching strategies constitute feasible and smooth operations in out-of-control situations.

The evaluation criteria that we analyze are the (i) frequency, (ii) regularity and (iii) delays. As for the first criterion, we are interested in whether the realized frequencies of the lines in the line plan resemble the desired frequencies. The second criterion regards the regularity of departure times of lines. For example, a line with a frequency of 4 should ideally depart every 15 minutes in both directions. The third criterion concerns the delays, which should of course not be too large.

In order to quantify the evaluation criteria we have defined three measures. The measures are defined for the operation of a line in a certain direction and can be computed at every departure at the associated terminal station. Between departures, the measures can be interpolated. The measures are defined in such a way that when a train line is operated perfectly (i.e. trip times are at their minimum and the trains depart from the terminal stations according to a perfect synchronized pattern), the train line scores exactly 1 for all measures at all times. This allows us to clearly observe deviations from an ideal scenario.

For the sake of notation, we start counting the departures from zero. The time of the i'th departure is denoted as t_i. Furthermore, we let p equal T/f_l, the period of a line.

Frequency. At the i'th departure of a line from a certain terminal station, the frequency measure is equal to the normalized frequency:

$$f_{\text{meas}} = \frac{i \times p}{t_i - t_0}. \tag{5.9}$$

This measure relates the desired time between the first and i'th departure to the realized time between the first and i'th departure. If there are fewer departures than desired, the measure is below one, and if there are more departures than desired, the measure is above one. If f_{norm} equals x in the long run, this indicates that there are $(x-1) \times 100$ percent more departures compared to an ideal scenario.

Regularity. The regularity measure is defined as follows:

$$r_{\text{meas}} = 1 - \frac{\sum_{j=1}^{i} |t_j - t_{j-1} - p|/p}{i}. \tag{5.10}$$

The measure relates the cumulated relative deviation from the optimal departure pattern to the number of departures. A long run value of x indicates that on average, there is an absolute deviation of $(1-x)p$ minutes between two departures. Since the cumulated deviation is nonnegative r_{meas} cannot be larger than one. Relative deviations are considered since a deviation of 10 minutes on a line with frequency 2 should be penalized less than a deviation of 10 minutes on a line with frequency 4.

Delays. For this measure we let t_{min} denote the minimum trip time and let d_i denote the delay of the i'th trip. The delay measure is given by:

$$d_{\text{meas}} = \frac{\sum_{j=0}^{i} t_{\text{min}} + d_i}{(i+1) \times t_{\text{min}}}. \tag{5.11}$$

This measure relates the cumulated trip time to the minimum cumulated trip time. Evidently, this measure cannot have a value below one. If the long run value is x, this indicates that the average delay is $(x-1)t_{\text{min}}$ minutes.

To evaluate the performance of a line, we take the average of the measures in both directions the line is operated in. For the performance of an entire line plan, we take the average of all lines. Doing so can result in a frequency measure of 1.00, even though the performance of the individual lines may be very bad, for example one line has frequency 1.5 and another line 0.5. However, in such a case we are still able to detect that the line plan has poor performance as both lines will have very bad scores on the regularity measure. Hence, it is justified to take the average.

Chapter 6

Computational Results

In this chapter, we present the computational results of the thesis. In Section 6.1, we describe the instances and settings that are used to test the algorithms. In Section 6.2, we discuss the results of the line planning model. In Section 6.3, we present the results of the simulations. The key findings are summarized in Section 6.4.

6.1 Problem Instances

We test the developed algorithms on the same instances that are used in Schouten (2017). The original line plan of the network that is considered is presented in Figure 6.1. In both instances the railway system is in a state of (near) out-of-control in a part of the depicted network. As can be seen, NS operates a dense line system in this part of the Netherlands, with six intercity lines and nine regional lines. Although the line connecting Utrecht Centraal, Hilversum and Almere does not dwell at all in between stations, it is still considered a regional line. Figure 6.1 also indicates which stations serve as decoupling stations, as specified by NS and ProRail.

The largest part of the considered railway network is double-tracked. To accommodate for higher frequencies of intercity and regional trains, there are four tracks between Utrecht Centraal and Amsterdam-Zuid and between Utrecht Centraal and Utrecht Overvecht. Furthermore, intercity trains can overtake regional trains at Amsterdam Muiderpoort, Weesp and Naarden-Bussum. Between Baarn and Den Dolder the network is single-tracked. Trains can pass each other at Soest.

For the passenger data we use an origin destination matrix with the average daily number of passengers between stations provided by NS. As the network that we consider is only a part of the Dutch railway network, there are many passengers that travel through the considered network without having both the origin and destination in this network. This is taken into account by including the passenger counts from and to major intercity stations outside the considered network (Amsterdam Sloterdijk, Schiphol Airport, Gouda, 's-Hertogenbosch, Eindhoven, Arnhem Centraal, Amersfoort and Lelystad Centrum). This suffices, as these stations account for

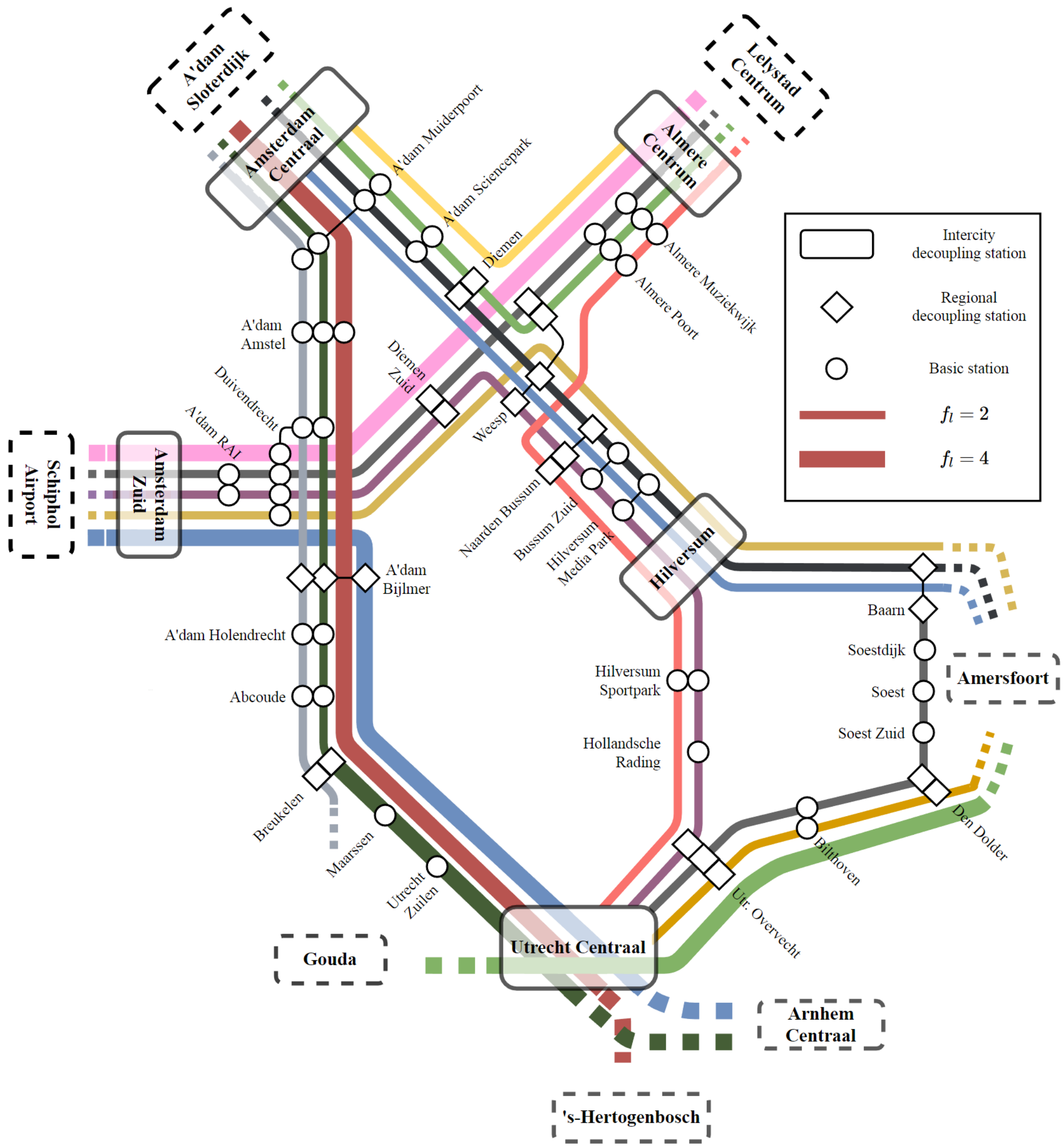


FIGURE 6.1: The regular line plan in a part of the Dutch railway network. The thickness indicates the frequency of the lines.

the majority of the neglected passengers. Even more, the algorithm only uses relative passenger counts, which are hardly affected if other stations are also included.

6.1.1 Small Disrupted Region

The first instance is a subregion of the network depicted in Figure 6.1, namely the region bounded by Utrecht Centraal, Den Dolder, Baarn and Hilversum. An out-of-control situation in this region could occur due to a power outage at Amersfoort, directly impacting five of the eight lines in this region. We assume buses are transporting passengers from Baarn and Den Dolder to Amersfoort, and vice versa.

The emergency scenarios of NS and ProRail can be used to determine the platform availability at Hilversum station, as this station also serves trains coming from the north. According to these scenarios, four of the five platforms at Hilversum are used to serve trains from the north. Hence, only a single platform is available to serve the disrupted region. Utrecht Centraal also serves trains from multiple directions, but this station has four platforms that are exclusively used for the direction Utrecht Overvecht. Therefore, we use four available platforms at Utrecht Centraal. At all other stations all platforms are available. An interesting aspect of this disrupted region is that while the most important lines (in terms of the number of passengers) in this region are the intercity lines between Hilversum and Amersfoort and between Utrecht Centraal and Amersfoort, it is not possible to operate any intercity trains in the disrupted region. The reason for this is that Baarn and Den Dolder are regional decoupling stations, such that intercity trains cannot turn at these stations. Furthermore, it is not allowed to operate an intercity line between Utrecht Centraal and Hilversum because there is no such line in the regular line plan. Hence, this poses the challenge to optimally use the available regional trains in order to also serve the passengers that normally take intercity trains.

6.1.2 Large Disrupted Region

The large instance that is considered is the entire network depicted in Figure 6.1, the disrupted region bounded by Amsterdam Centraal, Amsterdam Zuid, Almere Centrum, Utrecht Centraal, Den Dolder, Baarn and Hilversum. An out-of-control situation in such a large part of the Dutch railway network could occur after a combination of major disruptions at Amsterdam and Utrecht. In this instance it is possible to plan both regional and intercity lines.

In this instance, the emergency scenarios can only be used to determine the platform availability at Almere Centrum. At this station, two out of four platforms are available to serve the disrupted region. At Amsterdam Centraal, Amsterdam Zuid and Utrecht Centraal there are no scenarios specified that fit to the disrupted region. For example, there is no emergency scenario that specifies what to do at Amsterdam Centraal when both the track in the direction of Amsterdam Sciencepark and in the direction of Amsterdam Amstel is blocked. At Amsterdam Centraal we therefore assume that four of eleven platforms are available to serve the disrupted region, a rather conservative number considering the regular line plan. At Amsterdam Zuid and Utrecht Centraal the railway infrastructure makes it impossible for trains to turn directly at the arrival platform. At these stations, it is required to perform shunting movements in order to turn the trains. As shunting movements require some time and shunting train drivers, there cannot be too many trains per hour turning at these stations. To incorporate this in the model without explicitly modeling shunting movements, we include only a single available platform to serve the disrupted region at Amsterdam Zuid and Utrecht Centraal. Given that a large number of trains attend these stations in the regular line plan, the limited station capacity most likely is one of the bottlenecks in the large instance.

