Railway disruption management with passenger-centric rescheduling

B.M. Wentges

Master of Science Thesis

Delft University of Technology

Delft Center for Systems and Control

Railway disruption management with passenger-centric rescheduling

Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

B.M. Wentges

July 2, 2022

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

Abstract

Railway networks are subjected to disruptions on a daily basis, which may make the timetable unimplementable, which in its turn may significantly influence passenger satisfaction. In practice, train dispatchers are responsible for mitigating the influence of disruptions, such as delays of trains, train cancellations, and overcrowdedness at stations. The solutions they propose are highly dependent on their experience, often resulting in low quality solutions. In addition, the disruptions must be solved in a matter of minutes, which is challenging because of the problem scale and computational complexity. Effective models and solution approaches are required to mitigate the influence of disruptions.

In recent decades, railway rescheduling models have been developed to support train dispatchers and to improve rail services. A recent and promising model is the event-activity network model, which is a graph-based formulation that supports a wide variety of rescheduling measures. This thesis extends the event-activity network model by including rolling stock circulation with depot entry and exit operations to increase the practicability of the operator-centric model. In addition, a passenger-centric model is proposed by embedding detailed passenger-related factors into the operator-centric model, where the train capacity is included, and the detailed number of passengers in the railway network is calculated. Therefore, the effect of delays on passengers can be handled properly. The passenger-centric model can help minimize the number of waiting passengers on platforms to avoid overcrowding and to improve passenger satisfaction. In practice, the resulting passenger-centric mixed-integer linear programming (MILP) problem is hard to solve due to the introduction of binary variables for train orders, which are important for calculating the detailed number of passengers. An adaptive large neighborhood search (ALNS) algorithm is introduced to address the complexity due to train orders and to improve the computational efficiency of the passenger-centric method. With properly designed destroy and repair operators, the ALNS algorithm can explore the solution space efficiently. Therefore, a balanced trade-off between solution time and quality can be made.

Case studies are conducted based on the train lines operating between the stations of Utrecht and 's-Hertogenbosch in the Netherlands. The simulation results show that the developed model can explicitly include the number of passengers while considering the rolling stock circulation plan. Compared to directly solving an MILP problem using a commercial solver, ALNS can calculate solutions more efficiently while maintaining a high level of solution quality.

Table of Contents

Acknowledgements

I want to express my gratitude to my daily supervisor, Xiaoyu Liu, without whom I could not have achieved this final result. He has greatly aided me in creating this thesis and has greatly enhanced my academic writing.

I would like to thank my supervisor Bart De Schutter, for the monthly meetings, as he was able to maintain a high level overview of the project and ask cirtical questions.

Thanks to my roommates Wouter, Olivier, Daan and Francesco for hearing way more about trains than they ever wanted to. I want to thank Johan for the unexpected visits, and Jopie for always being around.

Delft, University of Technology B.M. Wentges
July 2, 2022

Chapter 1

Introduction

Railway transport is an important mode of transportation for many countries. Depending on the density of population in a region, railway traffic can get extremely complicated as more railway lines and trains are necessary for the high passenger demand, making the network vulnerable to disturbances and disruptions. An example of a railway network with high passenger demand is the Dutch railway network, where the largest railway operator, Nederlandse spoorwegen (NS), transported 1 million passengers per day in 2019[1]. During the same period, approximately 6000 disturbances and disruptions occurred, which is equivalent to an average of 12 per day[2]. In particular, disruptions have a major impact on the network capacity, which can make the timetable unimplementable, which in its turn may significantly influence passenger satisfaction. Finding effective rescheduling methods is essential for passenger satisfaction.

In practice, train dispatchers are responsible for mitigating the influence of disruptions. Train dispatching is a highly demanding job due to the scale and computational complexity of the problem. Pre-made contingency plans can be used in combination with the experience and skills of train dispatchers to create a disposition timetable (Chu and Oetting (2013)). Although this approach of creating a disposition timetable is used in practice, the resulting timetables are rarely optimal, since the disruption rescheduling problem is too complex to be addressed effectively by a human.

In recent decades, computer hardware and solution algorithms have improved greatly, making it possible to assist train dispatchers in finding better disposition timetables. Especially, disturbance management has been a well-researched topic, (Cacchiani et al. (2014), König (2020)), with a main focus on reducing train delays, i.e., ensuring that trains departing or arriving than intended do not have conflicts with other trains. By definition, disturbances can be handled by rescheduling the timetable without affecting the schedule of the rolling stock and the crew. Disruptions, on the other hand, have a greater impact than disturbances and require rescheduling of rolling stock and crew scheduling due to the significant delays and

[1]https://dashboards.nsjaarverslag.nl/reizigersgedrag
[2]https://www.rijdendetreinen.nl/statistieken/2019

cancellations caused by the disruption. For disruption management, most studies have been using the macroscopic model, omitting details such as signal, block sections, and switches. Due to the reduced complexity of the macroscopic model, a large part of the network can be analyzed, so the overall impact of the disruption can be considered. A recent and promising macroscopic model is the event-activity model, which supports a wide variety of rescheduling measures resulting in practical solutions. Most studies using the event-activity network have focused on the train scheduling step, i.e., determining and changing the arrival and departure times of trains; and, typically, simplify rolling stock circulation by using short-turning at terminal stations, or do not consider rolling stock circulation at all. When rolling stock circulation is not considered, solutions may become infeasible when implemented in real-life, therefore it is important to consider rolling stock circulation in a disruption management model.

In recent years, more studies have integrated passenger-related factors into their network. Disruptions can have significant effects on passengers, as disruptions often affect the passengers' original route to their destination. Furthermore, the implemented rescheduling measures, e.g., delaying and cancelling trains, may have an impact on the passenger route as train may not operate and transfers between train become unfeasible. Many studies have addressed the passenger routing problem by considering passenger rerouting in the model, which more accurately reflects the effect of rescheduling actions on passengers. However, a large increase in model complexity is often necessary to formulate the rerouting problem. Typically, a compromise is made between performance and practicability. In addition to affecting passenger routing, disruptions also have an impact on crowdedness at stations and trains. Especially, with an epidemic situation, e.g., COVID-19, ensuring a safe environment on railway networks has become more essential. Overcrowdedness in rail networks is already a significant issue in urban rail networks (Xu et al., 2014), and has been addressed by several studies by generating timetables based on detailed number of passengers. An accurate model to describe the number of passengers is important for disruption management.

Many passenger-centric models struggle to solve within acceptable time as the resulting mixed-integer linear programming (MILP) problems have a high complexity due to the passenger related variables. To deal with long computation times, most passenger-centric studies include an alternative algorithm to find a good balance between solution time and quality. Adaptive large neighborhood search (ALNS) is a promising and commonly employed algorithm for vehicle routing problems that recently has been introduced for railway scheduling problems. The railway disruption management problem based on the event-activity model has certain similarities to the routing problem. In this paper, we develop a novel disruption management approach based on the event-activity model and the ALNS algorithm.

1-1 Problem statement

In the literature, disruption management models have not considered a detailed number of passengers, making it difficult to accurately determine the crowdedness at stations and trains. Especially with the increase of passengers on railway networks and with the occurrence of epidemic situations, it is important that disruption management models consider a detailed number of passengers. The main research question on passenger-centric railway disruption management is:

> Can a detailed number of passengers formulation be integrated in a railway traffic disruption management problem to efficiently jointly optimize train delays, cancellation, and overcrowdedness at stations?

The main research question can be divided into the following subquestions:

1. How to build a railway management model to incorporate rolling stock circulation and detailed number of passengers in a railway network?
2. How to design an approach to effectively solve the passenger-centric railway disruption management problem?

1-2 Thesis outline

The outline of this thesis is as follows. In Chapter 2, basic knowledge about railway modelling is given. In Chapter 3 the operator-centric model is formulated and an extension is made to rolling stock circulation. In Chapter 4 a passenger-centric model is formulated by extending the operator-centric model with detailed passenger formulations. In Chapter 5 model transformations to MILP formulations are shown, and an ALNS approach is designed. In Chapter 6 the performance of the operator-centric, passenger-centric, and ALNS approaches is tested and compared. In Chapter 7 the discussion, conclusion, and suggestions for future work are given.

Chapter 2

Basic knowledge about railway traffic management

To model a railway network, basic modelling concepts regarding the network infrastructure are required. Additionally, an introduction to different rescheduling measures is necessary to handle disturbances and disruptions on the railway network. The remainder of this chapter is organized as follows. Section 2-1 introduces the different railway network models used in the literature. Section 2-2 explains the difference between disturbances and disruptions. Section 2-3 discusses the different planning steps and control actions to handle disruptions. In Section 2-4 different approaches for passenger integration are shown.

2-1 Network detail

2-1-1 Level of detail

In railway modelling, there are three frequently used detail levels of modelling to describe the railway network, i.e., microscopic models, macroscopic models, and mesoscopic models (see Fig. 2-1) (Albrecht et al. (2008)). Depending on the study objective, one level of detail may be preferred over another. A more in-depth description of each model is provided below.

In the microscopic model, the aim is to include all relevant details, e.g., signals, track segments, and track switches. The microscopic model allows it to develop solutions that are more practical than those found in less detailed models. However, the model is restricted in network size because the model complexity increases rapidly with network size. The microscopic model, for example, can be used to determine accurate train speed profiles, train movement at the level of block sections, or rail signaling.

The macroscopic model neglects most of the railway details. Typically, stations are described as nodes, and lines are the links that connect nodes. Details such as block sections and switches are left out to reduce the computational burden. Due to the lack of detail, more extensive networks can be described with reasonable complexity. The results of the macroscopic model are generally the arrival and departure times of trains at stations and may not guarantee feasibility, as trains could encounter conflicts at more detailed levels, such as conflicting trains at block sections. Therefore, the macroscopic result has to cooperate with a lower-level controller at the microscopic level before being implemented in real-life in most cases.

The mesoscopic model is between the macroscopic model and the microscopic model. The model is primarily described as a macroscopic model with additional details on specific parts, such as the entry tracks of a station. Compared to the macroscopic model, the mesoscopic model has a better chance of being feasible in real-life while maintaining reasonable computational complexity to handle more extensive networks.

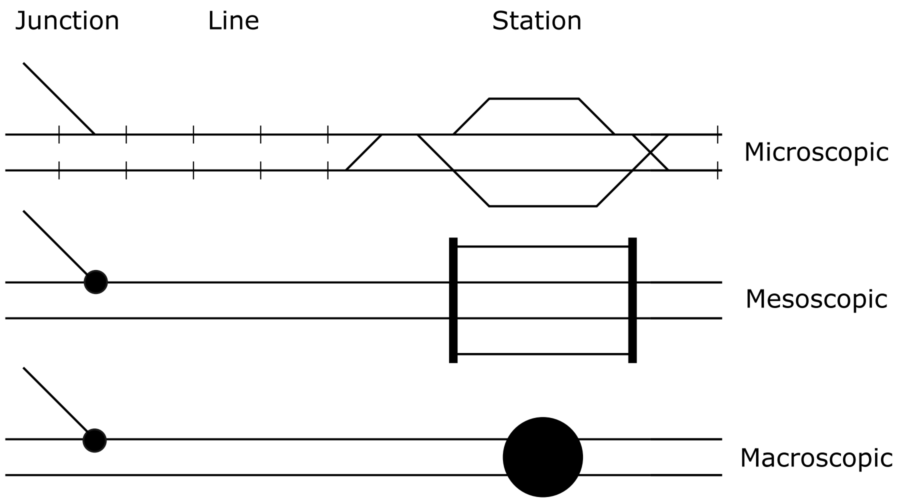


Figure 2-1: Junction, line, and station are shown at the microscopic, mesoscopic, and macroscopic detail levels.

2-1-2 Track layout

Different track layouts can be used to describe railway networks. (Corman and Meng, 2014) The simplest one is a single-track layout, where trains cannot pass or meet. Furthermore, a commonly used layout is the double-track layout, where two parallel tracks are separate. The track itself can be unidirectional or bidirectional. On a unidirectional track, trains can only run in one direction, while on a bidirectional track they can run in both directions. Typically, networks have single-bidirectional or double-unidirectional tracks. Some studies considered more parallel tracks than two parallel tracks, which in general form is written as "N-tracked".

The layout of the track is largely dependent on the type of railway investigated. In the literature on railways, there are mainly two types of railway networks, i.e., urban rail transit networks and interurban rail transit networks. Urban rail transit networks are usually found in large cities. These rail transit lines generally have less freedom of movement because the tracks are used by only one line and the station platform is not shared between different lines. Most urban rail transit networks have unidirectional tracks that are single- or double-tracked. Interurban rail transit networks can be found between cities. With an interurban rail transit network, resources, such as platforms and tracks, can be used between trains with different origins and destinations. Furthermore, the trains can overtake and cross each other. Additionally, urban rail transit networks usually have a high frequency of trains, and passengers can decide ad hoc which train to board. A passenger can have multiple transfers between lines before reaching their destination. Interurban rail transit networks are operated less frequently, and passengers usually plan their routes. Unlike the urban rail transit network, trains do not necessarily stop at every station they pass in an interurban rail network.

The literature on interurban rail transit networks can be divided into two groups, i.e., transportation with seat reservation and without seat reservation. Railway systems with seat reservation have trains with seats allocated for tickets. A passenger has to buy a ticket beforehand, and the train's capacity is limited to the number of seats. Railway systems without seat reservation do not allocate seats for each passenger. The capacity of trains without seat reservation is usually not limited to passenger seats. In this case, they focus on the total capacity of passengers.

2-2 Perturbations

2-2-1 Disturbances and disruptions

Punctuality is essential for passenger satisfaction. Under normal circumstances, trains can arrive at a station on time by running according to a predetermined regular timetable. However, minor unexpected events (e.g., overcrowdedness and bad weather) can occur during operation, which are known as disturbances. Disturbances cause trains to depart or arrive later than planned. Usually, a buffer time is included in the regular timetable such that a train can compensate for the delays, e.g., by speeding up or shortening the dwell times. However, in some serious cases, the delay may be too severe for the buffer time to compensate for the delay, which can result in conflicts with other trains and an infeasible timetable (Cacchiani et al. (2014)).

In the daily operation, more severe unexpected events (e.g., rolling stock breakdowns, station blockages, and extreme weather conditions) can occur, where disturbance management and its rescheduling measures are not effective enough. More decisive measures should be applied to create a new feasible timetable in which not only the timetable but also the rolling stock and crew are rescheduled (Jespersen-Groth et al. (2009)). Generally, two cases of disruption are discussed: complete blockade and partial blockade. A complete blockade means that all tracks in a segment are blocked, leading to an infeasible timetable as trains cannot reach their destination. A partial blockade means that a part of the tracks of a segment are blocked. Thus, trains can still pass through the blockade on parallel tracks with reduced capacity. Typically, the macroscopic model is used for disruption management since disruption affects a large part of the network, and the macroscopic model is able to describe a large part with practical complexity.