6.1.3 Parameter Settings

In Table 6.1 we provide an overview of the used parameters. We normalize the objectives such that we can use the same weight parameter for both objectives. After normalization it holds that if all trains are canceled, $\text{Obj}_{\text{edge}} = \text{Obj}_{\text{OD}} = 1$. For the small instance, the weight or cost of a line is set equal to $w = 0.01$. For the large instance, we set $w = 0.001$. For the period of the timetable we use $T = 60$ minutes, as in the regular timetable. Using a smaller value of T will result in too conservative line plans. For the dwell times, we use 2 minutes when a station continues and 5 minutes when a train turns, as in Schouten (2017). If we impose the dwell constraint, trains may only dwell longer than the minimum at the terminal stations. In other words, we impose the constraint in its most stringent form.

TABLE 6.1: Overview of the used parameters.

Parameter	Description	Value
w	'Cost' of a line	0.01 or 0.001
T	Period of the timetable	60 minutes
τ^H	Headway time	2 minutes
τ^C	Dwell time if train continues	2 minutes
τ^T	Dwell time if train turns	5 minutes
τ_l^s	Blocking time of line l at station s	$\begin{cases} \tau^H + \tau^C, & \text{if train continues} \\ \tau^H + \tau^T, & \text{if train turns} \end{cases}$
σ_l^s	Allowed additional dwell time of line l at station s	$\begin{cases} 60 \text{ minutes, if train is at terminal} \\ 0 \text{ minutes, else} \end{cases}$

6.1.4 Model and Algorithmic Settings

For a thorough analysis of the performance of the line planning algorithm, we perform tests on both instances with various settings of the algorithm. We differentiate between model settings, which influence the obtained solution and algorithmic settings, which only influence the *path* towards the solution. We define the following five model settings:

1. *Basic*: neither dwell constraints nor rolling stock constraints are included
2. *Dwell*: only dwell constraints are included
3. *Fixed*: only rolling stock constraints for fixed circulations are included
4. *Dwell+Fixed*: the dwell constraints and the rolling stock constraints for fixed circulations are included

5. *Flexible*, only rolling stock constraints for flexible circulations are included

As can be seen, these settings specify which constraints are included in the master problem of the line planning algorithm. Since adding constraints can only worsen the objective value, the follow relations hold:

$$\text{Obj}_{Basic} \leq \text{Obj}_{Flexible} \leq \text{Obj}_{Fixed} \leq \text{Obj}_{Dwell+Fixed}, \tag{6.1}$$

$$\text{Obj}_{Basic} \leq \text{Obj}_{Dwell} \leq \text{Obj}_{Dwell+Fixed}. \tag{6.2}$$

A final model setting is the objective that is used in the optimization. Clearly, different objective functions result in different optima.

Algorithmic settings influence the number of iterations required to find the solution and the computation time in general. These settings include (i) whether capacity is considered at station level or at platform level in the master problem and (ii) whether the slave problem produces one or multiple cuts per detected infeasibility. The level at which capacity is included in the master problem does not affect the final solution because the algorithm only terminates when all sub-stations have a feasible timetable. Since we included a capacity constraint in the slave problem, a sub-station can only have a feasible timetable if there is a platform assignment that does not violate the capacity constraint at platform level. Hence, considering capacity at platform level might reduce the number of necessary iterations before terminating, but does not affect the final solution. With the setting *multiple*, we indicate that all cuts (4.51) are added whenever a sub-station does not have a feasible timetable. Adding one or multiple cuts per infeasibility only affects the number of iterations and computation time since all added cuts are valid inequalities that only cut off infeasible solutions.

6.2 Line Planning Results

The line planning algorithm is implemented in Java on a Dell Precision Tower 5810 desktop running Windows 8 with an Intel Xeon E5-1650 v4 processor and 16 GB of RAM. We use CPLEX to solve both the master and slave problems to optimality. In this section we present the results of the algorithm for the different settings. We start by discussing the algorithmic performance of the algorithm. Subsequently, the line plans produced by the algorithm are analyzed for the two problem instances.

6.2.1 Algorithmic Performance

In Table 6.2 the number of iterations, cuts and computation times obtained with the different combinations of model and algorithmic settings are presented for both the small and large problem instance.

Impact of Algorithmic Settings

Whether one or multiple cuts are added with every detected infeasibility clearly has the largest effect on the number of iterations and the total computation time, which can mostly be attributed to the slave problems. Adding multiple cuts instead of one reduces the number of iterations up to a factor of two. This has an even larger effect on the time spent on the slave problems, as fewer infeasible C-PESPs are encountered and such infeasible C-PESPs are computationally expensive as the entire search tree has to be exhausted. Furthermore, while the master problem increases in size by adding more cuts, this cannot be observed to have an effect on the solving time of the master problem.

The level at which the capacity is considered in the master problem also has a clear impact on the performance of the algorithm, although its effect is less strong. It can be seen that considering capacity at platform level reduces the number of iterations before termination, the number of added cuts and the total computation time. In some cases, the computation time of the master problem can be seen to increase as platform level capacity is more complex, but the reduction in computation time of the slave problems almost always offsets this increase.

Impact of Model Settings

For the small disrupted region, the *Basic* and *Basic+Dwell* have the largest number of iterations, cuts and overall computation times. The number of iterations and cuts is smaller when rolling stock constraints are taken into account, which can be explained by the fact that including these constraints leads to less dense line plans, hence the infrastructure is less of a limiting factor and the algorithm is confronted with fewer infeasible C-PESPs. Next to that, the *Flexible* setting has a notably longer computation time (0.9s to 1.4s) of the master, especially considering that the algorithm found a feasible solution in a single iteration. The computation times for the different objectives are comparable.

For the large instance, computation times range from about 30 seconds to 8 minutes. The number of iterations for the *Basic* and *Basic+Dwell* settings are of the same size as for the other settings. The *Basic* setting spends the shortest time on the master problem and the *Flexible* setting the longest. This indicates that incorporating flexible circulations in the master problem strongly increases the complexity. However, if the algorithm adds multiple cuts per iteration and considers capacity at platform level, the total computation times are still manageable, 4 minutes and 1.5 minutes with the edge and OD objective, respectively. Incorporating the dwell time constraint in the master problem also increases the computation time spent on the master problem, but the increase is far less drastic than allowing the flexible circulations. As with the small instance, the objectives result in similar computation times.

TABLE 6.2: Number of iterations, cuts and computation times with the different model and algorithmic settings.

(A) Small instance with edge objective

Setting	Cuts per Infeasibility	Iterations Station	Iterations Platform	Cuts Station	Cuts Platform	Master CPU (s) Station	Master CPU (s) Platform	Total CPU (s) Station	Total CPU (s) Platform
Basic	One	8	6	23	18	0.9	0.2	25.7	24.2
	Multiple	4	4	60	60	0.2	0.1	4.0	6.0
Dwell	One	9	6	25	18	1.1	1.3	22.8	20.7
	Multiple	4	4	60	60	0.4	0.7	6.0	6.9
Fixed	One	3	3	2	2	0.1	0.1	0.6	1.2
	Multiple	2	2	2	2	0.0	0.1	0.4	0.7
Dwell+Fixed	One	3	3	2	2	0.3	0.4	1.4	1.5
	Multiple	2	2	2	2	0.2	0.2	0.8	0.8
Flexible	One	1	1	0	0	1.4	1.0	1.9	1.3
	Multiple	1	1	0	0	1.4	0.9	1.8	1.2

(B) Small instance with OD objective

Setting	Cuts per Infeasibility	Iterations Station	Iterations Platform	Cuts Station	Cuts Platform	Master CPU (s) Station	Master CPU (s) Platform	Total CPU (s) Station	Total CPU (s) Platform
Basic	One	9	7	23	19	1.0	0.5	23.0	19.2
	Multiple	4	4	60	60	0.1	0.1	6.3	5.9
Dwell	One	9	6	24	17	1.2	1.4	22.0	14.9
	Multiple	4	4	60	60	0.5	0.8	6.8	7.5
Fixed	One	2	2	1	1	0.1	0.1	0.9	1.2
	Multiple	2	2	2	2	0.1	0.1	0.8	1.1
Dwell+Fixed	One	2	2	1	1	0.3	0.4	1.1	1.4
	Multiple	2	2	2	2	0.2	0.4	1.3	1.4
Flexible	One	1	1	0	0	1.6	1.7	2.1	2.1
	Multiple	1	1	0	0	1.6	1.7	2.1	2.2

(C) Large instance with edge objective

Setting	Cuts per Infeasibility	Iterations Station	Iterations Platform	Cuts Station	Cuts Platform	Master CPU (s) Station	Master CPU (s) Platform	Total CPU (s) Station	Total CPU (s) Platform
Basic	One	8	5	50	35	0.7	0.9	118.4	73.4
	Multiple	4	3	153	105	0.3	0.5	54.8	39.3
Dwell	One	5	3	46	15	2.6	6.1	72.3	51.0
	Multiple	4	3	117	122	1.6	7.3	56.3	48.2
Fixed	One	6	4	48	31	2.0	1.5	62.3	44.1
	Multiple	5	4	316	294	1.4	1.6	51.8	40.4
Dwell+Fixed	One	7	4	57	22	9.0	10.8	111.3	54.5
	Multiple	5	3	316	286	5.5	6.9	58.3	40.4
Flexible	One	10	7	64	40	237.0	179.9	471.0	369.7
	Multiple	6	5	5087	5065	141.1	124.5	301.6	252.7

(D) Large instance with OD objective

Setting	Cuts per Infeasibility	Iterations Station	Iterations Platform	Cuts Station	Cuts Platform	Master CPU (s) Station	Master CPU (s) Platform	Total CPU (s) Station	Total CPU (s) Platform
Basic	One	6	5	36	24	1.3	1.0	91.8	78.3
	Multiple	3	3	140	105	0.4	0.5	42.1	41.4
Dwell	One	5	4	33	22	4.2	10.4	75.4	69.4
	Multiple	3	3	169	105	2.0	7.4	46.1	49.4
Fixed	One	6	3	38	22	3.5	2.1	68.8	33.9
	Multiple	4	3	148	130	2.4	1.9	46.9	32.6
Dwell+Fixed	One	6	3	38	13	9.7	10.7	79.0	44.7
	Multiple	4	2	148	121	6.7	7.3	50.7	29.0
Flexible	One	6	3	38	22	262.3	93.9	338.9	130.6
	Multiple	4	2	148	130	129.8	72.5	178.4	95.8

Finally, the number of generated cuts can be seen to be equal for some model settings. The explanation for this phenomenon is that the algorithm often adds the same cuts for the different settings, for instance to rule out 2-3 or 3-4 frequency combinations at certain stations. The number of cuts generated with the *Flexible* setting with the edge objective is noticeably larger if multiple cuts are added every infeasibility. This is caused by an inconsistent combination of five lines, whereas in most cases the minimal inconsistent combination only consist of two or three lines. As a consequence the set of all infeasible line combinations that can be inferred is the Cartesian product of five sets, resulting in a large number of cuts.

6.2.2 Small Disrupted Region

In Table 6.3 we present the values of the edge objective, OD objective, the number of lines and the number of required trains under fixed circulations of the optimal solutions for the different model settings of the small instance. The obtained line plans are visualized in Figures 6.2a-6.2f. If multiple settings result in the same line plan, this is indicated in the caption.

TABLE 6.3: Characteristics of solutions for the small instance. The column 'Trains' indicates the required number of trains to operate the line plan if all circulations are fixed.

(A) Small instance with edge objective

Setting	Edge obj.	OD obj.	Lines	Trains
Basic	0.013	0.029	5	13
Dwell	0.013	0.029	5	13
Fixed	0.137	0.262	4	8
Dwell + Fixed	0.137	0.262	4	8
Flexible	0.101	0.174	4	10

(B) Small instance with OD objective

Setting	Edge obj.	OD obj.	Lines	Trains
Basic	0.013	0.029	5	13
Dwell	0.013	0.029	5	13
Fixed	0.221	0.223	4	8
Dwell + Fixed	0.221	0.223	4	8
Flexible	0.169	0.161	5	10

Impact of the Dwell Constraint
As can be seen in the table and the figures, the dwell constraint does not have any effect on the obtained line plans. This implies that the solutions of the algorithm without the dwell constraints already satisfy this constraint. This can be explained by the fact that only at Hilversum there are fewer platforms available than in the regular line plan. As a consequence, every line has at least one terminal station with sufficient platform capacity, such that the additional dwell time can be spend there.

Impact of Rolling Stock Constraints
As expected, the objective values worsen when rolling stock constraints are taken into account; the solution without rolling stock constraints actually requires 13 trains,

whereas there are only 8 trains available. We can also observe clear benefits in terms of objective values when comparing fixed and flexible circulations. Furthermore, the *Flexible* setting finds rolling stock circulations that can operate line plans that would need 10 trains if fixed circulations were used. That is, by using flexible circulations 2 trains are saved compared to a rolling stock schedule with fixed circulations, ultimately resulting in more dense line plans.

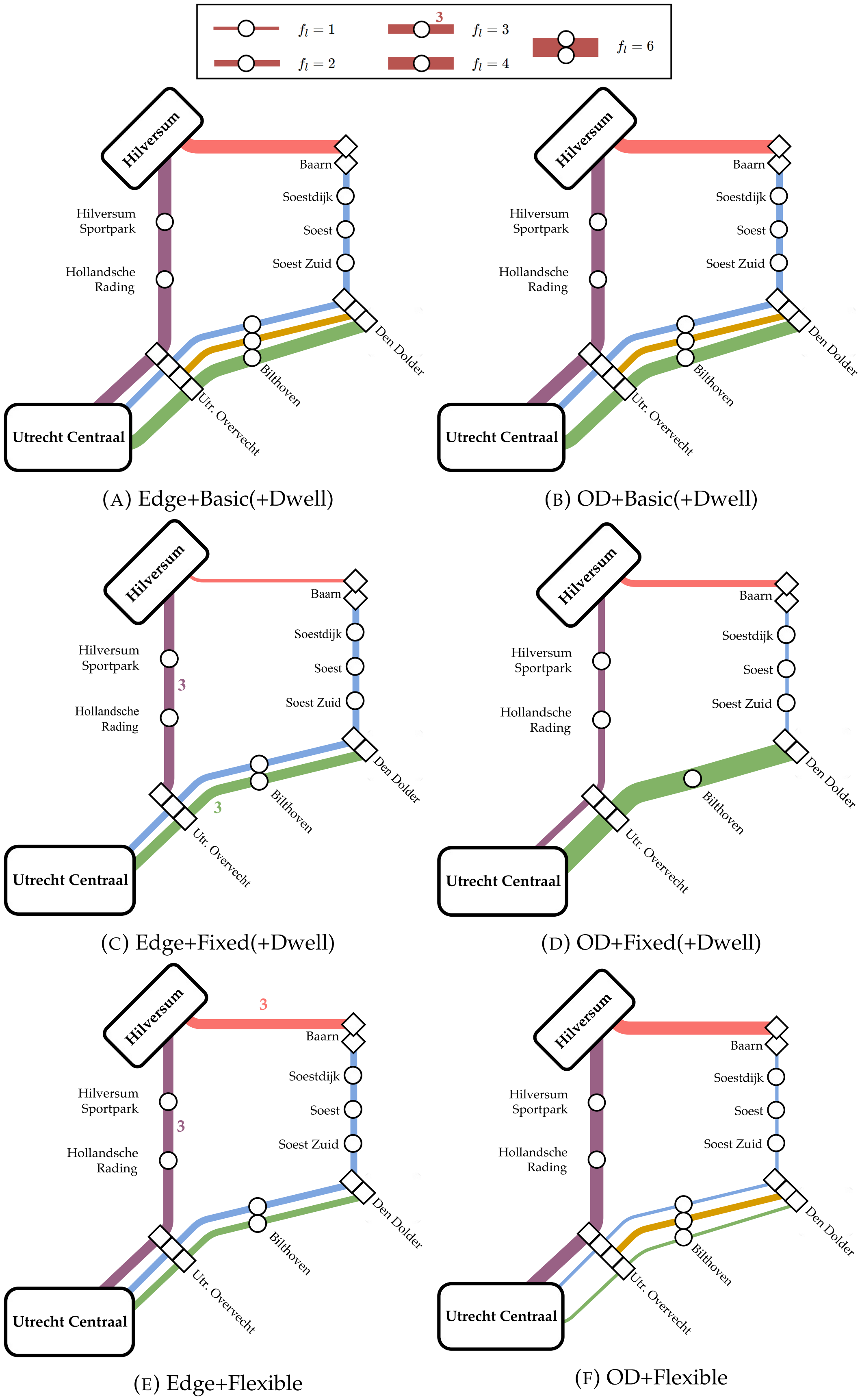
To further illustrate the benefits of flexible circulations, consider the line plan obtained with the edge objective and the *Flexible* setting in Figure 6.2e. In Table 6.4 some characteristics of the lines in this solution are presented. It can be observed that if fixed circulations are used, trains would have long dwell times between round trips. For example, a round trip Hvs-Brn takes 22 minutes, but as this line has a frequency of 3, the time between two departures of a train in the same direction actually is 40 minutes. To operate this line plan with only 8 trains, the algorithm selected two circulations. The first circulation contains the lines Ut-Hvs and Hvs-Brn. Performing a round trip corresponding to this circulation takes 51+22=73 minutes. As round trips start every 20 minutes, a train assigned to this circulation only has a downtime of 7 minutes before starting its next circulation. By assigning four trains to this circulation, all train services corresponding to the two lines in the circulation can be performed, saving one train compared to the situation with fixed circulations. Similarly, by assigning four trains to the circulation with the lines Ut-Brn and Ut-Dld all train services are covered, again saving one train.

TABLE 6.4: Solution obtained with the edge objective and *Flexible* setting.

Line	Frequency	Round trip time (min.)	Downtime under fixed circulations (min.)	Trains under fixed circulations
Ut - Hvs	3	51	9	3
Hvs - Brn	3	22	18	2
Ut - Brn	2	78	12	3
Ut - Dld	2	37	23	2

Edge Objective versus OD Objective

For the *Basic* setting, optimizing the edge objective results in the same solution as optimizing the OD objective. When rolling stock constraints are included, it is not possible to maintain high frequencies on all edges, hence differences emerge between the objectives. Most notably is that the solutions obtained with the OD objective only have one train per hour between Den Dolder and Baarn instead of two per hour. For the edge objective, this part of the network accounts for a large part of the objective function as it contains four of the in total eleven edges. On the other hand, for the OD objective the importance is rather small as there are not that many passengers that use this part of the network. The rolling stock that becomes available by reducing the frequency between Den Dolder and Baarn is used to increase the frequency between Hilversum and Baarn, an edge used by many passengers traveling between the eastern part of the Netherlands and Amsterdam; for the *Fixed* setting the frequency increases from 1 to 2, for the *Flexible* setting it increases from 3 to 4. The



(A) Edge+Basic(+Dwell)

(B) OD+Basic(+Dwell)

(C) Edge+Fixed(+Dwell)

(D) OD+Fixed(+Dwell)

(E) Edge+Flexible

(F) OD+Flexible

FIGURE 6.2: Line plans for the small disrupted region obtained with the different model settings.

OD objective with the *Flexible* setting leads to a somewhat disadvantageous solution for passengers traveling between Utrecht Centraal and Den Dolder. According to the formulation of the objective, the OD frequency comparing to these passengers equals $\min\{6, 4, 4\} = 4$. However, only two trains per hour run directly between Utrecht Centraal and Den Dolder. This demonstrates that OD objective does not always result in the most passenger oriented solution, because it does not differentiate between indirect and direct travel options.

6.2.3 Large Disrupted Region

In Table 6.5, we present the values of the edge objective, OD objective, the number of lines and the number of required trains under fixed circulations of the optimal solutions of the large instance for the different model settings. The obtained line plans are visualized in Figures 6.3a-6.3g.

General Observations
It can be observed that the intercity lines are the same for all settings. Compared to the original line plan, the intercity between Amsterdam Zuid and Utrecht Centraal is canceled entirely, as a consequence of the limited turning capacity at both of these stations. Next to that, the intercity frequency between Amsterdam Zuid and Almere Centrum is reduced from 4 to 2. The intercity lines from Amsterdam Centraal and Amsterdam Zuid to the east of the Netherlands maintain their frequency and turn in Hilversum. Another 'constant' in all line plans is the regional line between Amsterdam Centraal and Utrecht Centraal with frequency 4. A final general observation is that all line plans look relatively similar to the original line plan. Decoupling stations that do not serve as a terminal station in the regular line plan are only rarely used as terminal stations in the adapted line plans. A difference with the regular line plan emerging in most solutions is that the regular line plan contains regional lines with frequency 2 connecting Amsterdam Centraal and Amsterdam Zuid with both Almere Centrum and Hilversum. In most adapted line plans, the solution contains only a single regional line from Amsterdam Centraal and a single line from Amsterdam Zuid (both with frequency 4), such that many passengers will have to transfer at Weesp. The algorithm does this, as it reduces the number of lines but maintains the same edge and OD frequencies.

Impact of the Dwell Constraint
In contrast to the small instance, the dwell constraint does have an impact on the solution in the large instance. This implies that the line plans generated with the *Basic* setting with either objective and the *Fixed* setting with the edge objective do not satisfy the dwell constraint. The relevant line plans are depicted in Figures 6.3a and 6.3c.

TABLE 6.5: Solution characteristics of the large instance with different model settings. The column 'Trains' indicates the required number of trains to operate the line plan if all circulations are fixed. "IC" for intercity, "R" stands for regional.

(A) Large instance with edge objective

Setting	Edge obj.	OD obj.	IC Lines	R Lines	IC Trains	R Trains
Basic	0.034	0.031	5	8	13	34
Dwell	0.034	0.031	5	8	13	35
Fixed	0.060	0.069	5	7	13	28
Dwell + Fixed	0.059	0.065	5	9	13	28
Flexible	0.050	0.050	5	9	13	32

(B) Large instance with OD objective

Setting	Edge obj.	OD obj.	IC Lines	R Lines	IC Trains	R Trains
Basic	0.034	0.031	5	8	13	34
Dwell	0.034	0.031	5	8	13	35
Fixed	0.079	0.057	5	9	13	28
Dwell + Fixed	0.079	0.057	5	9	13	28
Flexible	0.053	0.048	5	7	13	30

TABLE 6.6: Characteristics of the line plans with settings *Basic* and *Fixed* (with the edge objective) to illustrate why the dwell constraint is violated.

Station	Turning trains	Platforms	Slack
Amsterdam Zuid	8	1	4 minutes
Almere Centrum	10	2	50 minutes

Line	f_l	δ_l	Required Slack
IC Asdz-Alm	2	9	18 minutes
R Asdz-Alm	4	13	52 minutes

It turns out that in both cases the dwell constraint is violated at Amsterdam Zuid and Almere Centrum. We illustrate this using Table 6.6, where the relevant characteristics of the line plans that violate the dwell constraint are presented. Without any additional dwell time, the slack at Amsterdam Zuid and Almere Centrum combined is 54 minutes. Since we imposed that all additional dwell time must be spend at the terminal stations, the lines between Amsterdam Zuid and Almere Centrum can only make use of this slack. However, because the additional dwell time of these lines are rather large (9 and 13 minutes) there is not enough capacity at these stations to accommodate for the additional dwell time.