2-2-2 Disruption phases

Typically, disruption management can be divided into three phases, i.e., the transition phase from the original timetable to the disruption timetable, the disruption phase, and the transition phase back to the original timetable (Ghaemi et al. (2017)). The first phase is at the beginning of the disruption, and railway traffic will suddenly decrease as some tracks will not be available for train services. In the second phase, the disruption timetable is often applied with a reduced number of trains. The third phase begins after the disruption is resolved, and operations can gradually resume to the original timetable. Especially, the first and third phases are hard to handle in practice, as few instructions are given to train dispatchers on how to deal with these phases. Therefore, it is essential to consider a model that considers all three phases to reduce the workload of the train dispatchers. Many disruption management models in the literature have only considered a part of the disruption phases and not all three of them. However, Ghaemi et al. (2018) have successfully created a disruption rescheduling model that considers all these phases, using a recovery time in which the schedule returns to its original timetable. Zhu and Goverde (2019) have used the same approach with a recovery time after the disruption occurred.

2-3 Railway planning and rescheduling

2-3-1 Railway traffic planning steps

It takes years to make a railway planning for a network. Typically, the process is divided into several steps in order to make them more manageable. For the creation of a regular timetable, these steps consist of demand analysis, line planning, train scheduling, rolling stock planning, and crew scheduling (Ghoseiri et al. (2004), Bussieck et al. (1997)). Demand analysis and line planning are part of strategic planning and are made years in advance and are less relevant for disruption rescheduling or daily operation. Train scheduling, rolling stock planning, and crew scheduling are part of the tactical level and are relevant for the daily operation. In the train scheduling step, the arrival and departure times of the train lines are determined, which results in a timetable. The timetable is then used as input for the rolling stock rescheduling step to assign rolling stock to each train service in the timetable. Finally, the crew rescheduling assigns crew to each rolling stock to ensure each train has its own drivers and conductors. Generally, timetable rescheduling is handled by the infrastructure manager of the railway network, while rolling stock and crew rescheduling are the responsibility of the train operator companies (Cacchiani et al., 2014). Schöbel (2017) argue that many railway planning models apply the planning steps sequentially, resulting in suboptimal solutions compared to an integrated approach. For an urban rail transit line, Wang et al. (2018) has integrated the rolling stock circulation for a train scheduling problem with depot entry and exit actions, adjusting departures and arrival times based on the available rolling stock. For disruption management models, simplified approaches are used for rolling stock circulation. Veelenturf et al. (2016) balance the rolling stock on each side of the disruption, to reduce the possibility of a rolling stock shortage. Louwerse and Huisman (2014) keep track of the number of reserve rolling stock at the start of the second phase of the disruption. In the work of Zhu and Goverde (2020b) and Zhu and Goverde (2019), rolling stock circulation is approached by using short turns at the terminal station, which is called OD-turning.

2-3-2 Rescheduling measures

To handle perturbations, the train dispatchers must adjust the timetable with rescheduling measures so that conflicting trains avoid each other and the delays of the trains are minimized. Generally, three rescheduling measures are applied to resolve conflicts and reduce small delays, i.e., retiming, reordering, and local rerouting. For disruption rescheduling, rescheduling measures are typically extended by cancellations, short-turning, adding stops, and inserting a train actions.

Retiming
Retiming is the process of changing the departure and arrival times of trains in a block section or station (see Fig. 2-2a). In practice, trains are usually delayed when a retiming action is performed, as trains are not allowed to depart earlier than the originally scheduled departure time. An early departure may be confusing for passengers, making it an undesirable rescheduling measure for train operators. An effective retiming action can be to retime a train to dwell longer at a station adjacent to the original conflicting block section. Retiming can also be applied to maintain the connection between trains so that passengers can still make their transfer when delays occur.

Reordering
Reordering refers to the process of changing the order of trains (see Fig. 2-2b). When a faster train is caught behind a slower train, the reordering measure can be quite helpful. By reordering the two successive trains, the faster train can pass the slower train at a stop and reduce the consecutive delay. Furthermore, because departure and arrival times are modified, the reordering measure also implies a retiming measure.

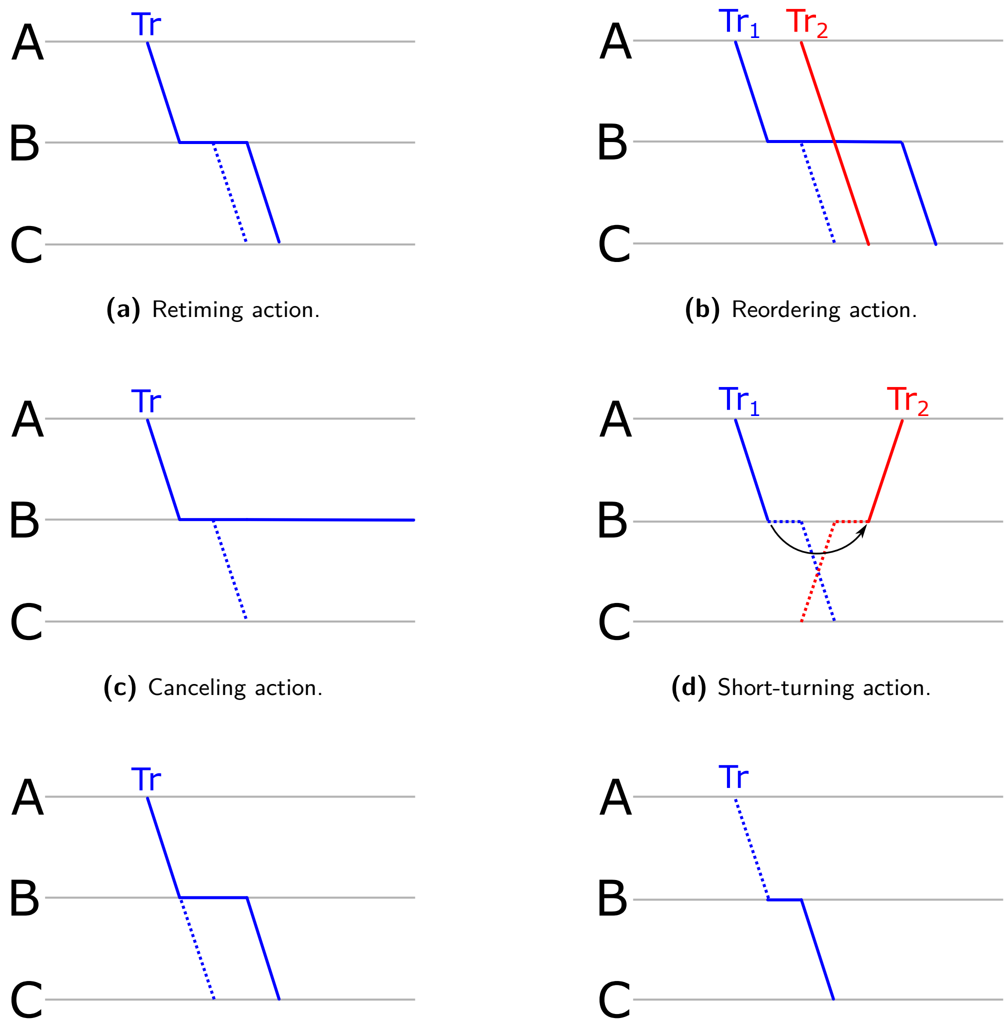
Local rerouting
For conflicting block sections, the local rerouting measure can be used to reroute trains through different (parallel) block sections while maintaining the same destination. Since block sections are used, local rerouting is usually used at the microscopic level.

Cancellation
Cancellation is a frequently used control action for disruptions (see Fig. 2-2c). Instead of starting a train service, the train will not depart from its starting station and remain in the shunting yard.

Short-turning
Short-turning is known as the action of turning a train around at a station and operating the train in the opposite direction (see Fig. 2-2d). If a complete blockade occurs, the short-turning measure is effective in keeping services running on both sides of the disruption. Furthermore, it is also efficient for the rolling-stock circulation at terminal stations. For the vast majority of the papers, short-turning is allowed at the station adjacent to the disruption. As short-turning is time-consuming, many trains will be canceled due to the lack of short-turning capacity at stations. Ghaemi et al. (2018) introduced a short-turning at 2 stations. However, some recent studies have allowed short-turning at all stations with short-turning capability; which is known as flexible short-turning (Zhu and Goverde (2019)).

Adding stop
The adding stop measure allows train services to get additional stops (see Fig. 2-2e). The adding stop measure is possible when trains skip some stations in the original timetable. The

advantages are threefold: first, passengers waiting at the station might have a new option to arrive at their destination. Second, the passenger demand for the following trains may decrease. Moreover, the crowdedness at the station decreases. However, the downside is that onboard passengers may experience a delay, which may result in a more significant delay when a transfer is missed.



(a) Retiming action. **(b)** Reordering action.

(c) Canceling action. **(d)** Short-turning action.

(e) Adding stop action. **(f)** Inserting train action.

Figure 2-2: The different control actions that can be applied.

Insert train

Insert train is the last rescheduling measure (see Fig. 2-2f). When short-turning is not possible due to the lack of arriving trains, an insert train action enables the operator to keep the service in operation by inserting a train from the shunting-yard into the network. Cavone et al. (2020) have introduced a formulation for inserting trains by moving them from and to the shunting yard.

2-4 Passenger integration

Most of the literature on disruption management is operator-centric and does not consider the effect of rescheduling on measures on passengers. Many studies focus on minimizing delays and cancellations. However, in recent years, there has been an increase in passenger-centric disruption management. The way of handling passenger-related factors is highly dependent on the problem and the type of network, i.e., urban railway network and interurban network.

For urban railway networks, the focus has been on accurately calculating the number of passengers in the network, as overcrowdedness at stations and trains is an important issue for these networks (Xu et al., 2014). Especially with an epidemic situation, e.g., COVID-19, resulting in lower available passenger capacity on networks. Wang et al. (2015) implemented passenger arrival rates at stations with a piece-wise affine function. Based on time-varying OD matrices and splitting rates, detailed passenger flows are handled. Instead of using OD-dependent data, Wang et al. (2018) uses time-varying OD-independent arrival rates, which is easier to access compared to OD-dependent data. However, the downside is that passengers' waiting time and onboard time cannot be accurately computed. In addition, the model is restricted to a single line without transfer behaviors. In the work of Yin et al. (2017) time-variant OD data are used to describe passenger demands at stations; however, no further routes need to be determined since there is only one line. Yin et al. (2021) considered time-dependent passenger demand for an urban rail network using a discretized MILP model. Transfer stations and travel paths based on historical data were considered in that paper; however, route choice is not included and the passenger route remains static and deterministic. In the work of Huang et al. (2020) a time-indexed formulation is used for the passenger flows with a discretization of 5 minutes, which is a reasonable time to approximate passenger flows.

For passenger-centric disruption rescheduling, the focus has been more on accurately describing passenger routing, also called timetable-dependent passenger behavior (Zhu and Goverde (2019)). In the work of Veelenturf et al. (2017), a framework with an iterative passenger flow simulation is used to reschedule the timetable and rolling stock. Zhu and Goverde (2019) reduced the complexity of passengers by considering a static passenger route choice, i.e., passenger routes are not affected by rescheduling measures. Zhu and Goverde (2020a) integrated passenger rerouting into the disruption management problem to minimize the generalized travel time of passengers. The passenger activity set is reduced by excluding activities that cannot be made to reduce complexity. Additionally, a transition network is introduced to describe the effects of rescheduling measures on passengers.

2-5 Conclusions

There are three commonly used modelling frameworks for railway networks, i.e., microscopic models, mesoscopic models, and macroscopic models. The models differ in level of detail and usually correspond with the size of the studied network; i.e., macroscopic models are used for large networks, while microscopic models are focused on small networks.

A lot of research has focused on train rescheduling when an unexpected event occurs. Unexpected events can be categorized into disturbances and disruptions. A disturbance is less severe and can be solved with rescheduling measures, e.g., retiming, reordering, and rerouting. Disruptions are more severe and require more rescheduling measures to make a feasible disruption timetable. Cancellations, short-turning, inserting trains, and adding stops are frequently used measures for disruptions. Especially, disruptions put a heavy workload on the train dispatchers as the transition phases of the disruptions are hard to handle. Recent models in the literature have considered these phases to reduce workload.

Passenger integration into railway models has become more popular in recent years. Most of the papers on urban railway networks have focused on accurately calculating the number of passengers in the network. The interurban disruption management models have mainly focused on accurately describing passenger routing.

Chapter 3

Operator-centric modelling including rolling stock circulation

In the event-activity network, the interaction between trains and stations is formulated considering various rescheduling measures. To ensure that rescheduling measures are applied safely without causing conflicts, the event-activity network is implemented with various activities, e.g., headway, running, and dwelling activities. In this chapter, the event-activity network is extended to include rolling stock circulation to ensure sufficient rolling stock is available for operation. Furthermore, shunting actions at intermediate stations are added. The remainder of this chapter is organized as follows. In Section 3-1 the model assumptions are shown and the event-activity network is developed by describing the necessary events and activities for an operator-centric model. In Section 3-2 the operator-centric event-activity network is described with constraints.

3-1 Operator-centric event-activity network

The daily operation of a railway network can be described as an event-activity network. The event-activity network describes the characteristics of the network by defining interactions between trains and stations. The event-activity network is essentially a graph-based formulation, where the nodes consist of events, and the arcs between events represent activities. This section will cover events and activities that are relevant to operator-centric disruption rescheduling, including retiming, cancelation, reordering, and flexible short-turning measures based on the work of Zhu and Goverde (2019). The event-activity network is extended by adding rolling stock circulation with depot entry and exit operations based on the work of Wang et al. (2018). To model rolling stock circulation, inventory events and shunting activities are introduced. In addition, shunting activities are introduced at intermediate stops of a train line to store rolling stock for future use. Modifications are made to the headway activities in order to model quadruple tracks in a network.

3-1-1 Model assumptions

A detailed list of assumptions for the developed railway traffic disruption rescheduling model is given:

1. In this thesis, our aim is to handle the case of a complete blockage between two stations, i.e., all present tracks between two stations are blocked. Furthermore, it is assumed that the duration of the disruptions is known beforehand and has a fixed time duration.
2. The railway network is formulated at the macroscopic detail level. The more detailed train assignment can be further addressed by using local control with a microscopic model, which is not in the scope of this thesis.
3. All train services within a train line use the same rolling stock and cannot use the reserve rolling stock of other train lines. The reserve rolling stock will always be available for operation when stored in the shunting yard.

3-1-2 Events

Events are the nodes of the graph-based event-activity network and are denoted by e. For the passenger-centric model there are three events required, i.e., departure, arrival and inventory events.

Departure and arrival events

There are two events necessary to describe the operation of a train service, i.e., the departure event and the arrival event, which refer to the departures and arrivals of trains at stations, respectively. Each event includes attributes that contain related information about the event. The event attributes can be listed as follows:

- Original scheduled time o_e. Describing the time of event e according to the original timetable without disturbances or disruptions.
- Station s_e. Describing the station where the departure or arrival event e occurs.
- Train service τ_e. Describing the train service associated with event e. A train service describes an individual train from its origin to its destination.
- Train line $\tau_{\text{line},e}$. Describing the train line associated with event e. A train line denotes the route's origin, destination, intermediate stops, and operating frequency. A train line comprises several train services, commonly on a half-hour basis in The Netherlands.
- Train type $\tau_{\text{type},e}$. Describing the train type associated with event e. Train types can be categorized based on their operating speed, e.g., intercity and regional trains.
- Track κ_e. Describing which rail track is used by event e. Railway networks can have multiple parallel tracks to run trains with different operating speeds separately.