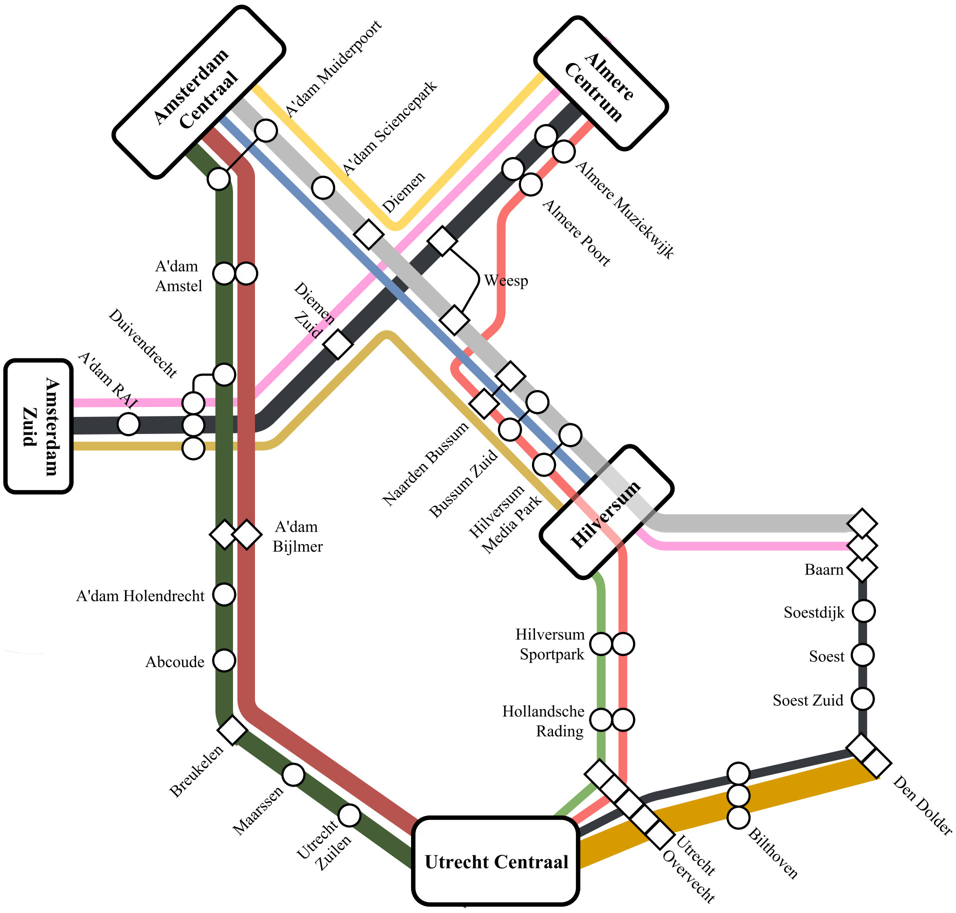
In the line plan obtained with the *Dwell* setting, the line plan is altered to satisfy the dwell constraint by letting the regional line from Amsterdam Zuid go to Hilversum instead of to Almere Centrum, and by letting the regional line from Amsterdam Centraal go to Almere Centrum instead of Baarn (direction Hilversum). These lines have much shorter additional dwell times, such that the constraint is no longer violated. The objective values do not change, as all edge frequencies remain the same. In the line plan obtained with the *Dwell+Fixed* setting and the edge objective, the constraint is satisfied by reducing the frequency of the regional line between Amsterdam Zuid and Almere from 4 to 2 and introducing new regional lines between Amsterdam

Zuid and Diemen Zuid with frequency 2 and between Diemen Zuid and Almere Centrum with frequency 1. This also spares a train, such that the frequency of the line between Utrecht Centraal and Hilversum can be increased. The value of the edge objective value actually increases when the dwell constraint is added, but this does not violate the relations (6.2) since the number of lines increases from 12 to 14.

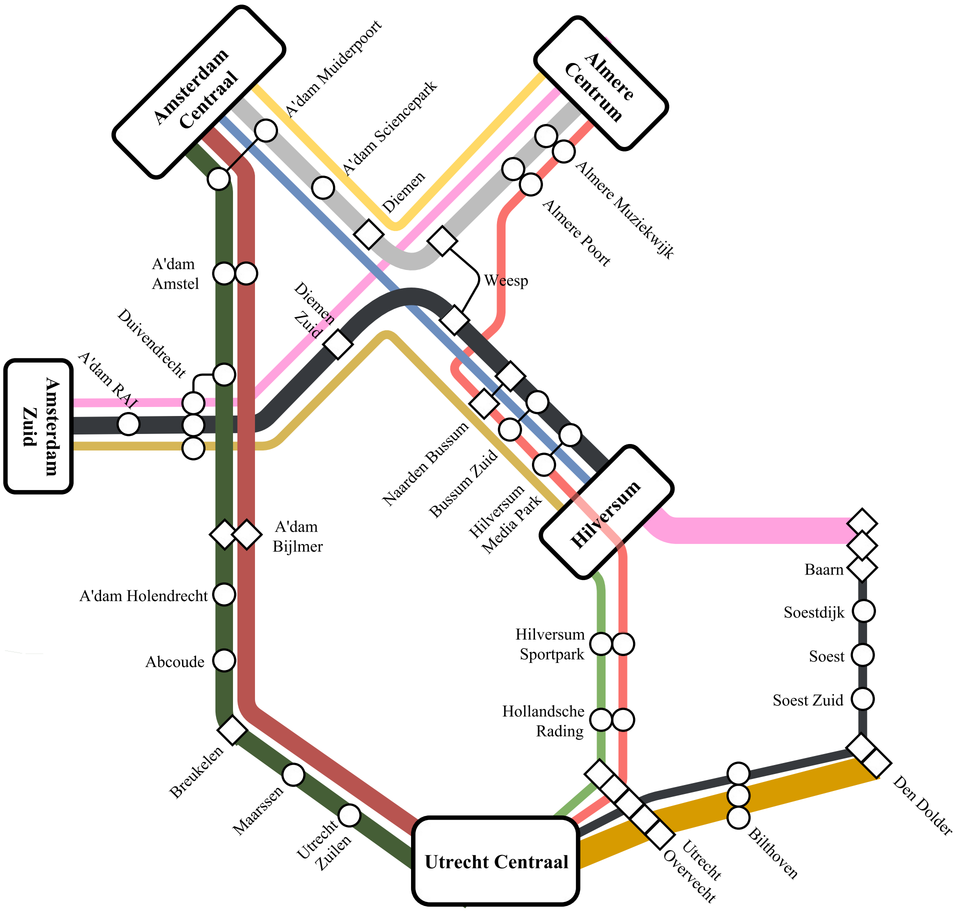
Impact of Rolling Stock Constraints and Objective
As in the small instance, the solutions without rolling stock constraints do not depend on the chosen objective. Next to that, the number of available intercity trains does not put a restriction on the line plan, as the number of necessary intercity trains is 13 with all settings, whereas there are 19 available. The regional lines are impacted when rolling stock constraints are included, with only 28 available in the disrupted region. As with the small instance, the *Flexible* setting leads to more efficient rolling stock circulations, saving four trains with the edge objective and two trains with the OD objective.

When optimizing the edge objective, it can be observed that the *Fixed* setting results in reducing the frequency of the line between Amsterdam Centraal and Baarn from 4 to 3 and reducing the frequency of the line between Utrecht Centraal and Den Dolder from 6 to 3. Next to that, the regional line between Hilversum and Baarn is canceled. It is not surprising that the decreases in frequency occur in this region, as the regional trains in this region have to be used to cover both the original regional lines and the original intercity lines. When the *Fixed* setting is used with the OD objective, the lines between Amsterdam Centraal and Baarn and between Utrecht Centraal and Den Dolder maintain their high frequency. Instead, the frequencies are reduced between Den Dolder and Baarn, Utrecht Centraal and Almere Centrum, and between Diemen Zuid and Almere Centrum, parts of the network with fewer passengers.

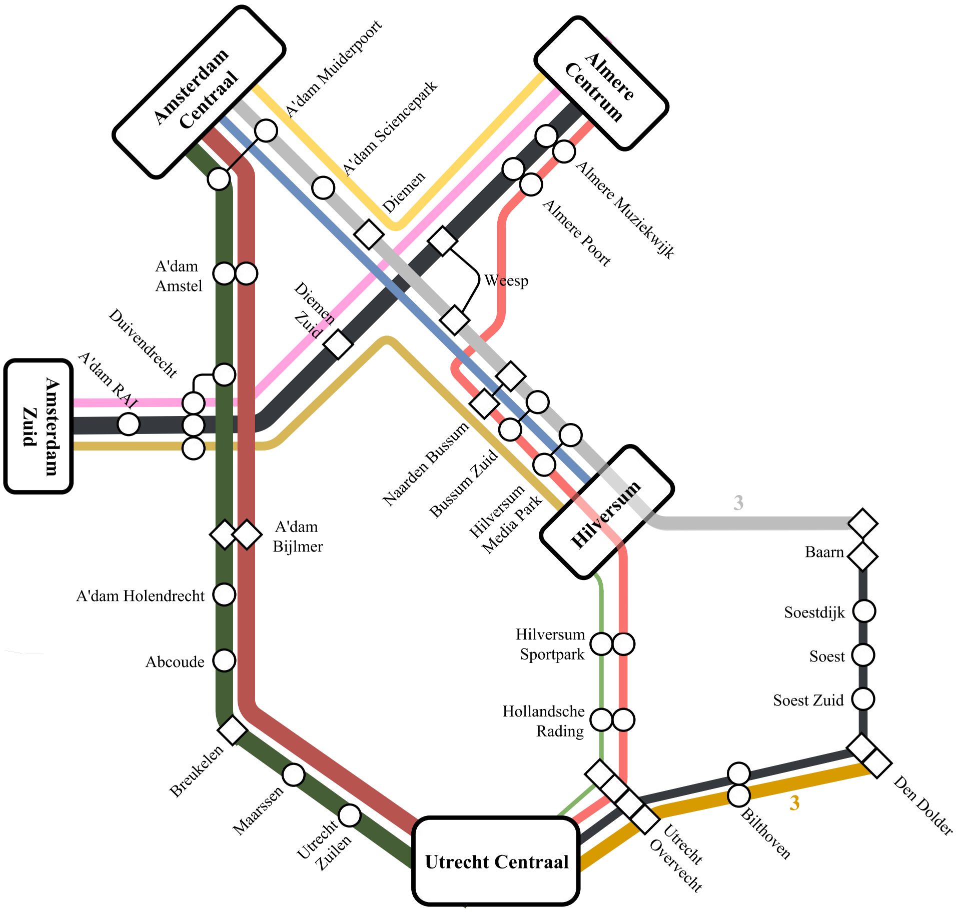
As in the small instance, allowing for flexible circulations results in more dense line plans. With the edge objective, the frequencies of lines that have frequency 3 with the *Fixed* setting are increased to 4 with the *Flexible* setting. Next to that, an additional regional line is introduced between Amsterdam Centraal and Weesp. With the OD objective, the *Flexible* settings leads to increased frequencies around Utrecht and between Weesp and Almere Centrum.



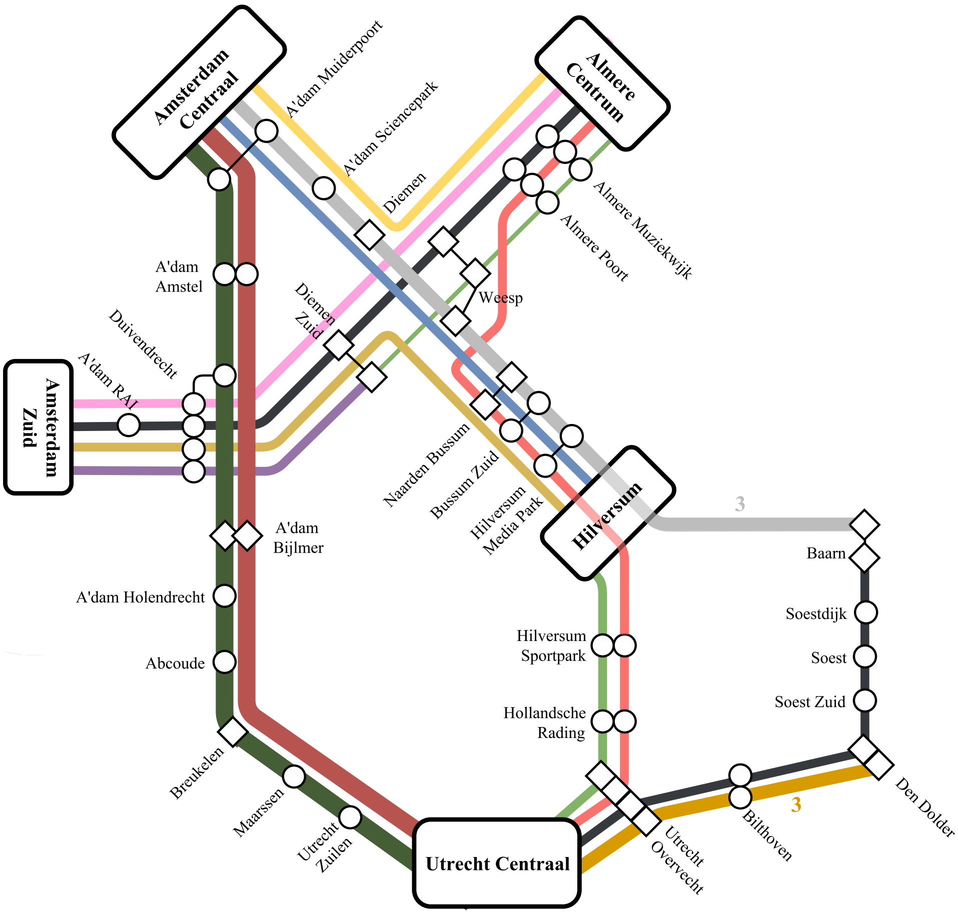
(A) (Edge or OD)+Basic



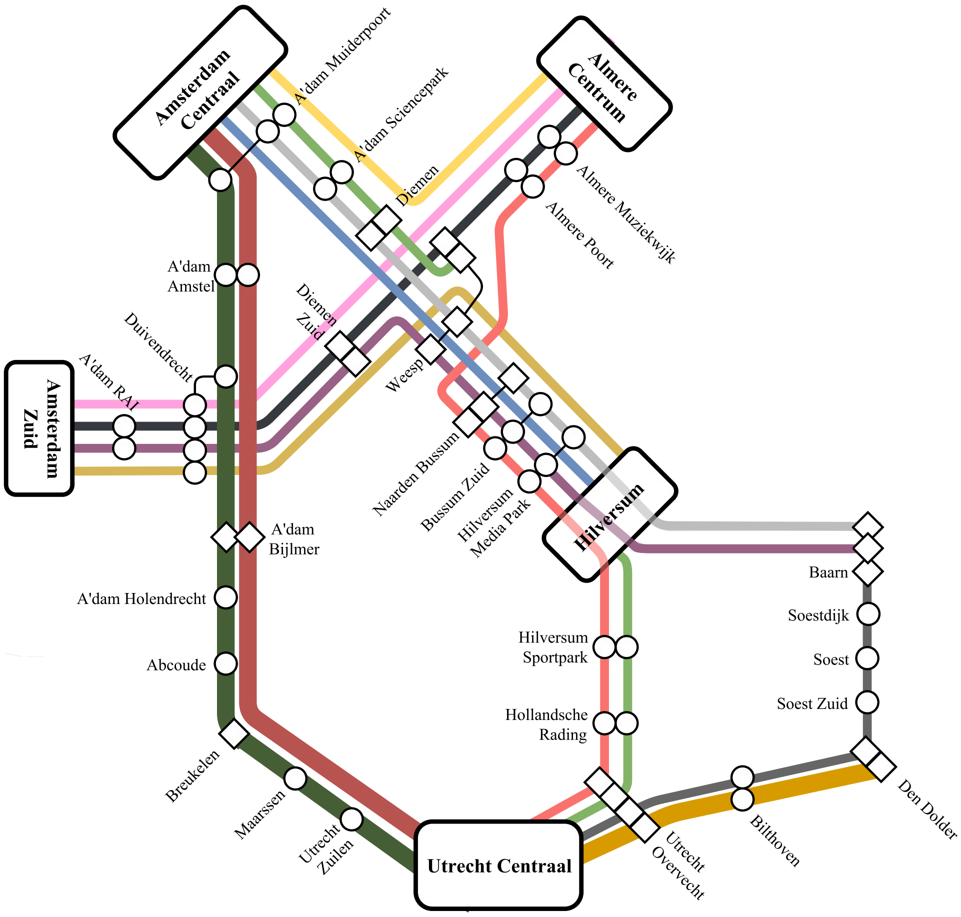
(B) (Edge or OD)+Dwell



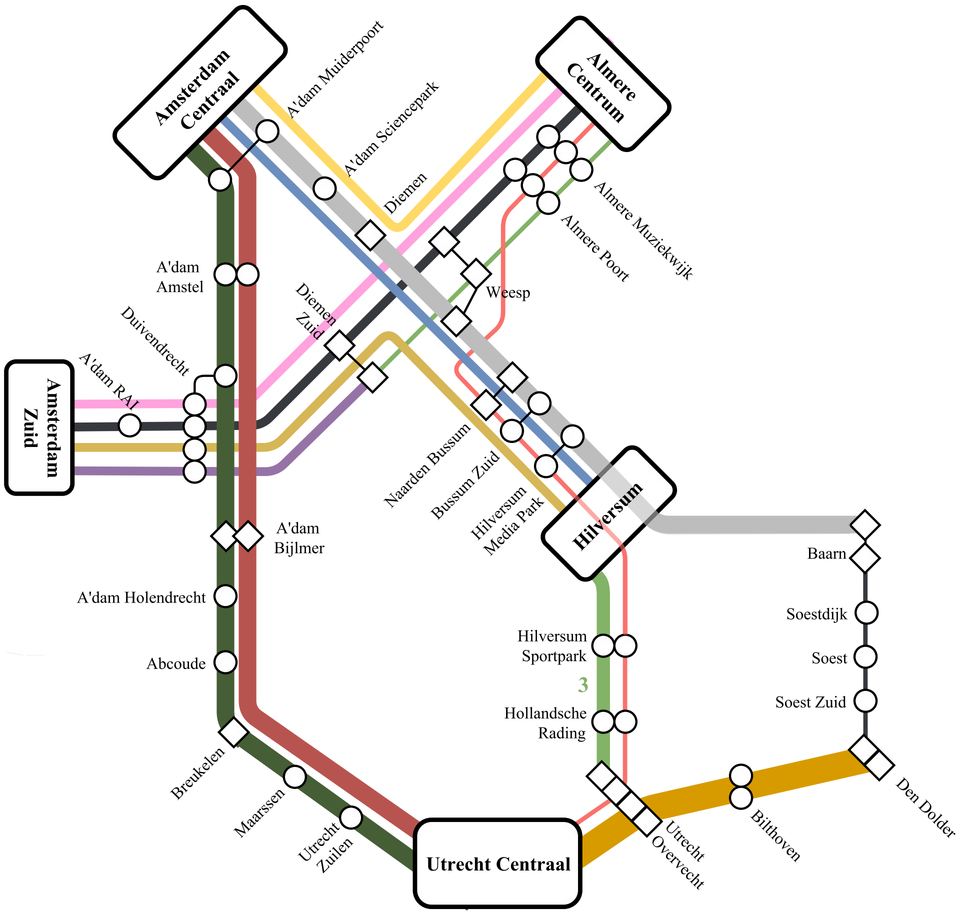
(C) Edge+Fixed



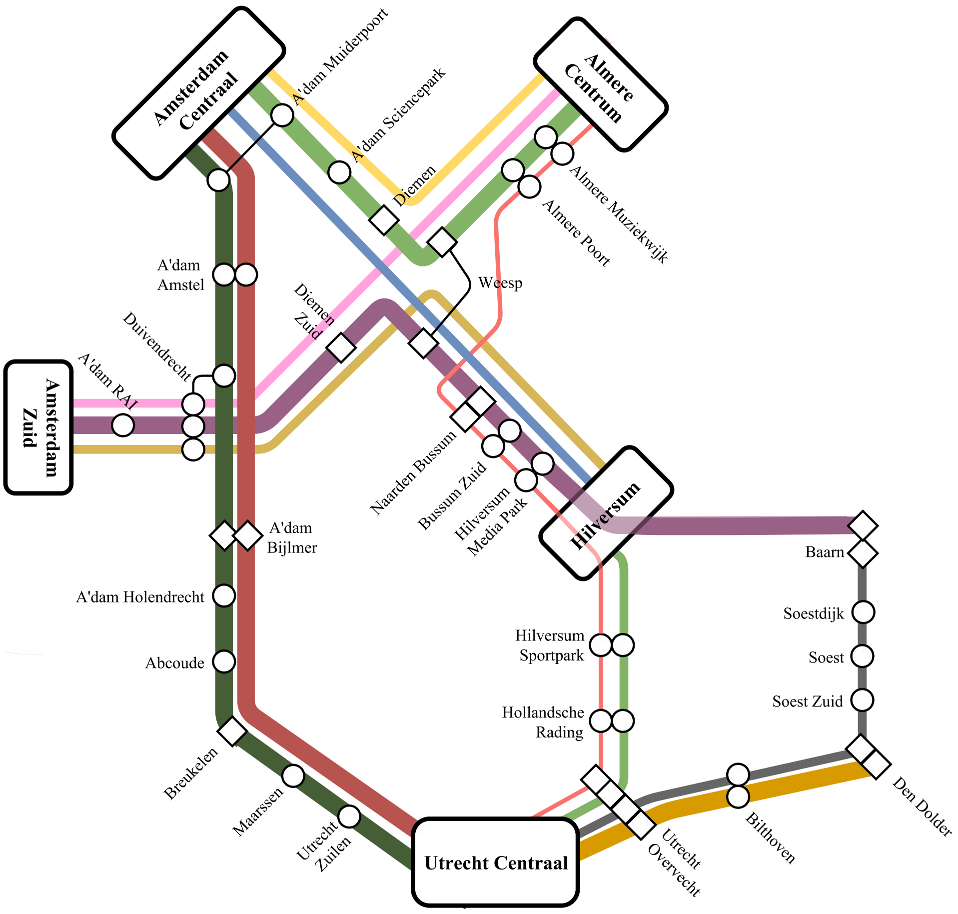
(D) Edge+Fixed+Dwell



(E) Edge+Flexible



(F) OD+Fixed(+Dwell)



(G) OD+Flexible

FIGURE 6.3: Line plans for the large disrupted region obtained with the different model settings. As in the other figures, the thickness indicates the frequency.

6.3 Simulation Results

We evaluate all line plans obtained for the small and large disrupted region in combination with the dispatching strategies described in Chapter 5. In all simulations, we use a simulated time of 4 hours. By analyzing and comparing the three performance measures, *frequency*, *delay* and *regularity*, we investigate which combinations of model settings and dispatching strategies result in smooth operations.

6.3.1 Small Disrupted Region

As there are no intercity lines in the line plans for the small disrupted region, we test four strategies: FIFO-STAT, FIFO-DYN, SYNC-STAT and SYNC-DYN. In Figure 6.4, the performance measures are plotted over time for all combinations of line plans (depicted in Figures 6.2a-6.2f) and dispatching strategies. The values of the performance measures after 4 hours are reported in Table A.1 in Appendix A.

As expected, the *Basic* line plan leads to a very bad performance regardless of the strategy. This line plan requires much more trains than there are available in the region. Therefore, the frequency measure is only about 0.7 for all strategies, indicating that the realized frequencies are 30 percent lower than desired. Next to that, the delay measure is around 1.1, implying that the on average trains are delayed with 10 percent of the minimum travel time. The line plan also performs badly on the regularity measure, because the times between two departures in the same direction are much longer than desired. The *Basic* line plan does lead to a stable performance, since the scores on the three measures are fairly constant after 1 hour.