- Operation direction d_e. Describing the train running direction, which can be in the up direction or down direction.

The set of departure events is denoted as E_{dep} and the set of arrival events as E_{arr}. The union of departure and arrival events comprises all train service events and is defined as:

$$E_{\text{train}} = E_{\text{arr}} \cup E_{\text{dep}}, \tag{3-1}$$

where E_{train} is the set of train service events.

Inventory events

A line may include a shunting yard with reserve rolling stock at certain stations that can be used in case of rolling stock shortage. To describe the shunting yard in the event-activity network, an inventory event is introduced with the following attributes

- Station s_e. Describing the corresponding station of the shunting yard, where the rolling stock inventory is kept.
- Train line $\tau_{\text{line},e}$. Describing the train line associated with the inventory of the shunting yard. Typically, different rolling stock is used for each train line to accommodate the requirements of the train line, e.g., train capacity and speeds; therefore, it is preferred to only use the rolling stock designated for the train line.
- Inventory number i_e. Describing the number of rolling stock available for use at a shunting yard.

The set of inventory events is denoted by E_{inv}.

3-1-3 Activities

Activities link events by describing the relationships between the events. Activities are denoted by a and pointed from an event e at the tail of the activity (tail(a)) to another event e' at the head of the activity (head(a)). In this thesis, the activities are divided into five groups, i.e., train service, headway, turning, station capacity, and shunting activities.

Train service activities

A train service consists of several departure and arrival events on the route from its origin to its destination. To describe the running, dwelling, and pass-through actions of trains on their route, the following activities are introduced:

- $A_{\text{run}} = \{(e, e')|e \in E_{\text{dep}}, e' \in E_{\text{arr}}, \tau_e = \tau_{e'}, s_{\text{next},e} = s_{e'}\}$ *Running activities* enable trains to traverse between two stations on a track. A running activity is defined between a departure event e and an arrival event e' at the two consecutive stations s_e and $s_{e'}$, i.e., $s_{\text{next},e} = s_{e'}$.

- $A_{\text{dwell}} = \{(e, e')|e \in E_{\text{arr}}, e' \in E_{\text{dep}}, \tau_e = \tau_{e'}, s_e = s_{e'}\}$ *Dwell activities* describe the dwell actions of trains at a station. Tail(a) and head(a) of a dwelling activity should be part of the same train service ($\tau_e = \tau_{e'}$) and occur at the same station ($s_e = s_{e'}$).

- $A_{\text{pass}} = \{(e, e')|e \in E_{\text{arr}}, e' \in E_{\text{dep}}, \tau_e = \tau_{e'}, s_e = s_{e'}\}$ *Pass-through activities* enable trains to pass a station when no stop is required at the station.

Headway activities

A minimum duration between trains should be maintained to ensure that trains operate on the same track safely. These safe distances can be expressed as headway activities. Reordering introduces extra complexity, as multiple candidate headways are considered due to different possible train orders. Headway activities are categorized into departure-departure headway activities, arrival-arrival headway activities, and arrival-departure headway activities. The main difference between these activities is based on the track layout of the stations. In this thesis, we assume the order of trains can only be changed at stations. Stations with multiple tracks offer the ability for trains to overtake, resulting in a change of train order, while stations with a single track do not provide the possibility of overtaking and maintain the order of the previous station. Therefore, different headways are required and formulated as follows:

- $A_{\text{head,de,de}} = \{(e, e')|e \in E_{\text{dep}}, e' \in E_{\text{dep}}, \tau_{\text{type},e} \neq \tau_{\text{type},e'}, d_e = d_{e'}, s_e = s_{e'}, \kappa_e = \kappa_{e'}, N_{s_e} \geq 2\}$ *Departure-departure headways* are constructed between departure events of trains that run on the same track ($\kappa_e = \kappa_{e'}$). The departure events should occur at the same station ($s_e = s_{e'}$) and the station should have at least two tracks ($N_{s_e} \geq 2$), otherwise the activity should belong to $A_{\text{head,ar,de}}$.

- $A_{\text{head,ar,ar}} = \{(e, e')|e \in E_{\text{arr}}, e' \in E_{\text{arr}}, \tau_{\text{next},e} = \tau_{e'}, \tau_{\text{type},e} \neq \tau_{\text{type},e'}, d_e = d_{e'}, s_e = s_{e'}, \kappa_e = \kappa_{e'}, N_{s_e} \geq 2\}$ *Arrival-arrival headways* are constructed between arrival events of trains than run on the same track ($\kappa_e = \kappa_{e'}$). Both events should occur at the same station ($s_e = s_{e'}$) and the station should have two tracks ($N_{s_e} \geq 2$), otherwise the activity should belong to $A_{\text{head,ar,de}}$.

- $A_{\text{head,ar,de}} = \{(e, e')|e \in E_{\text{arr}}, e' \in E_{\text{dep}}, \tau_e \neq \tau_{e'}, \tau_{\text{next},e} = \tau_{e'}, d_e = d_{e'}, s_e = s_{e'}, \kappa_e = \kappa_{e'}, N_{s_e} \geq 1\}$ *Arrival-departure headways* are required between events at stations with a single track $N_{s_e} \geq 1$, or between trains at stations that have designated tracks and can only enter after the previous train has departed.

The set of headway activities can be grouped as

$$A_{\text{head}} = A_{\text{head,de,de}} \cup A_{\text{head,ar,ar}} \cup A_{\text{head,ar,de}}. \tag{3-2}$$

Turning activities

Certain stations enable turning activities of rolling stock to provide train services in the opposite direction. Activities that are part of turning activities can be categorized into two groups, i.e., short-turning and OD-turning activities. Short-turning activities are introduced between events when trains have not arrived at their destination station. Therefore, they must suspend their current service to provide rolling stock for a train service operating in the opposite direction. OD-turning activities are used at terminal stations, where trains start and end their services. The OD-turning activity can be regarded as rolling stock circulation. Short-turning and OD-turning activities can be formulated as follows

- $A_{\text{turn}} = \{(e, e')|e \in E_{\text{arr}}, e' \in E_{\text{dep}}, \tau_e \neq \tau_{e'}, \tau_{\text{line},e} = \tau_{\text{line},e'}, s_e = s_{e'}, d_e \neq d_{e'}, o_{e'} - o_e \leq D\}$ *Short-turning activities* enable rolling stock to short-turn in order to be used by another train service running in the opposite direction. Short-turn activities are constructed between arrival and departure events of the same train line ($\tau_{\text{line},e} = \tau_{\text{line},e'}$) to ensure the same rolling stock is used, as different train lines, typically, use different rolling stock. Short-turning activities events occur at the same station ($s_e = s_{e'}$), and the events should happen in opposite directions ($d_e \neq d_{e'}$). As it is unpractical to excessively wait rolling stock at stations, the difference between arrival and departure events are kept within a maximum delay ($o_{e'} - o_e \leq D$).
- $A_{\text{odturn}} = \{(e, e')|(e, e') = a \in A_{\text{turn}}, e \in E_{terminal}, e' \in E_{terminal}\}$ *OD-turn activities* are short-turning activities that occur at terminal stations and are part of rolling stock circulation.

Station capacity activities

Stations have a certain number of tracks to accommodate dwelling trains. To ensure that the station capacity is not exceeded, capacity activity is constructed between arrival and departure events. The capacity activity can be formulated as

- $A_{\text{cap}} = \{(e, e')|e \in E_{\text{arr}}, e' \in E_{\text{dep}}, \kappa_e = \kappa_{e'}, d_e = d_{e'}, s_e = s_{e'}, N_{s_e,\tau_{\text{type}}} \geq 2, o_{e'} - o_e \leq D\}$ *Capacity activities* are used between arrival and departure events of stations with multiple tracks. This activity is used to keep track of the remaining station capacity.

Shunting activities

At stations with a shunting yard, rolling stock can be transferred from an arrival event to a shunting yard or from a shunting yard to a departure event. Moving rolling stock to a shunting yard allows it to be stored when it is not immediately required. When there is a shortage of rolling stock, the reserved rolling stock can be used from the shunting yard to serve as a departure event. Both activities can be summarized as follows

- $A_{\text{toshunt}} = \{(e, e')|e \in E_{\text{arr}}, e' \in E_{\text{inv}}, \tau_{\text{line},e} = \tau_{\text{line},e'}, s_e = s_{e'}\}$ *Move to shunting yard activities* enables rolling stock to run from arrival events to terminal stations to accompanying shunting yards of the station when rolling stock is no longer required for train services.
- $A_{\text{fromshunt}} = \{(e, e')|e \in E_{\text{inv}}, e' \in E_{\text{dep}}, \tau_{\text{line},e} = \tau_{\text{line},e'}, s_e = s_{e'}\}$ *Move from shunting yard activities* enables rolling stock to run from shunting yards to serve as departure events from terminal stations when additional rolling stock is required.

The set of shunting activities can be grouped as

$$A_{\text{shunting}} = A_{\text{toshunt}} \cup A_{\text{fromshunt}}. \tag{3-3}$$

3-2 Operator-centric constraints

Constraints are established to ensure that the event activity network complies with practicable railway operations to find an optimal disruption timetable subsequently. The constraints are classified primarily according to the activities mentioned in Section 3-1.

3-2-1 Constraints for cancelling and retiming trains

Cancelling and retiming a train service are two effective rescheduling measures to establish a new optimal disruption timetable. A binary decision variable is provided for each departure and arrival event in the event-activity network to determine whether the event e is cancelled. The binary decision variable is denoted as:

$$c_e = \begin{cases} 1 & \text{if event } e \text{ is canceled;} \\ 0 & \text{otherwise.} \end{cases} \tag{3-4}$$

To retime a train service, a continuous variable x_e is introduced for each event e. This variable represents the event's rescheduled occurrence time. The reschedule time is bounded by a minimum and a maximum, which is enforced by:

$$o_e - x_e \leq 0, \qquad \forall e \in E_{\text{arr}} \cup E_{\text{dep}}, \tag{3-5}$$

$$x_e - o_e \leq (1 - c_e)D, \qquad \forall e \in E_{\text{arr}} \cup E_{\text{dep}} \setminus E_{\text{exc}}, \tag{3-6}$$

where o_e denotes the original occurrence time of event e, constant D is the maximum delay allowed, and E_{exc} is the set of events that belong to trains that already departed at the start of the disruption. Constraint (3-5) enforces that the reschedule time of event e is equal to or greater than the initial occurrence time of the event, since trains generally do not depart earlier than their scheduled departure time. Constraint (3-6) imposes a maximum delay time on the reschedule time of event e. A maximum delay is set, as it is not desirable to delay trains excessively. Especially when a cyclic schedule is used, delaying more than one cycle is superfluous since another train service of the same train line is already scheduled to operate. The maximum delay time is not imposed on events in the set E_{exc} because these events are part of trains that already departed the starting station and cannot be cancelled, and may lead to infeasibility.

Given that the disruptions considered in this model are complete blockages, a subset of arrival and departure events that are scheduled at the original timetable on the disruption area are no longer feasible. Some activities will run through the disrupted area, and the related events should be cancelled or retimed to occur after the disruption:

$$x_e \geq t_{\text{end}}(1 - c_e), \quad \forall e \in E_{\text{dep}} : s_e = s_{\text{entry},d_e}, t_{\text{start}} \leq o_e \leq t_{\text{end}}, \tag{3-7}$$

where $s_e = s_{\text{entry},d_e}$ means that the station corresponding to event e should be identical with the station representing the entry station of the disruption in direction $d_e \in \{\text{up}, \text{down}\}$. Constant t_{start} denotes the start time of the disruption, while t_{end} denotes the end time of the disruption. Constraint (3-7) enforces reschedule time x_e to be greater than t_{end} unless the event is cancelled ($c_e = 1$).

To maintain the continuity of train services in the model, the arrival event for a running activity should be cancelled when the corresponding departure event is cancelled, which can be described as:

$$c_{e'} - c_e = 0, \quad \forall (e, e') \in A_{\text{run}}. \tag{3-8}$$

Train service continuity is also essential for station activities, i.e., $A_{\text{station}} = A_{\text{dwell}} \cup A_{\text{pass}}$. When an arrival event is cancelled, the corresponding departure event should also be cancelled:

$$c_{e'} - c_e = 0, \quad \forall (e, e') \in A_{\text{station}} : e \notin A_{\text{turn}} \cap E_{\text{arr}}, e' \notin A_{\text{turn}} \cap E_{\text{dep}}. \tag{3-9}$$

Constraint (3-9) does not apply to events with short-turn activities, as rolling stock from other train services may be short-turned and used to continue the train's service. Therefore, the cancellation of event e does not necessarily result in the cancellation of event e' when an event has a short-turning activity.

The rescheduled time between departure and arrival events should correspond to the minimum duration of the running activity that connects these events. This can be formulated as:

$$x_{e'} - x_e \geq L_a, \quad \forall a = (e, e') \in A_{\text{run}}, \tag{3-10}$$

where L_a is the minimum duration of activity a, which corresponds to the minimum time to traverse a track between stations. A maximum for running activities is set to avoid excessive running times

$$x_{e'} - x_e \leq L_{a,\max}, \quad \forall a = (e, e') \in A_{\text{run}}, \tag{3-11}$$

where $L_{a,\max}$ is the maximum duration of activity a.

For each dwelling activity, there is a minimum duration to ensure that passengers can safely board the train, which can be formulated as

$$x_{e'} - x_e \geq L_a, \quad \forall a = (e, e') \in A_{\text{dwell}}. \tag{3-12}$$

3-2-2 Constraints for each disruption phase

Typically, a disruption can be divided into three phases, i.e., the transition phase from the original timetable to the disruption timetable, the disruption phase, and the recovery phase. Each phase is subject to different constraints, and in this subsection, the constraints related to the transition phase from the original timetable to the disruption timetable and the recovery phase are discussed.

Before disruptions

Prior to a disruption, events should operate according to the original timetable without deviation. Arrival and departure events that occur before disruptions should not be cancelled or delayed

$$c_e = 0, \qquad \forall e \in E_{\text{arr}} \cup E_{\text{dep}} : o_e \leq t_{\text{start}}, \tag{3-13}$$

$$x_e - o_e = 0, \qquad \forall e \in E_{\text{arr}} \cup E_{\text{dep}} : o_e \leq t_{\text{start}}, \tag{3-14}$$

where t_{start} denotes the start time of the disruption.