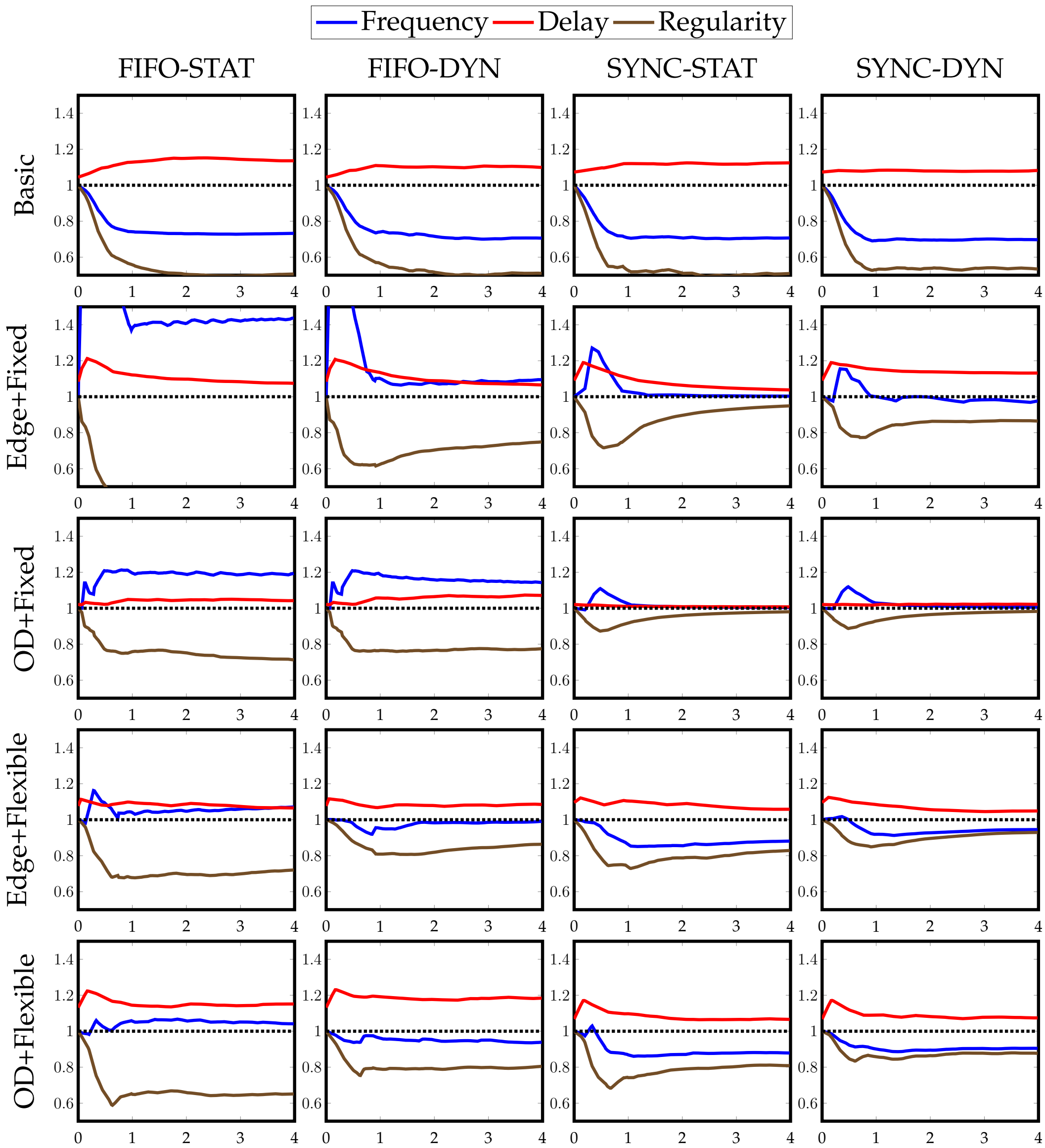


FIGURE 6.4: Simulation results of the small instance for the different line plans (arranged vertically) and different dispatching strategies (arranged horizontally). In the figures, the horizontal axis denotes the time in hours and the vertical axis the score on the three measures. The closer a measure is to the dashed line, the better the performance.

The performance of the line plans obtained with the *Fixed* setting considerably depends on the applied dispatching strategy. The *FIFO* principle results in realized frequencies that are much higher than as specified in the line plans, which is not surprising as the principle always instructs trains to depart as soon as possible. This behavior is also reflected in the low regularity measures. The *SYNC* principle leads to much better performance on the frequency and regularity measures since it instructs trains to wait at terminal stations in order to meet the required frequency. The frequency measures using this principle are very close to 1. Whether trains are assigned statically or dynamically has a small impact on the performance. Apart from a peak that occurs around 30 minutes into the simulation, the performance is very stable with the *SYNC* principle.

The frequencies of the line plans obtained with the *Flexible* setting and the FIFO principle are relatively closer to 1 compared to the *Fixed* setting as these line plans are more dense. The regularity measures however are still much smaller than 1. It turns out that the performance per line differs strongly if the FIFO-STAT strategy is used. Some lines have sufficient trains and too many departures but other lines have too few trains and therefore too few departures. As such, the average frequency measure is close to one, but the regularity score is very bad. The FIFO-DYN strategy performs much better than the FIFO-STAT measure, since trains are shared among lines such that lines cannot consistently have a shortage of trains. With the SYNC-STAT strategy the frequency measures for both line plans are 0.88, below 1 as expected since the line plans cannot be operated with fixed circulations. By applying the SYNC-DYN strategy, the frequency measures improve to 0.95 with the edge objective and 0.90 with the OD objective. The regularity measures also increase strongly when trains are assigned dynamically, from 0.83 to 0.93 and from 0.81 to 0.88 with the edge and OD objective respectively. The delay measures remain around 1.06.

6.3.2 Large Disrupted Region

For the large instance we test all six strategies: FIFO-STAT, FIFO-DYN, SYNC-STAT and SYNC-DYN, SYNC+COOR-STAT and SYNC+COOR-DYN. We first compare the performance of the FIFO principle in comparison with the STAT principle and the performance of the *STAT* principle in comparison with the DYN principle. The performances of these strategies are plotted for the different line plans (depicted in Figures 6.3a-6.3g) in Figure 6.5. Thereafter we compare the SYNC principle and SYNC+COOR principle. The values of the performance measures after 4 hours are reported in Table A.2 in Appendix A.

In general it can be seen that the delay measure is higher in the large instance compared to the small instance for all line plans and strategies. This makes sense, as the lines in the large disrupted region are longer, hence there are more stations where delays can occur. This also causes the frequency measures to be slightly lower in the large disrupted region when rolling stock restrictions are included, since the number of assigned trains is based on the minimum travel times. In other words, there may be still be a small shortage of trains despite the rolling stock restrictions. When rolling stock restrictions are not included, frequency measures are actually higher than in the small instance. This can be explained by the less severe shortage of trains that exists for the *Basic* and *Dwell* line plans. For intercity trains, there even is no shortage at all.

Another general observation is that the SYNC principle clearly outperforms the FIFO principle, as it results in better scores on the delay and regularity measure, largely offsetting its lower frequencies. Even more, the performance of strategies with the SYNC principle stabilizes much more quickly compared to the FIFO principle. Assigning trains dynamically rather than statically also increases the performance, most noticeably with respect to the regularity.

All line plans with fixed rolling stock constraints perform fairly well, especially when the SYNC principle is used. The DYN principle has similar performance as the STAT principle on the frequency and delay measures, but the regularity visibly improves when trains are assigned dynamically. There is no clear difference in performance between the *Edge+Fixed* line plan and the *Edge+Fixed+Dwell* line plan. This means that even though the *Edge+Fixed* line plan violates the dwell constraint and therefore there does not exist a timetable for this line plan where all travel times are at their minimum, this is not noticeable when operating the line system without a timetable. Further analysis showed that by including the dwell constraint the average waiting time at Almere Centrum (where the dwell constraint was violated) does reduce from about 2 minutes to 1 minute, but this improvement is accompanied by increased delays at other stations.

The line plans with flexible circulations have lower scores on the frequency measure compared to the line plans with fixed circulations, even when trains are assigned dynamically. One may however still prefer the line plans with flexible circulations, as these are more dense and can therefore transport more passengers. Even more, the frequency measure is still reasonably close to 1. All measures can be observed to improve when the DYN principle is applied.

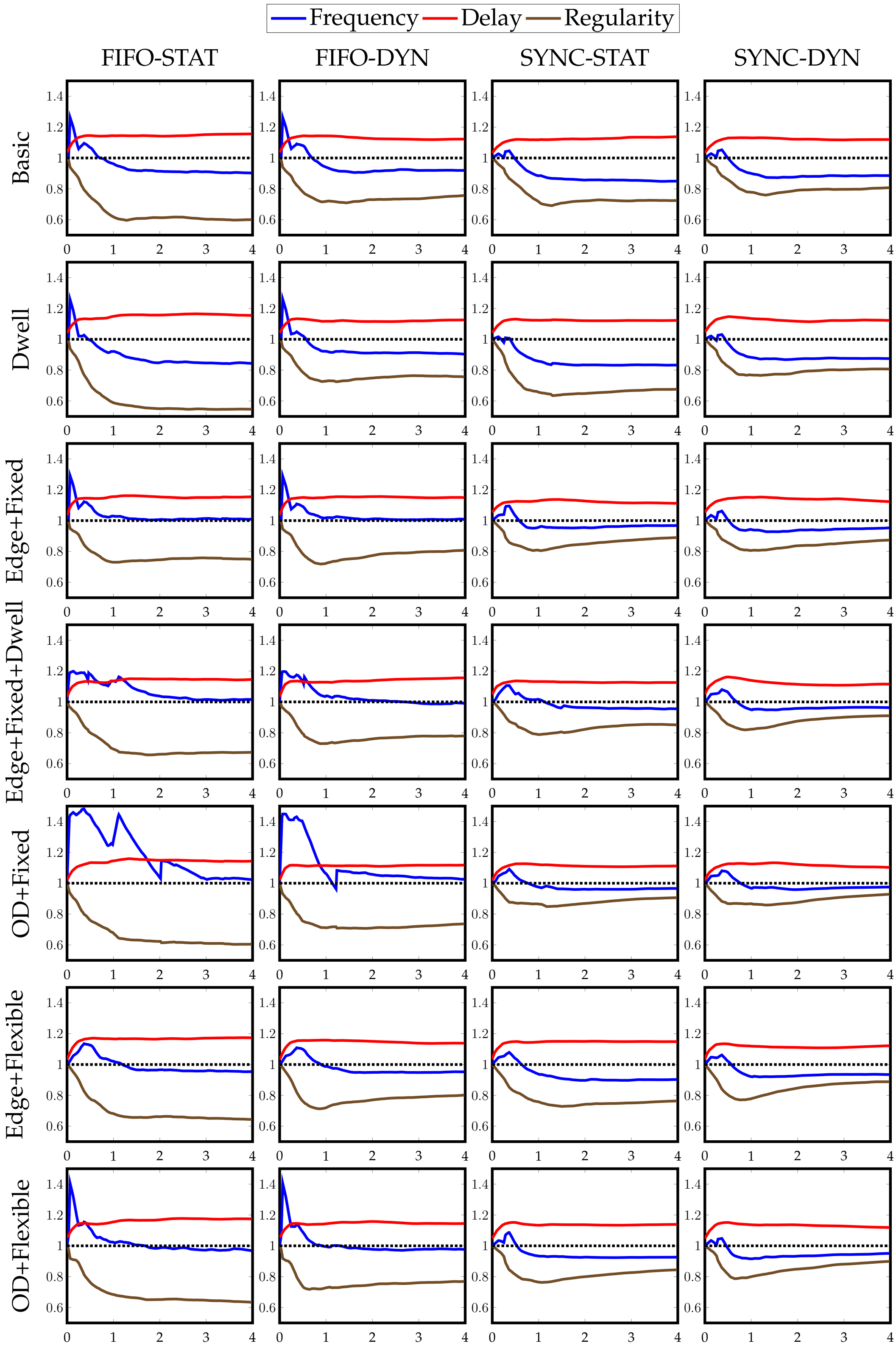
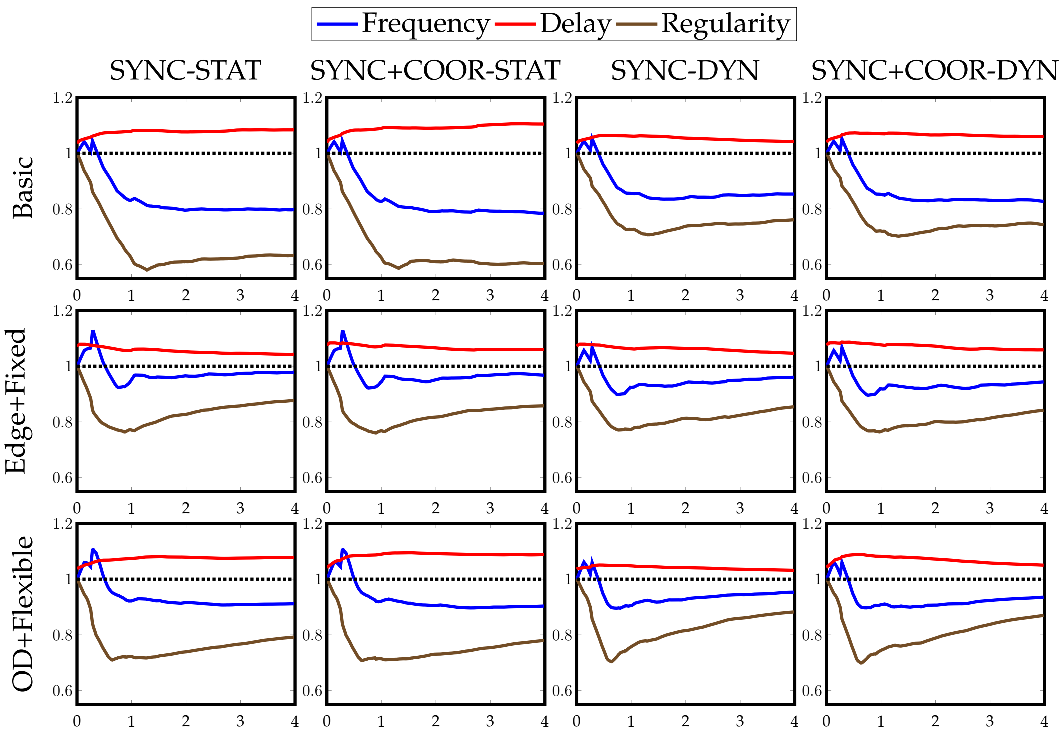


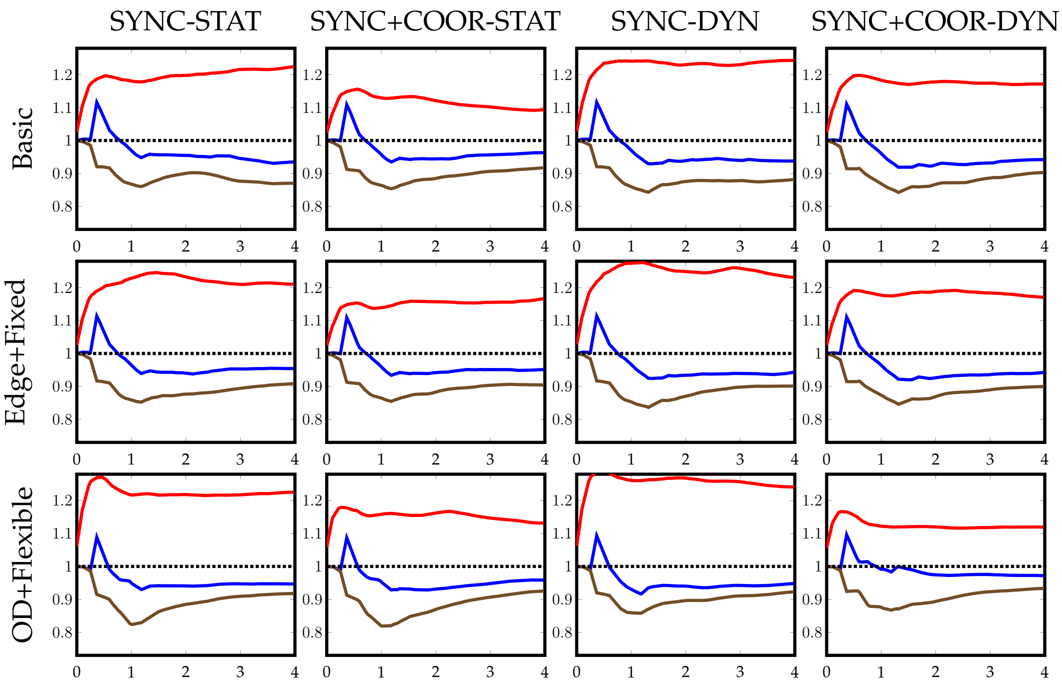
FIGURE 6.5: Simulation results of the large instance for the different line plans (arranged vertically) and different dispatching strategies (arranged horizontally). In the figures, the horizontal axis denotes the time in hours and the vertical axis the score on the three measures. The closer a measure is to the dashed line, the better the performance.

We can further analyze the differences between the performance of the SYNC and SYNC+COOR strategy using Figures 6.6a and 6.6b. Here, the performance of the dispatching strategies (except those with the FIFO principle) is presented for a selection of line plans and for regional and intercity lines separately. Note that the scale of the vertical axis is different than in the previous figures. The line plans with the dwell constraint are left out as they were observed to not lead to differences in performance. The values after 4 hours are again reported in Appendix A.

It can be noticed that the intercity lines endure much longer delays than the regional lines. With the SYNC+COOR principle, the delay measure of regional lines increases, as this principle dictates regional trains to wait at stations to let intercity trains pass. The increase however is only about 0.02, meaning that the total travel time of regional trains increases with 2 percentage points. On the other hand, the travel time of intercity trains decreases considerably, up to 12 percentage points. The performances on the other two measures do not seem to be impacted. As the increase in delay for regional trains is limited, it might be worthwhile to let regional trains wait even longer than the maximum of 3 minutes as is imposed in the simulation experiments.



(A) Regional



(B) Intercity

FIGURE 6.6: Simulation results for regional and intercity lines separately for a selection of line plans (arranged vertically) and dispatching strategies (arranged horizontally). In the figures, the horizontal axis denotes the time in hours and the vertical axis the score on the three measures. The closer a measure is to the dashed line, the better the performance.

6.4 Key Findings

The results show that the developed algorithm is able to provide practicable line plans in real time. The fastest computation times are obtained by considering capacity at platform level and generating all logically implied cuts when infeasibilities are encountered.

The selected settings and dispatching strategy greatly impact the density and performance of the line plan. If rolling stock restrictions are not taken into account, the algorithm produces an overly optimistic line plan that cannot be operated. With fixed rolling stock circulations, the resulting line plans are feasible. Applying the SYNC principle along with either the STAT or DYN principle results in good performance on all three analyzed performance measures, with dynamic assignment performing slightly better than static assignment. When flexible circulations are allowed in the line planning algorithm, the produced line plan is more dense and hence better for passengers if operated in a satisfactory manner. As expected, these line plans should be operated with the DYN principle, as the line plan is too dense to operate with fixed circulations. The SYNC+COOR principle shows clear benefits over the SYNC principle, strongly reducing the delays of intercity trains.

In contrast to the rolling stock constraints, the dwell constraint does not have a large influence on the line plan. As it turns out, the constraint is rarely violated when it is not included in the model. Even more, even when a line plan violates this constraint (implying that there exists no compatible timetable with minimum travel times), this is not visible in the results of the simulation.

Finally, the OD objective leads to more passenger oriented line plans and has similar computation times as the edge objective. It might be able to further improve the objective by differentiating between direct and indirect OD-frequencies.

Chapter 7

Conclusion

This thesis addressed disruption management strategies in (near) out-of-control situations occurring in railway systems. We developed a novel Benders'-like algorithm for line (re-)planning. In order to ensure that the resulting line plan is feasible with respect to the available railway infrastructure and rolling stock units, the proposed algorithm partially integrates line planning with timetabling and rolling stock scheduling. To promote the attractiveness of the adapted line plan from the passengers' point of view, we introduced a new objective that aims to minimize the decrease in travel options per hour between origin destination pairs (the OD-objective). Besides investigating which lines should be operated when the system gets out-of-control, we also analyzed how the adapted line system should be operated. To this end, we developed several local dispatching strategies that only require local forms of coordination.

Computational experiments based on disruptions in the Dutch railway network indicate that the algorithm performs well. The algorithm manages to find profitable and workable line plans for both a small and a large disrupted region within a couple of minutes. The fastest results are obtained when capacity is incorporated on platform level in the line planning model and all logically inferred cuts are added in every iteration. The OD-objective leads to more passenger oriented line plans than the previously used edge objective, with similar computation times.

Using simulation, we also demonstrated that by applying the appropriate dispatching strategies, the produced line plans can be operated smoothly with minimal coordination. If flexible rolling stock circulations are allowed in the line planning model (which results in more dense line plans), it is of course necessary to apply the DYN principle such that flexible turnings are also allowed when the trains are operated. Otherwise, the STAT principle, which fixes the assignment between trains and lines, suffices. Along with an appropriate turning principle, the SYNC principle, which holds trains at terminal station to enforce regular departure patterns, results in a strategy that generates stable and regular operations. The performance is improved further by letting regional trains wait for intercity trains at critical points in the network (the SYNC+COOR principle).

Considering that computerized support is currently only used for crew rescheduling at NS, it is unlikely that the algorithm developed in this thesis will be implemented in the next couple of years. However, based on the results we do recommend that NS makes the rescheduling process less rigid and more flexible, especially in out-of-control situations. Instead of holding off making rescheduling decisions until all information systems are up-to-date and the whereabouts of all the crew is known, reschedulers should continuously dispatch trains in disrupted regions, using the crew and resources they are certain are available. While further research is required, the results of this thesis indicate that this can lead to stable and regular operations, and therefore improved experiences of passengers.

In the line planning algorithm and dispatching strategies developed in this thesis train drivers and conductors are fixed to trains. However, crew duties must conform to many requirements. Most importantly, crews stationed outside the disrupted region but currently located inside this region must at some point return to their base, and vice versa. Therefore, a relevant topic for further research is crew rescheduling in out-of-control situations using local, possibly agent-based, scheduling principles. Other directions for further research include the transition from the adapted line plan to the regular line plan and timetable and the application of the methods and new mathematical formulations developed in this thesis to integrated models for railway planning, for instance in the 'eigenmodel' discussed in Schöbel (2017).

Bibliography

Abbink, E. J., Mobach, D. G., Fioole, P. J., Kroon, L. G., van der Heijden, E. H., & Wijngaards, N. J. (2010). Real-time train driver rescheduling by actor-agent techniques. *Public Transport*, *2*(3), 249–268.

Ball, R., Panja, D., & Barkema, G. T. (2016). *A two component railway model exhibiting service collapse* [Working Paper]. University of Warwick.

Borndörfer, R., Grötschel, M., & Pfetsch, M. E. (2007). A column-generation approach to line planning in public transport. *Transportation Science*, *41*(1), 123–132.

Burggraeve, S., Bull, S. H., Vansteenwegen, P., & Lusby, R. M. (2016). *Integrating robust timetabling in line plan optimization for railway systems* (Tech. Rep.). Technical Report. Preprint, presented at EURO'16.

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

Caprara, A., Kroon, L., & Toth, P. (2011). Optimization problems in passenger railway systems. *Wiley Encyclopedia of Operations Research and Management Science*.

Claessens, M., van Dijk, N. M., & Zwaneveld, P. J. (1998). Cost optimal allocation of rail passenger lines. *European Journal of Operational Research*, *110*(3), 474–489.