After disruptions

When a disruption is resolved, the train services should gradually return to their nominal operation, i.e., run according to the original timetable. Therefore, a maximum recovery time constant R is introduced as the upper limit. The end time of a disruption is represented by t_{end}, and when the maximum recovery time is added, the nominal condition should be reached after $t_{\text{end}} + R$. Arrival and departure events that originally occur after this maximum should not be cancelled or delayed, resulting in the following constraints:

$$c_e = 0, \qquad \forall e \in E_{\text{dep}} : o_e \geq t_{\text{end}} + R, \tag{3-15}$$

$$x_e - o_e = 0, \qquad \forall e \in E_{\text{dep}} : o_e \geq t_{\text{end}} + R. \tag{3-16}$$

For events that are scheduled to occur after $t_{\text{end}} + R$, the corresponding arrival event of the same running activity should not be cancelled or delayed:

$$c_{e'} = 0, \qquad \forall (e, e') \in A_{\text{run}} : o_e \geq t_{\text{end}} + R, \tag{3-17}$$

$$x_{e'} - o_{e'} = 0, \qquad \forall (e, e') \in A_{\text{run}} : o_e \geq t_{\text{end}} + R. \tag{3-18}$$

3-2-3 Constraints for headways

To ensure safe operation between trains, sufficient distance between trains, known as headways, should be maintained. To introduce constraints for headways, a binary decision is introduced to define the train order

$$q_{e,e'} = \begin{cases} 1 & \text{if event } e \text{ takes place before event } e'; \\ 0 & \text{otherwise.} \end{cases} \tag{3-19}$$

Depending on the sequence of events e and e', a minimum headway should be maintained between trains running in the same direction. To enforce the minimum headway, the following constraints are introduced

$$x_{e'} - x_e \geq L_a - M(1 - q_{e,e'} + c_e + c_{e'}), \qquad \forall a = (e, e') \in A_{\text{head}}, \tag{3-20}$$

$$x_e - x_{e'} \geq L_a - M(q_{e,e'} + c_e + c_{e'}), \qquad \forall a = (e, e') \in A_{\text{head}}, \tag{3-21}$$

where M is set to be equal to the number of seconds in a day, making it possible to automatically satisfy the constraint with the big-M method. Constraint (3-20) ensures that the minimum headway L_a is respected when event e occurs before event e' (i.e., $q_{e,e'} = 1$). Constraint (3-21) enforces that the minimum headway is maintained when event e' occurs before event e (i.e., $q_{e,e'} = 0$). Both constraints are automatically satisfied when e or e' are cancelled, i.e., $c_e = 1$ or $c_{e'} = 1$.

There are three subsets of headways, i.e., $A_{\text{head,de,de}}$, $A_{\text{head,ar,ar}}$ and $A_{\text{head,ar,de}}$. Activities from the set $A_{\text{head,de,de}}$ are used between departure events at stations where trains can overtake, while activities from the sets $A_{\text{head,ar,de}}$ and $A_{\text{head,ar,ar}}$ are constructed between events where the trains cannot overtake. Train services related to activities of the subset $A_{\text{head,ar,de}}$ cannot overtake as these activities are constructed at stations with a single platform. Overtaking is not possible with activities of the subset $A_{\text{head,ar,ar}}$ as overtaking is not possible on an open track.

By the definition of the sets $A_{\text{head,ar,ar}}$ and $A_{\text{head,ar,de}}$ the order is determined by the preceding flexible headway activity. As a result, the binary decision variable $q_{e'',e'''}$ of activities in $A_{\text{head,ar,de}}$ and $A_{\text{head,ar,ar}}$ should be equal to the preceding flexible activity, denoted by

$$q_{e,e'} = q_{e'',e'''}, \quad \forall (e'', e''') \in A_{\text{head,ar,de}} \cup A_{\text{head,ar,ar}} : (e,e') \in A_{\text{head,ar,de}} \cup A_{\text{head,de,de}} \cup A_{\text{head,ar,ar}}, \tau_e = \tau_{e''}, \tau_{e'} = \tau_{e'''}, s_{\text{next},e} = s_{e''}. \tag{3-22}$$

Constraint (3-22) ensures that the order of activities in the set $A_{\text{head,ar,de}}$ and $A_{\text{head,ar,ar}}$ ($q_{e'',e'''}$) are determined by the order of the previous flexible activity ($q_{e,e'}$). The previous activity can be in the sets $A_{\text{head,ar,de}}$, $A_{\text{head,ar,ar}}$, and $A_{\text{head,de,de}}$, depending on the track layout. The constraints should apply exclusively to activities that are between the same trains, i.e., $\tau_e = \tau_{e''}$ and $\tau_{e'} = \tau_{e'''}$ and occur at stations that are adjacent to each other $s_{\text{next,e}} = s_{e''}$.

3-2-4 Constraints for flexible short-turning

Recall that flexible short-turning enables trains to select short-turning activities at any station that facilitates short-turning. Each departure and arrival event at one of the short-turn stations can have multiple short-turn activity candidates; however, only one short-turn activity can be selected for each event. To determine which short-turn activity is chosen, the following decision variable is introduced:

$$m_a = \begin{cases} 1 & \text{if short-turn activity } a \text{ is selected;} \\ 0 & \text{otherwise.} \end{cases} \tag{3-23}$$

When a short-turning activity is selected, the continuity of train services related to that short-turning activity is changed. At the tail(a) of the short-turning activity, the train service suspends its service to release its rolling stock. At the head(a) of the short-turning activity is a train service that receives rolling stock to continue operating its train service. To maintain consistent train operation, the remaining events of the train service at the tail end of the short-turning activity train service should be cancelled, whereas the train service at the head end of the short-turn activity should have its previous events cancelled.

When a short-turning action is chosen or a train is cancelled, operational consistency can be maintained at arrival events by

$$c_{e'} \leq c_e + \sum_{\substack{a \in A_{\text{turn}},\\ \text{tail}(a)=e}} m_a + z_{a'} \leq 1, \qquad \forall e \in E_{\text{arr}} : (e,e') \in A_{\text{station}}, a' \in A_{\text{toshunt}} \tag{3-24}$$

$$c_{e'} \geq \sum_{\substack{a \in A_{\text{turn}},\\ \text{tail}(a)=e}} m_a + z_{a'}, \qquad \forall e \in E_{\text{arr}} : (e,e') \in A_{\text{station}}, a' \in A_{\text{toshunt}}, \tag{3-25}$$

where $z_{a'}$ is a binary variable for shunting activity, introduced in Section 3-2-6. Constraint (3-24) ensures that when departure event e' is cancelled (i.e., $c_{e'} = 1$), arrival event e is cancelled,

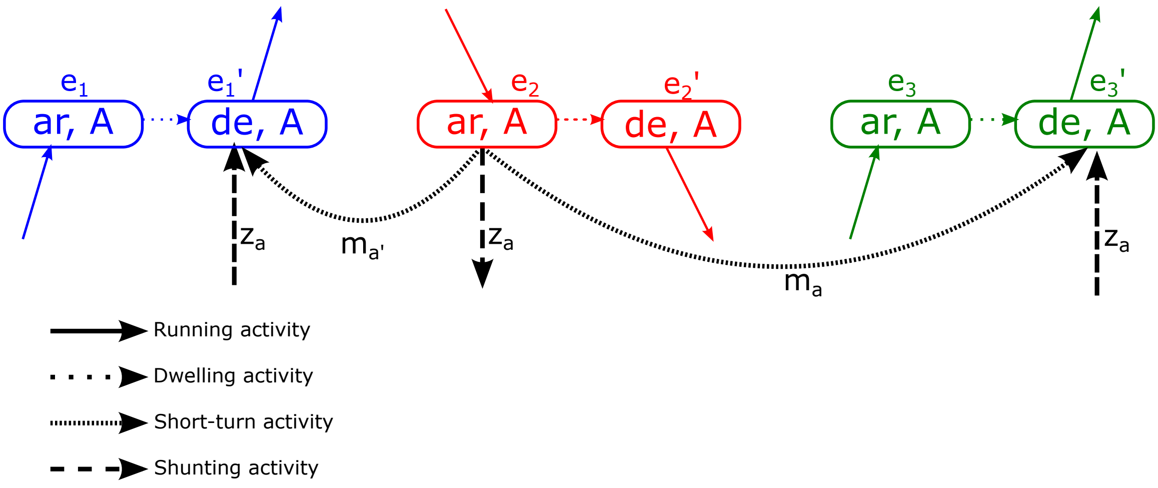


Figure 3-1: Possible short-turning and shunting activities at a station.

short-turned, or moved to a shunting yard. The summation term expresses all short-turning activities that can be selected from the arrival event. Constraint (3-25) ensures that when a short-turn activity is selected, the related departure event e' is cancelled, as there will be no rolling stock available for the departure event.

When a short-turning action is chosen or a train is cancelled, operational consistency should be maintained at departure events by

$$c_e \leq c_{e'} + \sum_{\substack{a \in A_{\text{turn}}, \\ \text{head}(a)=e}} m_a + z_{a'} \leq 1, \qquad \forall e' \in E_{\text{dep}} : (e, e') \in A_{\text{station}}, a' \in A_{\text{fromshunt}} \tag{3-26}$$

$$c_e \geq \sum_{\substack{a \in A_{\text{turn}}, \\ \text{head}(a)=e}} m_a + z_{a'}, \qquad \forall e' \in E_{\text{dep}} : (e, e') \in A_{\text{station}}, a' \in A_{\text{fromshunt}}. \tag{3-27}$$

Constraint (3-26) ensures that when arrival event e is cancelled, departure event e' is cancelled, continues by selecting a short-turning activity, or rolling stock from the shunting yard is selected. The summation term expresses all short-turning activities that can be selected from the departure event. Constraint (3-27) ensures that when a short-turn activity is selected at departure event e', the previous arrival event e is cancelled to prevent the presence of two rolling stocks for the same arrival event.

When a short-turning activity is selected, the related event of the activity should respect the minimum duration of the short-turning action by

$$M(1 - m_a) + x_{e'} - x_e \geq L_a, \quad \forall a = (e, e') \in A_{\text{turn}}, \tag{3-28}$$

Constraint (3-28) enforces that the minimum short-turn duration is respected when the short-turn activity a is selected ($m_a = 1$). When the short-turn activity is not selected ($m_a = 0$), the constraint automatically satisfies the big-M method.

In Fig. 3-1 an example of short-turning and shunting at station 'A' is given. The red train, operating in the down direction, has two short-turn activities, i.e., to the blue train and to the green train, and a shunting activity. Therefore, the red train has four options, i.e., continue the service ($c_{e'_2} = 0$)), short-turn to the green train ($m_a = 1$), short-turn to the blue train ($m_{a'} = 1$) or move to the shunting yard ($z_a = 1$).

3-2-5 Constraints for station capacity

Stations have limitations on the number of trains that can be accommodated concurrently. To ensure that the capacity of the stations is not exceeded, constraints are introduced to limit the number of trains. The constraints are categorized into two groups, i.e., single track and multiple track stations, as these stations require different formulations to describe station capacity.

Singletrack

For stations with a single track for each train type, the correct station capacity is maintained by the $A_{\text{head,ar,de}}$ and $A_{\text{fixed,ar,de}}$ headway activities and their related constraints of Section 3-2-3. These constraints ensure that a subsequent train can only arrive when the previous train has departed from a station, satisfying the station capacity of a single-track station.

Multiple tracks

Different tracks can be available for a train at stations with multiple tracks for each train type. A train can only arrive on a track at a station when the previous train has departed from the same track. To determine whether a track is in use, the following binary variable is introduced:

$$\epsilon_{e,e'} = \begin{cases} 1 & \text{if arrival event } e \text{ occurs before departure event } e'; \\ 0 & \text{otherwise.} \end{cases} \tag{3-29}$$

An arrival event e can occur before or after departure event e', and the related reschedule times x_e, and $x_{e'}$ should obey the order of these events, which can be formulated as:

$$x_e - x_{e'} \geq -M(\epsilon_{e,e'} + c_e + c_{e'}), \qquad \forall (e,e') = a \in A_{\text{cap}} \tag{3-30}$$

$$x_{e'} - x_e \geq -M((1 - \epsilon_{e,e'}) + c_e + c_{e'}), \quad \forall (e,e') = a \in A_{\text{cap}}. \tag{3-31}$$

Recall that A_{cap} is the set of activities constructed between arrival and departure events at stations with multiple tracks for each train type. Constraint (3-30) ensures that x_e occurs after $x_{e'}$ when ($\epsilon_{e,e'} = 0$) and is otherwise automatically satisfied. Constraint (3-31) ensures that $x_{e'}$ occurs after x_e when ($\epsilon_{e,e'} = 1$) and is otherwise automatically satisfied.

The arrival event of earlier operated trains will always occur prior to the arrival event of later operated trains of the same type. For a station with several tracks for each train type, the station capacity can be defined simply by considering whether an earlier train has left the station. The capacity of a station with several tracks can be expressed as follows:

$$\sum_{\substack{a\in A_{\text{cap}}\\ \text{tail}(a)=e}} \epsilon_{e,e'} \leq \chi_{s,\tau_{\text{type}}} - 1, \quad \forall (e,e') = a \in A_{\text{cap}} : s = s_{e'}, \tag{3-32}$$

where $\chi_{s,\tau_{\text{type}}}$ is the station capacity of train type τ_{type}. The left-hand side summation term expresses all departure events e' that are connected by the capacity activity with event e and that have not occurred yet, e.g., another train service is using a platform at the station of event e.

3-2-6 Constraints for rolling stock circulation

Rolling stock is circulated at terminal stations between trains that arrive at their destination station and trains that depart to begin their train service. For train services that are at their origin station, rolling stock is required, while train services at their terminal station must dispose of their rolling stock. Arrival events have multiple ways of disposing of their rolling stock, i.e., by selecting one of the OD-turning candidates or by moving the rolling stock to the shunting yard, such that it can be used for later departure events. Departure events have multiple ways to receive rolling stock, i.e., by selecting one of the OD-turning candidates or by using reserve rolling stock from the shunting yard. To decide which activity is selected for departure and arrival events at terminal stations, the following binary decision variable is introduced:

$$z_a = \begin{cases} 1 & \text{if rolling stock circulation activity } a \text{ is selected;} \\ 0 & \text{otherwise.} \end{cases} \tag{3-33}$$

First, the binary decision variable z_a can be used to formulate the circulation of rolling stock at departure events. Unless train services are cancelled, each departing train service starting at a terminal station requires rolling stock to operate, which can be retrieved by using rolling stock from an arriving train service or from the shunting yard

$$1 - c_{e'} = z_a + \sum_{\substack{a'\in A_{\text{circ}},\\ \text{head}(a')=e'}} z_{a'}, \quad \forall (e,e') = a \in A_{\text{fromshunt}}. \tag{3-34}$$

Constraint (3-34) ensures that when departure event e' is cancelled ($c_{e'} = 1$), no rolling stock circulation activity is selected, as rolling stock is no longer required for the departure event. When the departure event e' is not cancelled ($c_{e'} = 0$), the left-hand side becomes equal to one. Therefore, an OD-turn activity or a move from the shunting yard activity should be selected. On the right-hand side, the variable z_a corresponds to the use of rolling stock from the shunting yard, while the summation expresses the selection of one of the OD-turning candidates at event e.

For arrival events at terminal stations, the same approach can be used as for departure events. Unless an arrival event is cancelled, each arrival event should dispose of its rolling stock by moving it to a departure event or by moving it to the shunting yard

$$1 - c_e = z_a + \sum_{\substack{a'\in A_{\text{circ}},\\ \text{tail}(a')=e}} z_{a'}, \quad (e,e') = a \in A_{\text{toshunt}}. \tag{3-35}$$

Constraint (3-35) ensures that when arrival event e is cancelled ($c_e = 1$), no rolling stock circulation activity is selected since no rolling stock will arrive at the terminal station. When arrival event e is not cancelled ($c_e = 0$), the left-hand side becomes equal to one. Therefore, an OD-turn activity or a move to the shunting yard activity should be selected. On the right-hand side, the variable z_a corresponds to moving rolling stock to the shunting yard, while the summation expresses the selection of one of the OD-turning candidates at event e.

When an OD-turn activity is selected, a minimum time between events is required, as rolling stock cannot be instantly available after arriving at a terminal station. It takes time to make the rolling stock available for the next train service, which can be maintained by

$$x_{e'} - x_e + M(1 - z_a) \geq L_a, \quad \forall (e, e') = a \in A_{\text{odturn}}. \tag{3-36}$$

Constraint (3-36) ensures that the difference between the reschedule time of the departure event $x_{e'}$ and the reschedule time of the arrival event x_e is greater than L_a when an OD-turn activity is selected ($z_a = 1$). When an OD-turn activity is not selected ($z_a = 0$), the big-M method automatically satisfies the time duration constraint.

When retrieving rolling stock from a shunting yard, it is critical to determine whether there is still sufficient rolling stock in the shunting yard. This can be accomplished by counting the shunting yard activities associated with previous events at the same station

$$\sum_{\substack{(e'',e''')=a',\\ a'\in A_{\text{fromshunt}},\\ e'''=e}} \sum_{o_{e''}<o_{e'}} z_{a'} - \sum_{\substack{(e'',e''')=a',\\ a'\in A_{\text{toshunt}},\\ e''=e}} \sum_{o_{e'''}+D<o_{e'}} z_{a'} \leq i_e, \quad \forall (e, e') = a \in A_{\text{fromshunt}}, \tag{3-37}$$

where i_e denotes the number of rolling stocks present in the shunting yard. The first term of (3-37) expresses the number of rolling stocks that moved from the shunting yard to arrival events. Only activities preceding $o_{e'}$ are considered, as shunting yard activities occurring after activity a are irrelevant. The second term expresses the number of rolling stocks that were moved from arrival events to the shunting yard. Only activities up to $o_{e'} - D$ are considered, since arrival events that occur in the time window $(o_{e'} - D, o_{e'})$ are considered by OD-turning activities.

Fig. 3-2 shows an example of the activities involved with rolling stock circulation at a terminal station. Each arrival event at the terminal station can OD-turn to a departure event or move to the shunting yard. For departure events, the options are to retrieve rolling stock with a OD-turn or from the shunting yard.

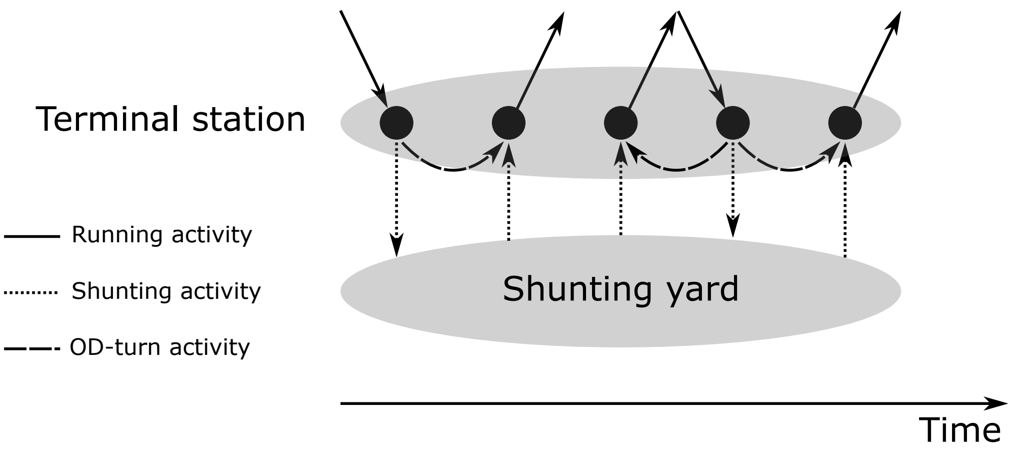


Figure 3-2: Rolling stock circulation with shunting and OD-turn activities.

3-3 Conclusions

An operator-centric event-activity model is proposed to describe the necessary relations to ensure safe and practical operation within a railway traffic network. Three events are introduced, i.e., departure event, arrival event, and inventory event, to represent the departure of a train from a station, the arrival of a train at a station, and the inventory of reserve rolling at stations. To describe the relations between these events, multiple activities are introduced in five categories, i.e., train service activities, headway activities, station activities, turning activities, and shunting activities. An extension to current models is made by including depot entry and exit operations at terminal stations and intermediate stations with a shunting yard, resulting in a more practical solution compared to other event-activity networks.

Chapter 4

Passenger-centric modelling considering train capacity and waiting passengers

In Chapter 3, an operator-centric model is created without considering passengers. To examine the effects of rescheduling measures on passengers, the event-activity network is extended to include events and activities that allow passengers to navigate through the network. An accurate number of waiting passengers is considered by calculating the number of waiting passengers based on the departure times of the trains. In addition, train capacity is considered, denying passengers to board when the train capacity is exceeded. The remainder of this chapter is organized as follows. In Section 4-1 the model assumptions are shown and the operator-centric event activity is extended to accommodate passenger routing through the network. In Section 4-2, the event-activity network is described with constraints.

4-1 Passenger-centric event-activity network

The operator-centric event-activity network of Section 3-1 is extended to include detailed passenger-related factors. To accurately describe passenger movement within the network, new events and activities are introduced based on the work of Zhu and Goverde (2019). A duplicate departure event is introduced to construct waiting, boarding, and transfer activities. A new dummy event is introduced to initialize the number of waiting passengers at the start of the optimization.

4-1-1 Model assumptions

1. The passenger origin-destination demands are assumed to be time-independent, as the arrival rate typically would not change significantly during a relatively short period of the full disruption.

2. Passengers waiting at a station will always choose the first available route and will not skip a train to take a different route. Each OD-pair has access to a maximum of one route per departure event.

3. It may occur that passenger routes are not available because of the disruption. It is assumed that passengers will wait at the station until the disruption is over and will not leave the station.

4. Passenger groups are assumed to be a continuous number instead of an integer number. As the number of passengers can become quite large, the inaccuracy can be ignored.

4-1-2 Events

The network is extended with a duplicate departure event to integrate passengers in the model. The duplicate departure event is used to construct waiting, boarding, and transfer activities. In comparison to the departure event, the duplicate departure event contains an additional property, λ_e, which links the event to the corresponding departure event. The attributes of the departure event can be summed up as follows:

- Original scheduled time o_e. Describing the time of event e according to the original timetable without disturbances or disruptions.
- Station s_e. Describing the station where the duplicate departure event e occurs.
- Train service τ_e. Describing the train service associated with event e. A train service is the run of a train from the origin to the destination.
- Train line $\tau_{\text{line},e}$. Describing the train line associated with event e. A train line denotes the origin, destination, intermediate stops, and operating frequency of the route. A train line consists of several train services.
- Train type $\tau_{\text{type},e}$. Describing the train type associated with event e. Typically, train types can be categorized based on their operating speed.
- Track κ_e. Describing which railway track is used by event e. Railway networks can have multiple parallel tracks to separately run trains with different operating speeds.
- Operation direction d_e. Describing the train running direction, which can be in the up direction or down direction.
- Related duplicate departure event λ_e. Describes the departure event of which this duplicate is a copy.

The set of duplicate departure events is denoted as E_{dde}.

A dummy event for each station is introduced to initialize the number of waiting passengers at the first departure event within the time window. The dummy event has the following attributes:

- Scheduled time o_e. Describing the time of the dummy event e. The time corresponds with the start time of the optimization.
- Station s_e. Describing the station where the event occurs.

The set of dummy events is denoted as E_{dum}.

4-1-3 Activities

Activities are constructed between two different events as a directed arc. To accommodate passenger behaviour, the following activities are formulated

- $A_{\text{boarding}} = \{(e, e')|e \in E_{\text{dde}}, e' \in E_{\text{dep}}, \lambda_e = e'\}$. *Boarding activities* allow passengers to board the train corresponding to event e.
- $A_{\text{transfer}} = \{(e, e')|e \in E_{\text{arr}}, e' \in E_{\text{dep}}, o_{e'} - o_e \geq L_a, \tau_e \neq \tau_{e'}, s_e = s_{e'}\}$ *Transfer activities* allow passengers to transfer from one train service to another train service at a station.
- $A_{\text{waiting}} = \{(e, e')|e \in E_{\text{dum}}, e' \in E_{\text{dum}}, o_{e'} - o_e \leq D\}$ *Waiting activities* allow passengers to wait for another train service at their origin station.

4-2 Passenger-centric constraints

The extension of the passenger-centric event-activity network of Section 4-1 describes passengers' behaviours in the network through various activities, i.e., boarding activities, transfer activities, and waiting activities. In this Section, the event-activity network is used to formulate passenger routes. A new formulation based on the work of Wang et al. (2015) is introduced to accurately calculate the number of waiting passengers at stations using the reschedule time of departure events. In addition, a formulation is introduced to deny passengers to board a train when the train capacity is exceeded.

4-2-1 Passenger routing

The routes of passengers are required to accurately describe the impact of rescheduling measures on passengers. Passenger routes can be constructed by using origin-destination (OD) demands. The OD-demands are assumed to be time-independent during a disruption, as no significant changes in arrival rates are expected as we consider full disruption during a certain period, the passenger rate typically does not change significantly during a relatively short period. In this thesis, passengers with the same origin and destination are regarded as one group, and each passenger group g contains the following attributes:

$$(O_g, D_g), \tag{4-1}$$

where O_g is the origin of passenger group g and D_g is the destination of passenger group g. The set of passenger groups is defined as G.

For each passenger group, a route is defined between the origin and destination. A route is constructed between the duplicate departure event and the arrival event at the destination of the passenger group. Typically, the route for each passenger group is pre-given and can be determined by using historical data. The set of activities that are taken by the passenger group is defined by $A_{g,e}$, where e denotes the duplicate departure event at which the group starts.

Some passenger groups will have a transfer on their route, resulting in two subsets of activities, i.e., pre-transfer activities and after-transfer activities. Since a transfer means that passengers will take two trains, it could be possible that only the pre-transfer activities are available. To account for this, we have these two subsets.

Due to the disruption and the resulting rescheduling measures, some routes might become infeasible. A binary decision variable is introduced to define whether the route of group g starting at duplicate departure event e is available

$$u_{g,e} = \begin{cases} 1 & \text{if the route of group } g \text{ starting at event } e \text{ is available;} \\ 0 & \text{otherwise.} \end{cases} \tag{4-2}$$

Passenger routes can only be available when all corresponding activities of the route are feasible. Activities can only be feasible when the corresponding events, i.e., tail(a) and head(a), are not cancelled

$$u_{g,e} \leq (1 - c_e), \qquad \forall (e, e') = a \in A_{g,e}, \forall g \in G, \forall e \in E_{\text{dde}}, \tag{4-3}$$

$$u_{g,e} \geq \sum c_e - (N_{g,e} - 1), \quad \forall (e, e') = a \in A_{g,e}, \forall g \in G, \forall e \in E_{\text{dde}}, \tag{4-4}$$

where $N_{g,e}$ is the number of activities of $A_{g,e}$. Constraint (4-3) ensures that a passenger route becomes unavailable (i.e., $u_{g,e} = 0$) when one of the events on the route is cancelled ($c_e = 1$). Constraint (4-4) ensures that the route is available (i.e., $u_{g,e} = 1$) when no activities of the passenger route are cancelled.

Note that in this thesis, the activities of $A_{g,e}$ that determine the variable $u_{g,e}$ consist of the activities until a transfer. This will allow a passenger group to board a train and wait at the transfer station, even when the connecting train is not available.

4-2-2 Waiting passengers

There are two key components to calculating the number of waiting passengers at each station, i.e., the departure of trains and the passenger arrival rates. When a departure event occurs, the number of waiting passengers will change depending on the number of boarding and transferring passengers. Between two departure events, waiting passengers are accumulated according to the arrival rate. Assuming that the arrival rate is constant, the number of waiting passengers in each group immediately before a departure event can be calculated as follows

$$w_{\text{before},g,e} = w_{\text{after},g,e'} + \zeta_g (x_e - x_{e'}) + n_{\text{trans},g,e}, \quad \forall g \in G, e \in E_{\text{dde}} \tag{4-5}$$

where $x_{e'}$ is the occurrence time of previous event e'; ζ_g is the number of passengers arriving per second for the passenger group g, variable $w_{\text{after},g}(x_{e'})$ is the number waiting passengers immediately after the previous e', and $n_{\text{trans},g,e}$ is the number of transferring passengers.

The total number of waiting passengers before each departure event e can be calculated by the summation of waiting passengers in each passenger group:

$$w_{\text{before},e} = \sum_{g \in G} w_{\text{before},g,e}, \quad \forall e \in E_{\text{dde}}. \tag{4-6}$$

When a departure event occurs, some passengers board the train to travel to their destination. The number of passengers wanting to board a train is denoted by the continuous variable $\eta_{\text{want},g,e}$. The number of passengers wanting to board a train is constrained by the availability of their preferred route:

$$\eta_{\text{want},g,e} \leq w_{\text{before},g,e} + M u_{g,e}, \quad \forall g \in G, \forall e \in E_{\text{dde}}, \tag{4-7}$$
$$\eta_{\text{want},g,e} \geq w_{\text{before},g,e} - M u_{g,e}, \quad \forall g \in G, \forall e \in E_{\text{dde}}, \tag{4-8}$$
$$\eta_{\text{want},g,e} \geq 0, \quad \forall g \in G, \forall e \in E_{\text{dde}}, \tag{4-9}$$
$$\eta_{\text{want},g,e} \leq M u_{g,e}, \quad \forall g \in G, \forall e \in E_{\text{dde}}, \tag{4-10}$$

where $\eta_{\text{want},g,e} = 0$ when the passenger route is not available ($u_{g,e} = 0$). Constraints (4-7) and (4-8) ensure that variable $\eta_{\text{want},g,e}$ is equal to $w_{\text{after},g,e}$ when the route is available ($u_{g,e} = 1$). When the route is not available ($u_{g,e} = 0$), the constraints are automatically satisfied. Constraint (4-9) ensures that the number of passengers who want to board is not negative. Constraint (4-10 ensures that the number of passengers who want to board a train is zero when their route is not available.

The total number of boarding passengers at departure event e can be calculated by the sum of all boarding passenger groups

$$\eta_{\text{want},e} = \sum_{g} \eta_{\text{want},g,e}, \quad \forall g \in G, \forall e \in E_{\text{dde}}. \tag{4-11}$$

When a train is near its maximum passenger capacity, not all passengers who want to board the train can board the train. The variable $\eta_{\text{can},e}$ is introduced to represent the number of passengers that can board the train. The number of passengers that can board the train is equal to the minimum of the number of waiting passengers and the remaining space of the train

$$\eta_{\text{can},e} = \min(\eta_{\text{want},e}, n_{\text{remain},e}), \quad \forall e \in E_{\text{dde}}, \tag{4-12}$$

where $n_{\text{remain},e}$ is the remaining passenger capacity of the train at event e.