Codato, G., & Fischetti, M. (2006). Combinatorial Benders' cuts for mixed-integer linear programming. *Operations Research*, *54*(4), 756–766.

Dollevoet, T., Huisman, D., Kroon, L., Schmidt, M., & Schöbel, A. (2014). Delay management including capacities of stations. *Transportation Science*, *49*(2), 185–203.

Dollevoet, T., Huisman, D., Schmidt, M., & Schöbel, A. (2012). Delay management with rerouting of passengers. *Transportation Science*, *46*(1), 74–89.

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

Fioole, P.-J., Kroon, L., Maróti, G., & Schrijver, A. (2006). A rolling stock circulation model for combining and splitting of passenger trains. *European Journal of Operational Research*, *174*(2), 1281–1297.

Gattermann, P., Harbering, J., & Schöbel, A. (2017). Line pool generation. *Public Transport*, *9*(1-2), 7–32.

Goerigk, M., & Schmidt, M. (2017). Line planning with user-optimal route choice.

European Journal of Operational Research, *259*(2), 424-436.

Goossens, J.-W., Van Hoesel, S., & Kroon, L. (2004). A branch-and-cut approach for solving railway line-planning problems. *Transportation Science*, *38*(3), 379–393.

Goossens, J.-W., van Hoesel, S., & Kroon, L. (2006). On solving multi-type railway line planning problems. *European Journal of Operational Research*, *168*(2), 403–424.

Haahr, J. T., Wagenaar, J. C., Veelenturf, L. P., & Kroon, L. G. (2016). A comparison of two exact methods for passenger railway rolling stock (re)scheduling. *Transportation Research Part E: Logistics and Transportation Review*, *91*, 15–32.

Huisman, D. (2007). A column generation approach for the rail crew re-scheduling problem. *European Journal of Operational Research*, *180*(1), 163–173.

Jespersen-Groth, J., Potthoff, D., Clausen, J., Huisman, D., Kroon, L., Maróti, G., & Nielsen, M. N. (2009). Disruption management in passenger railway transportation. In *Robust and online large-scale optimization* (pp. 399–421). Springer.

Kaspi, M., & Raviv, T. (2013). Service-oriented line planning and timetabling for passenger trains. *Transportation Science*, *47*(3), 295–311.

Kepaptsoglou, K., & Karlaftis, M. (2009). Transit route network design problem: review. *Journal of transportation engineering*, *135*(8), 491–505.

Kroon, L., Huisman, D., Abbink, E., Fioole, P.-J., Fischetti, M., Maróti, G., ... Ybema, R. (2009). The new dutch timetable: The OR revolution. *Interfaces*, *39*(1), 6–17.

Kroon, L., Maróti, G., & Nielsen, L. (2014). Rescheduling of railway rolling stock with dynamic passenger flows. *Transportation Science*, *49*(2), 165–184.

Lamorgese, L., Mannino, C., & Piacentini, M. (2016). Optimal train dispatching by Benders'-like reformulation. *Transportation Science*, *50*(3), 910–925.

Liebchen, C., & Möhring, R. H. (2007). The modeling power of the periodic event scheduling problem: railway timetables—and beyond. In *Algorithmic methods for railway optimization* (pp. 3–40). Springer.

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

Nachtigall, K., & Jerosch, K. (2008). Simultaneous network line planning and traffic assignment. In *Oasics-openaccess series in informatics* (Vol. 9).

Nielsen, L. K. (2011). *Rolling stock rescheduling in passenger railways: Applications in short-term planning and in disruption management* (No. EPS-2011-224-LIS).

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

Odijk, M. A., Lentink, R., & Steenbeek, A. (2002). *CADANS/Conflex: Functionele beschrijving van het onderdeel Conflex)* (Tech. Rep.). ORTEC Consultants BV, CWI.

Panja, D. (2016). *Improving the Resilience of Railway Systems.* Research Project NWO.

Park, B. H., Seo, Y.-I., Hong, S.-P., & Rho, H.-L. (2013). Column generation approach to line planning with various halting patterns—application to the korean high-speed railway. *Asia-Pacific Journal of Operational Research*, *30*(04), 1350006.

Peeters, L. (2003). *Cyclic Railway Timetable Optimization* (No. EPS-2003-022-LIS).

Potthoff, D., Huisman, D., & Desaulniers, G. (2010). Column generation with dynamic duty selection for railway crew rescheduling. *Transportation Science*, *44*(4), 493–505.

Schlechte, T., Borndörfer, R., Erol, B., Graffagnino, T., & Swarat, E. (2011). Micro–macro transformation of railway networks. *Journal of Rail Transport Planning & Management*, *1*(1), 38–48.

Schmidt, M., & Schöbel, A. (2015). Timetabling with passenger routing. *OR spectrum*, *37*(1), 75–97.

Schöbel, A. (2007). *Integer programming approaches for solving the delay management problem*. Springer.

Schöbel, A. (2012). Line planning in public transportation: models and methods. *OR spectrum*, *34*(3), 491–510.

Schöbel, A. (2015). Integration of line planning and timetabling. In *Proceedings of the Conference on Advanced Systems in Public Transport.*

Schöbel, A. (2017). An eigenmodel for iterative line planning, timetabling and vehicle scheduling in public transportation. *Transportation Research Part C: Emerging Technologies*, *74*, 348–365.

Schöbel, A., & Scholl, S. (2006). Line planning with minimal traveling time. In *Atmos 2005-5th workshop on algorithmic methods and models for optimization of railways.*

Schouten, S. (2017). *Line planning in (near) out-of-control situations* (Unpublished master's thesis). Tilburg University.

Serafini, P., & Ukovich, W. (1989). A mathematical model for periodic scheduling problems. *SIAM Journal on Discrete Mathematics*, *2*(4), 550–581.

Vanderbeck, F., & Wolsey, L. A. (2010). Reformulation and decomposition of integer programs. *50 Years of Integer Programming 1958-2008*, 431–502.

Veelenturf, L. P., Kidd, M. P., Cacchiani, V., Kroon, L. G., & Toth, P. (2015). A railway timetable rescheduling approach for handling large-scale disruptions. *Transportation Science*, *50*(3), 841–862.

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

Veelenturf, L. P., Potthoff, D., Huisman, D., & Kroon, L. G. (2012). Railway crew rescheduling with retiming. *Transportation research part C: emerging technologies*, *20*(1), 95–110.

Veelenturf, L. P., Potthoff, D., Huisman, D., Kroon, L. G., Maróti, G., & Wagelmans, A. P. (2014). A quasi-robust optimization approach for crew rescheduling. *Transportation Science*, *50*(1), 204–215.

Yu, G., & Qi, X. (2004). *Disruption management: framework, models and applications*. World Scientific.

Appendix A

Additional Simulation Results

TABLE A.1: Scores on the three performance measures after 4 hours of simulated time for the different line plans and dispatching strategies in the small disrupted region.

		Frequency		Delay		Regularity	
Line Plan	Strategy	STAT	DYN	STAT	DYN	STAT	DYN
Basic	FIFO	0.73	0.71	1.14	1.10	0.51	0.51
	SYNC	0.71	0.70	1.12	1.08	0.51	0.53
Edge+Fixed	FIFO	1.44	1.09	1.07	1.07	0.46	0.75
	SYNC	1.00	0.98	1.04	1.13	0.95	0.87
OD+Fixed	FIFO	1.19	1.14	1.04	1.07	0.71	0.77
	SYNC	1.00	1.01	1.01	1.02	0.98	0.98
Edge+Flexible	FIFO	1.07	0.99	1.07	1.08	0.72	0.86
	SYNC	0.88	0.95	1.06	1.05	0.83	0.93
OD+Flexible	FIFO	1.04	0.94	1.15	1.18	0.65	0.80
	SYNC	0.88	0.90	1.07	1.07	0.81	0.88

TABLE A.2: Scores on the three performance measures after 4 hours of simulated time for the different line plans and dispatching strategies in the large disrupted region.

Line Plan	Strategy	Frequency STAT	Frequency DYN	Delay STAT	Delay DYN	Regularity STAT	Regularity DYN
Basic	FIFO	0.90	0.92	1.16	1.12	0.60	0.76
	SYNC	0.85	0.89	1.14	1.12	0.72	0.81
	SYNC+COOR	0.85	0.87	1.10	1.10	0.73	0.80
Dwell	FIFO	0.84	0.90	1.15	1.13	0.55	0.76
	SYNC	0.83	0.87	1.12	1.12	0.68	0.81
	SYNC+COOR	0.83	0.87	1.09	1.08	0.68	0.80
Edge+Fixed	FIFO	1.01	1.01	1.15	1.15	0.75	0.81
	SYNC	0.97	0.95	1.11	1.12	0.89	0.87
	SYNC+COOR	0.96	0.94	1.10	1.11	0.88	0.87
Edge+Fixed+Dwell	FIFO	1.02	0.99	1.15	1.16	0.67	0.78
	SYNC	0.96	0.96	1.12	1.11	0.86	0.91
	SYNC+COOR	0.95	0.94	1.12	1.11	0.84	0.87
OD+Fixed	FIFO	1.02	1.03	1.14	1.12	0.60	0.74
	SYNC	0.97	0.98	1.11	1.10	0.91	0.93
	SYNC+COOR	0.96	0.96	1.10	1.11	0.90	0.92
Edge+Flexible	FIFO	0.95	0.95	1.17	1.14	0.64	0.80
	SYNC	0.90	0.93	1.15	1.12	0.76	0.89
	SYNC+COOR	0.89	0.93	1.14	1.10	0.75	0.88
OD+Flexible	FIFO	0.97	0.98	1.17	1.14	0.63	0.77
	SYNC	0.93	0.95	1.14	1.12	0.84	0.90
	SYNC+COOR	0.93	0.95	1.11	1.08	0.84	0.90

TABLE A.3: Performance of regional lines and intercity lines separately with different line plans and dispatching strategies. The closer a measure is to 1.00, the better the performance.

(A) Regional

	Strategy	Frequency		Delay		Regularity	
		STAT	DYN	STAT	DYN	STAT	DYN
Basic	SYNC	0.80	0.85	1.08	1.04	0.63	0.76
	SYNC+COOR	0.79	0.83	1.10	1.06	0.61	0.74
Edge+Fixed	SYNC	0.98	0.96	1.04	1.05	0.88	0.85
	SYNC+COOR	0.97	0.94	1.06	1.06	0.86	0.84
OD+Fixed	SYNC	0.98	0.98	1.03	1.04	0.93	0.92
	SYNC+COOR	0.96	0.97	1.07	1.06	0.90	0.90
Edge+Flexible	SYNC	0.88	0.93	1.09	1.04	0.69	0.88
	SYNC+COOR	0.87	0.92	1.11	1.06	0.70	0.86
OD+Flexible	SYNC	0.91	0.95	1.08	1.03	0.79	0.88
	SYNC+COOR	0.90	0.94	1.09	1.05	0.78	0.87

(B) Intercity

	Strategy	Frequency		Delay		Regularity	
		STAT	DYN	STAT	DYN	STAT	DYN
Basic	SYNC	0.93	0.94	1.22	1.24	0.87	0.88
	SYNC+COOR	0.96	0.94	1.09	1.17	0.92	0.90
Edge+Fixed	SYNC	0.95	0.94	1.21	1.23	0.91	0.90
	SYNC+COOR	0.95	0.94	1.17	1.17	0.90	0.90
OD+Fixed	SYNC	0.93	0.94	1.26	1.26	0.87	0.91
	SYNC+COOR	0.95	0.93	1.15	1.20	0.90	0.90
Edge+Flexible	SYNC	0.94	0.94	1.25	1.27	0.89	0.91
	SYNC+COOR	0.93	0.95	1.20	1.16	0.86	0.91
OD+Flexible	SYNC	0.95	0.95	1.23	1.24	0.92	0.92
	SYNC+COOR	0.96	0.97	1.13	1.12	0.93	0.93