The number of passengers per group that can board a train is denoted by $\eta_{\text{can},g,e}$, and the number of passengers that can board a train at a departure event can be summed up by all groups as

$$\eta_{\text{can},e} = \sum_{g \in G} \eta_{\text{can},g,e}, \quad \forall e \in E_{\text{dde}}. \tag{4-13}$$

When a departure event occurs, some waiting passengers board the train, and the number of waiting passengers after the boarding process is:

$$w_{\text{after},g,e} = w_{\text{before},g,e} - \eta_{\text{can},g,e}, \quad \forall g \in G, \forall e \in E_{\text{dde}}. \tag{4-14}$$

The remaining capacity of a train depends on the total capacity of the train, the number of alighting passengers, and the number of passengers on board, which can be described by

$$n_{\text{remain},e} = C - n_{\text{after},e'} + n_{\text{alight},e}, \quad \forall e \in E_{\text{dde}}. \tag{4-15}$$

where C is the maximum train capacity; $n_{\text{after},e'}$ is the passengers on board of the train after its previous departure event e' and $n_{\text{alight},e}$ is the number of alighting passengers.

The number of alighting passengers can be calculated by determining the passenger groups with a transfer or destination at the station of the current event by

$$n_{\text{alight},e} = \sum_{e' \in \tau_{\text{pre}}} \sum_{g \in G_{\text{alight},e'}} \eta_{\text{can},g,e'}, \quad \forall e \in E_{\text{dde}}, \tag{4-16}$$

where $\tau_{\text{pre},e}$ denotes the preceding events e' of the corresponding train and $G_{\text{alight},e}$ denotes the set of passenger groups with a transfer or destination at event e. Constraint (4-16) calculates the total number of alighting passengers at event e by summing up the total number of boarded passengers groups at previous events e' of the corresponding train service with a transfer or destination at station e.

The number of passengers on board a train after a departure can finally be described by

$$n_{\text{after,e}} = n_{\text{after},e'} - n_{\text{alight},e} + n_{\text{can},e}, \quad \forall e \in E_{\text{dde}}. \tag{4-17}$$

Using the number of waiting passengers, the maximum congestion level (Yin et al. (2021)) can be determined at each station, denoted as ξ_s. The congestion level can be determined by

$$\xi_s \geq w_{\text{before},e}/S_{\text{cap},s}, \quad \forall s \in S, \forall e \in E_{\text{dde}} : s = s_e \tag{4-18}$$

where $S_{\text{cap},s}$ is the station capacity of station s and $w_{\text{total},e}$ is the number of waiting passengers before each event. The set S denotes all stations.

4-3 Conclusions

In this chapter, a passenger-centric event-activity network is proposed to determine an accurate number of waiting passengers considering the capacity of trains. First, the event-activity network is extended with two events, i.e., a duplicate departure event and a dummy event. The duplicate departure event is used to support the boarding and transferring of passengers. Boarding, waiting, and transfer activities are constructed between departure and duplicate departure events in the event-activity network to allow passengers to navigate through the network. By using the passenger-centric event-activity network, a routing formulation is developed to determine if passenger routes are still feasible with the considered rescheduling measures. A new detailed waiting passenger formulation is proposed to extend the passenger-centric model by considering the effects of the reschedule time on the number of passengers while considering train capacity.

Chapter 5

Solution methods including an ALNS algorithm

The passenger-centric railway disruption management problem based on the event-activity model presented in Chapters 3 and 4 can be expressed as a mixed-integer linear programming (MILP) problem to determine a new timetable when disruptions occur. An MILP formulation requires linear constraints containing continuous and integer variables. The operator-centric problem already consists of linear constraints, continuous variables, and integer variables. Therefore, it can be solved directly. The passenger-centric constraint problem requires transformations before it can be solved as an MILP problem. A formulation is required to determine the train order at stations to calculate the detailed number of passengers. Inspired by the well-researched vehicle routing problem (VRP)(Adewumi and Adeleke (2018)), we handle train ordering in the formulation by the VRP flow constraints, so that many advanced solution approaches in VRP can be used. The resulting optimization problem can be transformed into an MILP problem. However, the passenger-centric MILP formulation introduces a significant number of binary variables, which makes the MILP problem hard to solve. Adaptive large neighborhood search (ALNS) is a widely used metaheuristic for solving VRPs, which is capable of balancing solution time and quality with destroy and repair operators. In this section, the ALNS algorithm is adapted to be compatible with the passenger-centric model, and a railway disruption management algorithm is developed.

The remainder of this chapter is organized as follows. In Section 5-1 two objective functions are introduced for the operator-centric and passenger-centric model. In Section 5-2 the requirements for an MILP formulation are shown. In Section 5-2-4 the MILP solution method is introduced for the operator-centric and passenger-centric model. In Section 5-3 an ALNS metaheuristic is proposed for the transformed passenger-centric model to obtain faster solution times.

5-1 Problem statement

An optimization problem requires an objective function that quantifies the performance of the obtained solution. For the railway traffic disruption rescheduling problem, the optimization can have different objective functions depending on the goal of the user. In this thesis, two objective functions are considered, i.e., for the operator-centric and the passenger-centric model. The operator-centric objective function minimizes train delays and cancelations, ensuring a minimum deviation from the original timetable. The passenger-centric objective function extends the passenger-centric objective function by also considering the station capacity, improving passenger satisfaction.

The operator-centric objective function is formulated as follows

$$J = \sum_{e \in E_{\text{arr}}} \Psi_{\text{delay}}(x_e - o_e) + \sum_{e \in E_{\text{arr}}} \Psi_{\text{cancel}} c_e, \tag{5-1}$$

$$\text{s.t.} \quad \text{constraints (3-5) - (3-37)} \tag{5-2}$$

where w_{delay} is the weight related to the reschedule time for each arrival event, and w_{cancel} is the weight of the binary cancellation variable for each event. With the weights, a trade-off can be made between the number of cancellations and delays. Adding a high weight to the cancellation weight w_{cancel} will favor a cancellation above a delay when the delay reaches a certain threshold (Zhan et al., 2015).

The passenger-centric objective function is formulated as follows

$$J = \sum_{e \in E_{\text{arr}}} \Psi_{\text{delay}}(x_e - o_e) + \sum_{e \in E_{\text{arr}}} \Psi_{\text{cancel}} c_e + \sum_{\forall s \in S} \Psi_{\text{sc}} \xi_s, \tag{5-3}$$

$$\text{s.t.} \quad \text{constraints (3-5) - (3-37) and (4-1) - (4-17)} \tag{5-4}$$

where Ψ_{sc} is the weight of the maximum station capacity for each station. A higher value for the station capacity weight Ψ_{sc} will result in reduced congestion levels at stations; however, it will decrease operator-centric performance as more cancellations and delays occur. The optimization should satisfy the constraints of (5-4) consisting of the constraints introduced in the operator-centric model of Section 3-2 and the passenger-centric model of Section 4-2.

5-2 Mixed-integer linear programming transformations

In this section, two transformations are introduced to be able to directly solve the passenger-centric problem, i.e., the transformation of event orders and the transformation of min function (4-12). The order of events is handled by the general idea of VRP. The min function is transformed into linear inequialities by using the method in Bemporad and Morari (1999). Finally the problem can be transformed into an MILP problem.

5-2-1 Introduction to VRP

The goal of the vehicle routing problem (VRP) is to connect nodes or customers in the most efficient manner to minimize the objective function, e.g., minimization of distance or time.

The VRP approach has the advantage of being extensively studied, making it possible to find efficient algorithms to solve the problem (Adewumi and Adeleke (2018)). The general VRP can be formulated as

$$\min z = \sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij}x_{ij}$$

subject to

$$\sum_{j=1}^{n} x_{ij} = 1, \qquad \forall i \in \{1, 2, \ldots, n\} \tag{5-5}$$

$$\sum_{i=1}^{n} x_{ij} = 1, \qquad \forall j \in \{1, 2, \ldots, n\} \tag{5-6}$$

$$x_{ij} \in \{0, 1\}$$

where d_{ij} is the distance from i node to j node. The binary variable x_{ij} determines whether the route from i to j is selected. The objective function is denoted as z, and in the generalized case minimizes the total distance between the nodes. Constraints (5-5) and (5-6) ensure that each node is visited once. These constraints can be used to describe the event order at stations. For each station, the order of events can be regarded as a vehicle routing problem (VRP), in which each duplicate departure event is regarded as a node, and the goal is to connect the departure events in the most efficient manner to minimize the objective function.

An issue that can arise when optimizing a VRP is the generation of subtours, in which multiple closed routes are generated instead of a single route. To eliminate subtours, Miller et al. (1960) introduced an approach in which time variables are assigned to each visited node. A node can only be visited if the previous node has a lower time; hence, it is impossible to form subtours. As station events already have assigned time variables, this approach is suitable for station event ordering to eliminate subtours, as no significant adjustments are required.

5-2-2 Station event ordering

Recall that in (4-5) a formulation was introduced to calculate the number of waiting passengers before each event by using the time of the previous event x_e. In the work of (Wang et al. (2015)) this formulation can be used directly, as there is no reordering in their urban railway network. Using a predetermined order of events, the preceding event is known and can be expressed as a linear constraint. To accurately describe the number of waiting passengers in an event-activity network with reordering, a linear formulation is required to describe the order of events at each station to determine which event precedes which event.

In this thesis, a binary decision variable is introduced to determine whether the event e is the preceding event of event e' to represent the order of the events at each station:

$$\alpha_{e,e'} = \begin{cases} 1 & \text{if event } e \text{ is the previous occurring event of } e' \text{ at the same station} \\ 0 & \text{otherwise.} \end{cases} \tag{5-7}$$

The variable $\alpha_{e,e'}$ is constructed between each duplicate departure event at each station.

To enforce the definition of the binary variable $\alpha_{e,e'}$ in (5-7) the reschedule times must satisfy

$$x_e < x_{e'} + M(1 - \alpha_{e,e'}), \quad \forall e, e' \in E_{\text{dde}} \cup E_{\text{dum}} : s_e = s_{e'}. \tag{5-8}$$

Constraint (5-8) ensures that event e occurs before event e' when $\alpha_{e,e'} = 1$. Otherwise, e is not the previous event of e' and the constraint is automatically satisfied with the big-M method. The constraint is expressed as less than ($<$) instead of less than or equal ($\leq$) to ensure no subtours are created.

Fig. 5-1 shows an example with 3 examples with the usage of variable $\alpha_{e,e'}$. The horizontal axis represents time. Therefore, $\alpha_{e,e'} = 1$, as e is the previous event of event e', and $\alpha_{e',e''} = 1$, as e' is the previous event of e''.

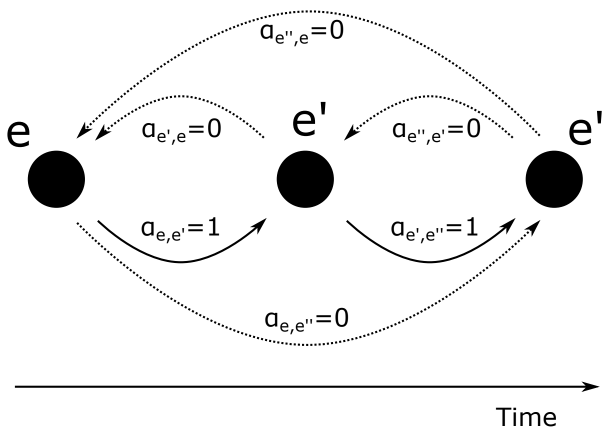


Figure 5-1: Event ordering of three events.

The general VRP formulation states that a tour must be closed, i.e., the starting node must also be the ending node. As the station event order is in the time domain, the tour cannot be completed as the ending node cannot occur before the starting node. Therefore, the formulation is adjusted to fit an open tour. The event order at each station can be found by determining for each event e' which event is the previous event e. Except for the dummy event, each event e' has a previous event that occurred. The summation of binary decision variables satisfies

$$\sum_{e,s_e=s} \sum_{e',s_{e'}=s} \alpha_{e,e'} = \Pi_{s_e}, \quad \forall s \in S \tag{5-9}$$

where Π_{s_e} is the number of duplicate departure events at station s_e and S is the set of stations.

Each event has at most one previous and one following event and satisfies

$$\sum_{e \in E, e \neq e', s_e = s_{e'}} \alpha_{e,e'} \leq 1 \quad e' \in E_{\text{dde}}, \tag{5-10}$$

$$\sum_{e' \in E, e \neq e', s_e = s_{e'}} \alpha_{e,e'} \leq 1 \quad e \in E_{\text{dde}}. \tag{5-11}$$

Constraint (5-10) ensures that events have one single previous event. Constraint (5-11) ensures that events have one single subsequent event. To account for the first and last events that occur, both constraints are set as an inequality.

The event order also affects the calculation of waiting passengers, as variable $w_{\text{before},g,e'}$ is dependent on the variable $w_{\text{after,e}}$ of the previous event. To account for the different event orders, the continuous variable $w_{\text{before},g,e,e'}$ is introduced to represent the number of passengers waiting for event e' after event e. The number of waiting passengers before event e' can be formulated as:

$$
\begin{aligned}
&w_{\text{before},g,e,e'} \leq M\alpha_{e,e'}, && \forall e, e' \in E_{\text{dde}} : s_e = s_{e'}, && \text{(5-12)}\\
&w_{\text{before},g,e,e'} \geq 0, && \forall e, e' \in E_{\text{dde}} : s_e = s_{e'}, && \text{(5-13)}\\
&w_{\text{before},g,e,e'} \leq w_{\text{after},g,e} + \zeta_g(x_e - x_{e'}) + M(1-\alpha_{e,e'}), && \forall e, e' \in E_{\text{dde}} : s_e = s_{e'}, && \text{(5-14)}\\
&w_{\text{before},g,e,e'} \geq w_{\text{after},g,e} + \zeta_g(x_e - x_{e'}) - M(1-\alpha_{e,e'}), && \forall e, e' \in E_{\text{dde}} : s_e = s_{e'}. && \text{(5-15)}
\end{aligned}
$$

Constraint (5-12) ensures that $w_{\text{before},g,e,e'} = 0$ when the corresponding event e is not the previous event of e', i.e., $\alpha_{e,e'} = 0$. Constraint (5-13) ensures that $w_{\text{before},g,e,e'}$ is equal to or greater than zero, as the waiting passenger cannot be negative. Constraints (5-14) and (5-15) ensure that $w_{\text{before},g,e,e'}$ is equal to the number of waiting passengers and arriving passengers when event e' succeeds e, i.e., $\alpha_{e,e'} = 1$.

Finally, the total number of waiting passengers before event e' can be determined by

$$w_{\text{before},g,e} = \sum_{e \in E_{\text{dde}}, s_e = s_{e'}} w_{\text{before},g,e,e'}, \quad \forall e' \in E_{\text{dde}}. \tag{5-16}$$

Constraint (5-16) sums all possible $w_{\text{before},g,e,e'}$ to account for the different possible event orders. According to (5-12) - (5-15), one variable $w_{\text{before},g,e,e'}$ will contribute to the total, while the rest will be zero.

5-2-3 Transformation of min function

Another non-linear term is introduced when calculating the number of boarding passengers in (4-12):

$$\eta_{\text{can},e} = \min(\eta_{\text{want},e}, n_{\text{remain},e}), \quad \forall e \in E_{\text{dde}}, \tag{5-17}$$

A linearization of the minimization of boarding passengers (4-12) is required to formulate an MILP problem. A binary decision variable is introduced to define whether $w_{\text{before,e}}$ is smaller than $n_{\text{remain},e}$.

$$y = \begin{cases} 1 & \text{if } \eta_{\text{want},e} \text{ is smaller than } n_{\text{remain},e}; \\ 0 & \text{otherwise.} \end{cases} \tag{5-18}$$

When $y = 1$, $\eta_{\text{want},e}$ should be smaller than $n_{\text{remain},e}$, while $n_{\text{remain},e}$ should be smaller than $\eta_{\text{want},e}$ when $y = 0$. The following constraints are used to enforce the definition of y:

$$
\begin{aligned}
&n_{\text{remain},e} - \eta_{\text{want},e} \leq My, && \forall e \in E_{\text{dde}}, && \text{(5-19)}\\
&\eta_{\text{want},e} - n_{\text{remain},e} \leq M(1-y), && \forall e \in E_{\text{dde}}. && \text{(5-20)}
\end{aligned}
$$

To ensure that $\eta_{\text{can},e}$ is equal to the minimum of $n_{\text{remain},e}$ and $\eta_{\text{want},e}$

$$\eta_{\text{can},e} \leq \eta_{\text{want},e}, \quad \forall e \in E_{\text{dde}}, \tag{5-21}$$
$$\eta_{\text{can},e} \leq n_{\text{remain},e}, \quad \forall e \in E_{\text{dde}}, \tag{5-22}$$
$$\eta_{\text{can},e} \geq \eta_{\text{want},e} - M(1-y), \quad \forall e \in E_{\text{dde}}, \tag{5-23}$$
$$\eta_{\text{can},e} \geq n_{\text{remain},e} - My, \quad \forall e \in E_{\text{dde}}. \tag{5-24}$$

Constraints (5-21) and (5-22) ensure that $\eta_{\text{can},e}$ will never be larger than $w_{\text{before},e}$ and $\eta_{\text{want},e}$. Constraints $(5-23)$ and $(5-24)$ ensure that $\eta_{\text{can},e}$ is greater than or equal to $w_{\text{before},e}$ and $\eta_{\text{want},e}$ depending on the binary decision variable y.

5-2-4 MILP formulation

The MILP formulation is a preferred formulation for railway disruption management problems (Cacchiani et al. (2014)), as MILP approaches provide fast computational times and can be solved efficiently by using existing solvers, compared to, e.g., non-linear formulations. In standard form, the MILP model can be described as

$$\begin{aligned} \min_{x} \quad & c^T x \\ \text{s.t.} \quad & Ax \leq b \\ & x \geq 0 \\ & x_i \in \mathbb{Z}, \quad i \in \{1, \ldots, n_{\text{b}}\} \\ & x_j \in \mathbb{R}, \quad j \in \{n_{\text{b}}+1, \ldots, n_{\text{b}}+n_{\text{r}}\} \end{aligned} \tag{5-25}$$

where c is a constant weight vector and x is the variable vector, consisting of n_{b} integer (or binary) variables and n_{r} continuous variables. The coefficients of the inequality constraints are contained in the matrix A. Equality constraints can be preserved by rewriting them as inequality constraints.

Combining the operator-centric objective with the operator-centric constraints, the passenger-centric MILP problem can be solved directly, since the constraints introduced in Section 3-2 are formulated as linear constraints. To solve the passenger-centric model as an MILP problem, the station event ordering and the min function transformation are required.

5-3 ALNS algorithm

The MILP solution method provides a way to solve the railway disruption problem directly and accurately. For small-scale networks, the MILP approach can be applied practically; however, for large-scale networks, the number of binary variables grows exponentially, resulting in long solution times and making the MILP method not suitable to use in real-time. Therefore, we seek a solution method that efficiently addresses the discrete combination optimization problem. Ropke and Pisinger (2006) introduced adaptive large-scale neighborhood search (ALNS) as an extension of the large neighborhood search heuristic for a VRP. ALNS can efficiently explore a large solution space with properly designed destroy and repair operators. Since the formulation introduced to handle the event ordering at stations (Sec. 5-2-4) is based on the VRP formulation, the ALNS algorithm is a suitable option. A part of the literature has focused on the application of the ALNS algorithm to railway scheduling problems. In the work of Yin et al. (2021), ALNS is applied effectively to a rail transit network scheduling problem in which the emphasis is on coordinating the passenger flows of different train lines to reduce crowding. Dong et al. (2020) applied the ALNS algorithm with a wide variety of operators to a train stop planning problem. Their method handles a mixed-integer non-linear programming problem. In this section, an adaptation is made to the destroy and repair algorithms so that they can be applied to the disruption rescheduling problem.

5-3-1 Introduction

ALNS is developed based on the large neighborhood search (LNS) algorithm (Shaw (1998)), which explores the solution space using a destroy and repair operator. A destroy operator destroys a portion of a feasible solution, resulting in an infeasible solution that requires a repair operator to become feasible again. The destroy operators are responsible for the exploration of the solution space, while the repair operator tries to find the best solution within the neighborhood. By alternating between the destroy and repair operators, LNS can explore large parts of the solution space. Although LNS works with only one destroy and repair operator, ALNS provides an adaptive layer on top of LNS, where multiple destroy and repair operators can be selected based on the efficiency of each operator. An adaptive weight is made for each operator that influences the chance of being selected for the current iteration. ALNS can produce solutions more diversified than LNS because different parts of the solution can be destroyed.

5-3-2 Initial solution generation

The initial solution of ALNS should be properly designed, as it has a significant impact on the performance of the algorithm. For the railway disruption rescheduling problem, the initial order of the original timetable is a solid starting point. Typically, trains should operate according to the original timetable before the disruption occurs. Using the original ordering gives a solution that is most relevant to the original situation, while the destroy and repair operators can gradually develop a new solution with diverging event orders.

5-3-3 Destroy operators

To destroy a part of the solution found in each iteration, three destroy operators are used, i.e., destroy event orders in a random station, destroy event orders in a random train, and destroy event orders first cancelled train line.

Destroy event orders at a random station

The destroy station event order operator randomly selects a station and partially destroys the order of events that occur within a time window. The destroy operator can be formulated as

$$\{a_{e,e'} | x_e - x_{e'} \leq \Delta, s_e = s_{e'}, s_e = s, \} \tag{5-26}$$

where the constant Δ is the maximum allowed difference between events that are part of the destroy operator and s is the selected station of the destroy operator. The constant Δ must be chosen carefully, as a too large value can result in a large destroyed solution, resulting in long repair solution times, especially at stations with many departure events. On the other hand, a too small Δ results in too few events being able to be destroyed, and the approach might have a hard time finding the global optimum. The operator is particularly efficient at stations with several departure events and can ensure that the order of events for lines with transfers can be rearranged.

In Fig. 5-2a, a visual example of the destroy operator is given. The black dots are events. The gray ovals denote the destroyed events. Only the orders of events at one station that occur within Δ from each other are destroyed.

Destroy event orders in a random train line

At each station the train line passes, the destroy line order operator destroys the previous and next events of the train line. This destroy operator is particularly useful for trains that can have a short-turn in the rescheduled timetable, as a short-turn typically causes a delay for a large part of the train service. To ensure that short-turning is possible, each previous and next event is destroyed, allowing the departure times of the line to be delayed past the other events. The operator can be described as

$$\{a_{e,e'} \cap a_{e'',e} | \tau_e = \tau_r, s_e = s_{e'}, s_e = s_{e''}\} \tag{5-27}$$

where τ_r is the train selected randomly.

In Fig. 5-2b, a visual example of the destroy operator is given. The black dots are events. The gray ovals denote the destroyed events. The order of events around the train line is destroyed.

Destroy event orders first cancelled train line

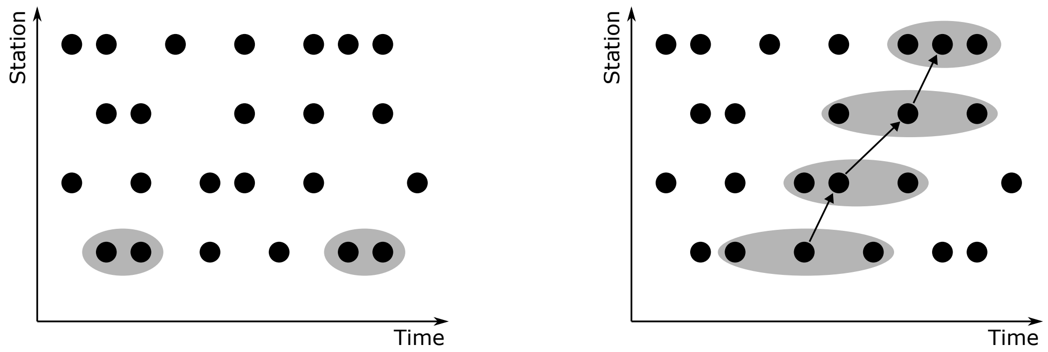
Many train conflicts occur at the beginning of the disruption, which may result in the ALNS algorithm remaining in the transition stage (more information about the transition stage can be found in 5-3-4). Trains at the start may be canceled because there are no other rescheduling measures available due to the fixed order. This operator will provide more solution space by destroying the event order around the selected train. In this operator, the train that corresponds to the first occurring cancelled event is selected. The event destroying component is similar to the destroy random train line operator:

$$\{a_{e,e'} \cap a_{e'',e} | \tau_e = \tau_f, s_e = s_{e'}, s_e = s_{e''}\} \tag{5-28}$$

where τ_f is the train corresponding to the first occurring event that is cancelled.

5-3-4 Repair operator

The repair operator is used to solve the destroyed solution directly as an MILP formulation. Within the solution space of the destroyed solution, the repair operator will find the optimal



(a) Destroy event order at a station. **(b)** Destroy event order around a train line.

Figure 5-2: Destroy operators for ALNS.

solution. The primary reason for solving the destroyed solution directly is that the event ordering is only a component of the total problem. It is challenging to build a heuristic repair operator because the event ordering is heavily dependent on the rescheduling measures, i.e., the rescheduling measures influence the sequence of events and vice versa.

At the beginning of a disruption, already departed trains cannot be cancelled, as the trains already left their station of origin. Compared to trains that still need to depart, the options to avoid conflicts are reduced to short-turning, relocating to a shunting yard at an intermediate station, or delaying them until the disruption is resolved. For the MILP approach, no problems arise as the rescheduling measures can be applied without limitations; however, for the ALNS approach, the rescheduling measures are restricted to the fixed station event order, which may result in infeasibility of the model. To overcome the infeasibility of the model, the phase constraint in (3-13) can be relaxed. In general, cancellation of past departures should be avoided; hence, the associated variables are moved to the objective function with a large penalty, called the request bank. The new optimization problem allows undesirable solutions to be generated to avoid infeasibility during the start of the ALNS algorithm. This stage is also known as the transition stage (Ropke and Pisinger (2006)).

The relaxed objective function can then be written as:

$$\min \sum_{e \in E_{\text{arr}}} \Psi_{\text{delay}}(x_e - o_e) + \sum_{e \in E_{\text{arr}}} \Psi_{\text{cancel}} c_e + \sum_{\forall s \in S} \Psi_{\text{sc}} \xi_s \sum_{\forall e \in E_{\text{arr}} \cup E_{\text{dep}} : o_e \leq t_{\text{start}}} c_e \gamma, \tag{5-29}$$

$$\text{s.t. constraints (3-5) - (3-12) and (3-14) - (3-37) and (4-3) - (4-17)} \tag{5-30}$$

where γ is the weight of cancellations from past events. The weight must be chosen to be $\gamma \gg \Psi_{\text{cancel}}, \Psi_{\text{delay}}, \Psi_{\text{delay}}$ to ensure that past events are cancelled when no feasible options are possible.

Note: The past cancellation can also be seen as a practical implementable solution, where a past cancellation means that a train must go to the nearest shunting yard.

5-3-5 Adaptive searching strategy

The main part where ALNS extends from LNS is the adaptive layer that is used to select operators. Based on the weights, the algorithm chooses an operator where the weights can dynamically change during the search process based on the past success. Using the roulette wheel principle, the methods can be selected. The roulette wheel can be formulated as (Ropke and Pisinger (2006)):

$$\frac{w_j}{\sum_{i=1}^{3} w_i}, \tag{5-31}$$

w_j is the probability of selecting heuristic j. Since there are three destroy operators, the probability can be determined by dividing by the sum of all the weights.

The power of ALNS is the adjustment of weights that are adjusted based on the current performance of a destroy operator. To determine the performance of the operator, the parameter σ is introduced, which resembles the reward that is given when an operator is successful, i.e., the heuristic finds a new best solution. As the operators themselves are random and their performance can be highly dependent on the randomness of the operator, the weights of the operators should not be updated at each iteration, as this can lead to fluctuating results. Therefore, segments are introduced over which the average is taken to better reflect the performance of the operator. A segment is a predefined number of iterations of ALNS over which the weights are updated. For example, a segment could be defined as 15 iterations, such that each weight can be updated with the performance of an average of 5 iterations, if we have 3 destroy operators. To calculate the new weights at the end of each segment, the following formula is introduced

$$w_{i,j+1} = w_{ij}(1-r) + r\frac{\pi_i}{\sigma_i}, \tag{5-32}$$

where w_{ij}, $w_{i,j+1}$ are the weight of heurstic i, for segment j and $j+1$. Variable π_i is the performance of the heurstic i during segment j, which depends on the parameter σ. Parameter θ_i, resembles the number of selection of heuristic during i.

Algorithm 1 shows an overview of the ALNS algorithm. In line 1 the inputs are given. In line 2 the best order a^b is set as the initial order. In line 3 an initial solution x is made using the MILP formulation and the initial ordering. In line 5 the iterations are started. In line 6 one of the destroy operators is selected based on the weights. In line 8 the best order a^b is destroyed with destroy operator d. In line 9 the solution time is set to zero. In lines 10-12 the destroyed order a^d is used as input to solve the MILP problem until a solution is found or the maximum iteration time $t_{\max}$ is met. In lines 13-16 the variables are replaced when the new solution of the current iteration is better than the best solution. In line 18 the algorithm is stopped when a number of iterations is reached or the maximum algorithm time is exceeded.

Algorithm 1 ALNS Algorithm

Input: the initial ordering a, MILP model m
$a^b \leftarrow a$;
$x^b \leftarrow r(m(a^b))$;
$w \leftarrow (1, ..., 1)$;
repeat
  select destroy operator $d \in \Omega$ using w;
  select repair operator r;
  $a^d \leftarrow d(a^b)$;
  $t \leftarrow 0$;
  while $t < t_{\max}$ **do**
    $x^t \leftarrow r(m(a^d)$;
  end while
  if $c(x^t) < c(x^b)$ **then**
    $x^b \leftarrow x^t$;
    $a^b \leftarrow a \in x^b$;
  end if
  update w;
until Stop criterium is met
return x^b, a^b

5-4 Conclusions

In this chapter, two objective functions are proposed, i.e., for the operator-centric model and the passenger-centric model. To directly solve the passenger-centric problem, two transformations are introduced, such that the passenger-centric model can be directly solved as an MILP problem. The transformation for handling train orders at the stations introduces a significant number of binary variables. An adaptive large neighborhood search is developed to improve the solution speed of the original MILP problem. The ALNS approach is modified to fit the train ordering formulation of the disruption management model. Three heuristic destroy operators are introduced to explore the solution space. The ALNS offers a way to make a trade-off between solution performance and quality.

Chapter 6

Case study

In this chapter, three case studies are considered, i.e., to illustrate the effectiveness of rolling stock circulation, the passenger-centric model, and the ALNS algorithm, respectively. The remainder of this chapter is organized as follows. In Section 6-1 the general setup of the case studies is shown. In Section 6-2 the performance of the rolling stock circulation is tested using the operator-centric model. In Section 6-3 the novel passenger-centric formulation is evaluated. In Section 6-4 a comparison is made between the MILP approach and the ALNS algorithm. In Section 6-5 the conclusions of the case studies are drawn.

6-1 Set-up

For the case studies, we use a part of the Dutch railway network between the cities Utrecht (Ut) and 's-Hertogenbosch (Ht) are considered. In practice, five train lines operate every 30 minutes in both directions on the considered railway network. The train lines have different starting and terminal stations, as shown in Table 6-1. Each starting and terminal station is equipped with a shunting yard and sufficient trains to ensure normal operation. To evaluate the impact of disruptions and rescheduling measures on passengers and rolling stock, several significant stations outside the Ut-Ht segment are also considered. In Fig. 6-1 the railway network is shown.

For each case study, we will use two scenarios, i.e., the large-scale case study with all 5 lines operating of Table 6-1, and the small-scale case study with 3 lines operating, consisting of trains SP6000, SP6900 and IC3500.

A more detailed layout of the tracks between Ut and Ht is given in Fig. 6-2. The blue tracks are designated for regional trains (Sprinters), which stop at each intermediate station between Ut and Ht. The red tracks are designated for intercity trains that only make stops at Ut and Ht. A large part of the segment consists of 2 tracks in each direction, allowing the sprinters and intercities to run separately. The tracks around Culemborg (Cl) and Zaltbommel (Zbm) stations are single track in each direction, meaning that sprinters and intercities must use the same track. More detailed information on each train line is given in Table 6-1.

For the optimization the time window between 8:00 and 11:00 is considered. A disruption is set to occur between 8:52 and 10:00 between the Culemborg (Cl) and Geldermalsen (Gdm) stations. The disruption is a full blockage, i.e., both tracks are not available, and no railway traffic is possible between Cl and Gdm until the disruption is over. The main parameters for this optimization are listed in Table 6-2.

All case studies are solved in Python using the optimization software GUROBI release 9.5.0. The experiments are performed on a computer with an Intel Core i7-9750H CPU and 16GB RAM.

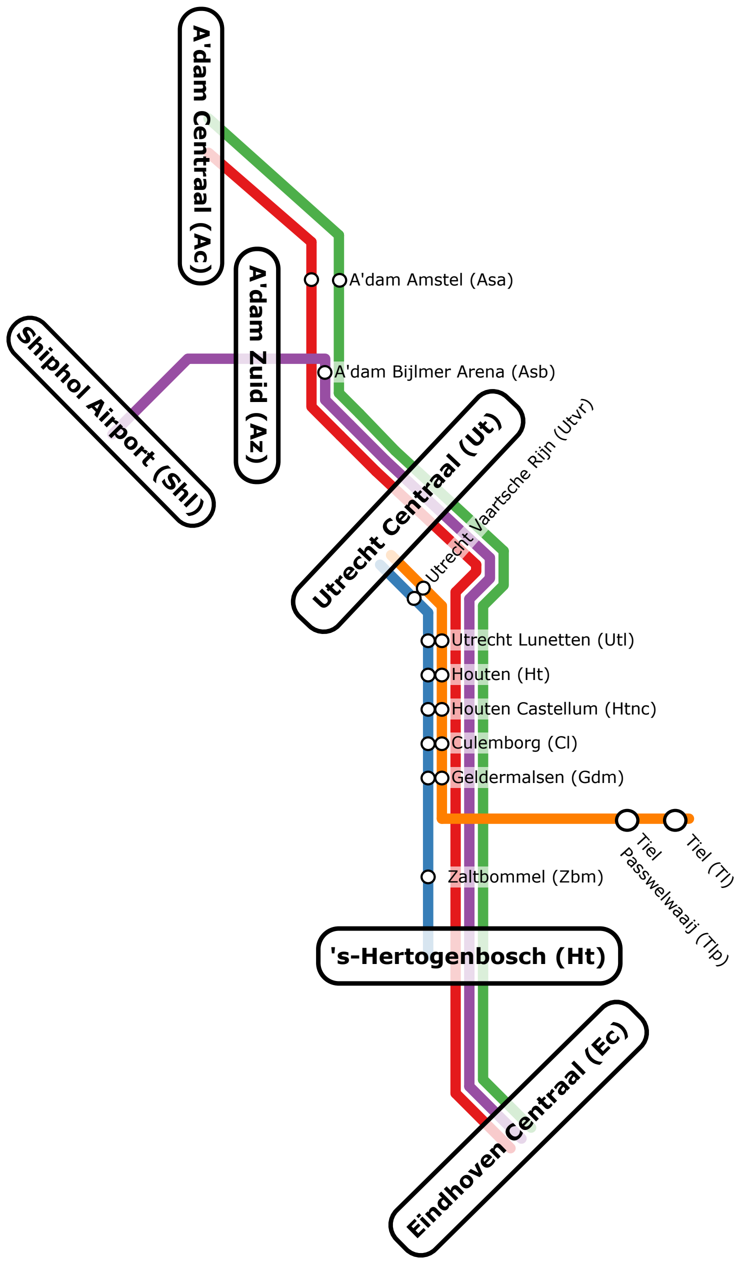


Figure 6-1: The considered railway network.

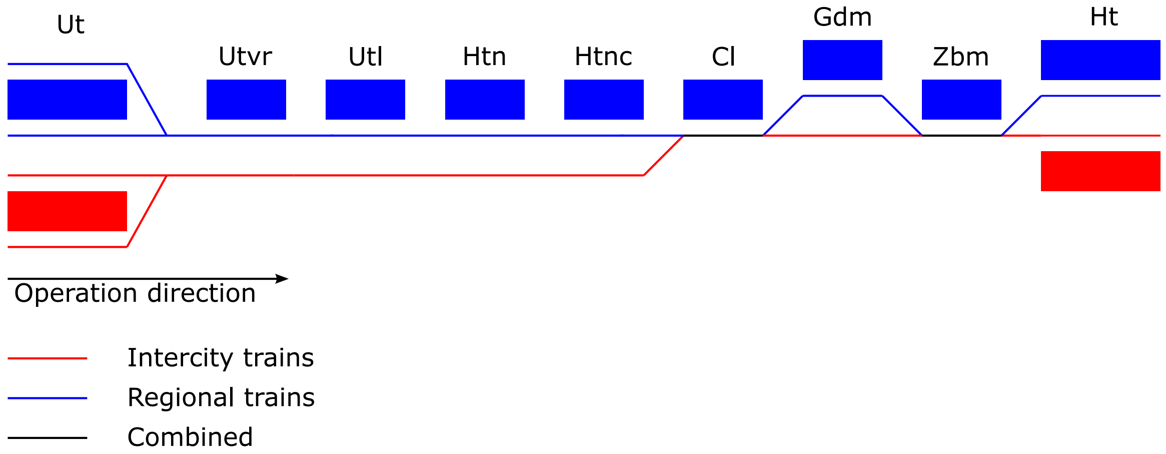


Figure 6-2: Schematic layout of a railway line in one operation direction.

Table 6-1: Information and parameters of the train lines.

Train line	Train type	Terminal station 1	Rolling stock 1	Terminal station 2	Rolling Stock 2
IC800	Intercity	Asa	4	Ec	2
IC3500	Intercity	Asa	4	Ec	4
IC3900	Intercity	Shl	4	Ec	3
SP6000	Sprinter	Ut	2	Tl	3
SP6900	Sprinter	Ut	1	Ht	3

Table 6-2: Parameters for the optimization.

Parameters	Value
Minimum headway	120 s
Short-turn	300 s
Minimum dwell time	30 s
Recovery time	3600 s
Maximum delay	1800 s

6-2 Case study A - Performance of rolling stock circulation

To compare the effectiveness of including rolling stock circulation, a model is used that assumes infinite reserve rolling stock. In theory, these models give good results; however, when applied in practice, trains may be cancelled due to a lack of reserve rolling stock. To evaluate the practicability of the infinite rolling stock solution, a second optimization cancels trains at their origin station when there would be insufficient rolling stock in real-life. Furthermore, two optimizations are conducted to evaluate the effectiveness of rolling stock circulation, i.e., an optimization with rolling stock circulation without shunting at intermediate stations and an optimization with rolling stock circulation with shunting at intermediate stations. In Table 6-5 the stations with shunting-yards are shown.

The four optimizations are conducted on the large-scale case study using the MILP formulation of (5-1). The weights for the objective function are set to $\Psi_{\text{cancel}} = 100$ and $\Psi_{\text{delay}} = 0.1$. The generated disruption timetables are shown in Fig. 6-3 and Fig. 6-4, and the found objectives are shown in Table 6-3. To provide a clear overview, the stations between Ut-Ht are shown in the disruption timetables. The complete disruption timetables with all the stations can be found in Appendix A-1. In the timetables, the red rectangle represents the disruption between stations Cl and Gdm. The solid lines represent the new paths of the trains. The dashed lines show the original path of the trains that are cancelled. The solid lines show the original path of trains that are delayed.

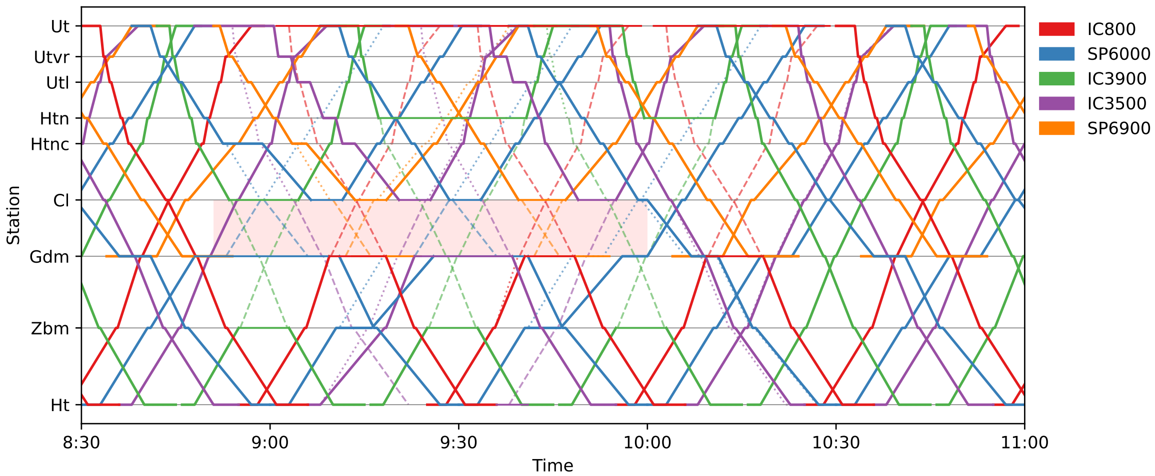
Table 6-3: Objective functions

	Best Objective	Cancellations	Delay
Infinite rolling stock	3450	12	22498
Simulated with infinite rolling stock	6018	36	24183
Rolling stock circulation	5245	32	21453
Rolling stock circulation with intermediate shunting	4481	31	13812

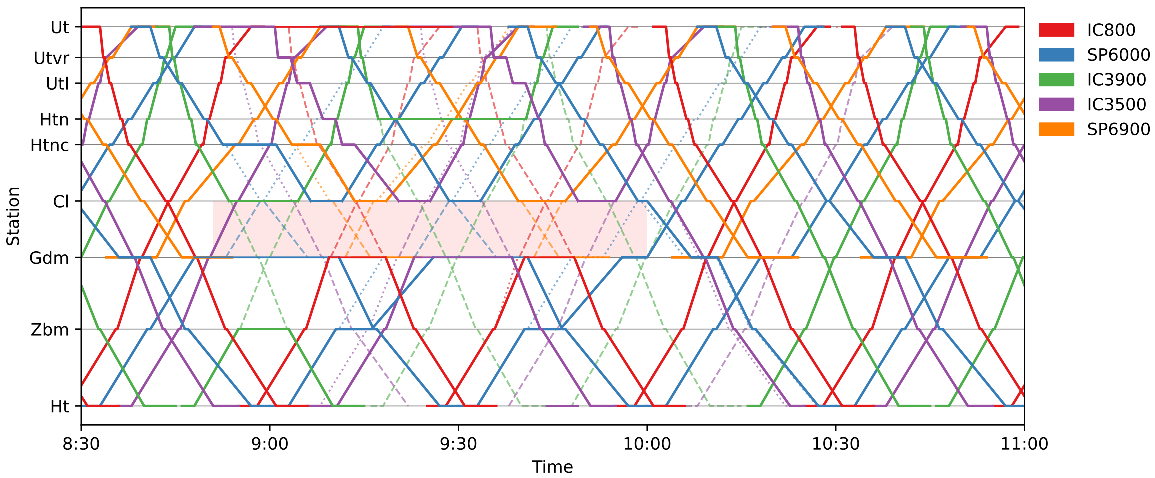
When comparing the model with infinite rolling stock and the practical solution, a significant increase in cancellations can be seen. In the down direction, the third IC3900 (green) train that short-turns at Htn is cancelled. Furthermore, a difference in operation can be seen for the IC800 (red) after the disruption has ended. Additionally, in the up direction, the third IC3500 (purple) train, and the second and third IC3900 trains that short-turns at station Zbm are cancelled. Looking at Fig. 6-4 most of the train lines appear to have sufficient rolling stock; however, the intercities have additional stops after stations Ut and Ht, e.g., Ec and Asa, which require additional running time before the trains can be served as rolling stock in the opposite direction. In Appendix A-1 the full disruption timetable is shown. Table 6-3 shows that the practical solution has an increase of 74% in best objective when compared to the infinite rolling stock optimization. The increase in the best objective is mainly due to an increase in cancellations from 12 to 36. The simulated model attempts to apply the same disruption timetable as the model with infinite rolling stock; however, many trains are cancelled due to a lack of reserve rolling stock at the terminal stations.

The rolling stock circulation without intermediate shunting model can lower the best objective value by considering the reserve rolling stock in the optimization. When comparing the disruption timetables of Fig. 6-3b and Fig. 6-4a, it can be observed that the IC3500 (purple) running from Ht to Gdm between 9:00 and 9:30, which short-turns at Gdm, is cancelled in the model that considers rolling stock circulation. By considering rolling stock, the model is able to strategically cancel the train services between 9:00 and 9:30, in order to keep reserve rolling stock for the run at 10:00 from Ht to Ut.

The model with rolling stock circulation and intermediate shunting shows great improvement compared to the model with rolling stock circulation without intermediate circulation. The best objective is reduced from 5245 to 4481 by reducing one cancellation and reducing the total arrival delay from 21453 to 13812 seconds. Especially, the reduced total arrival delay is

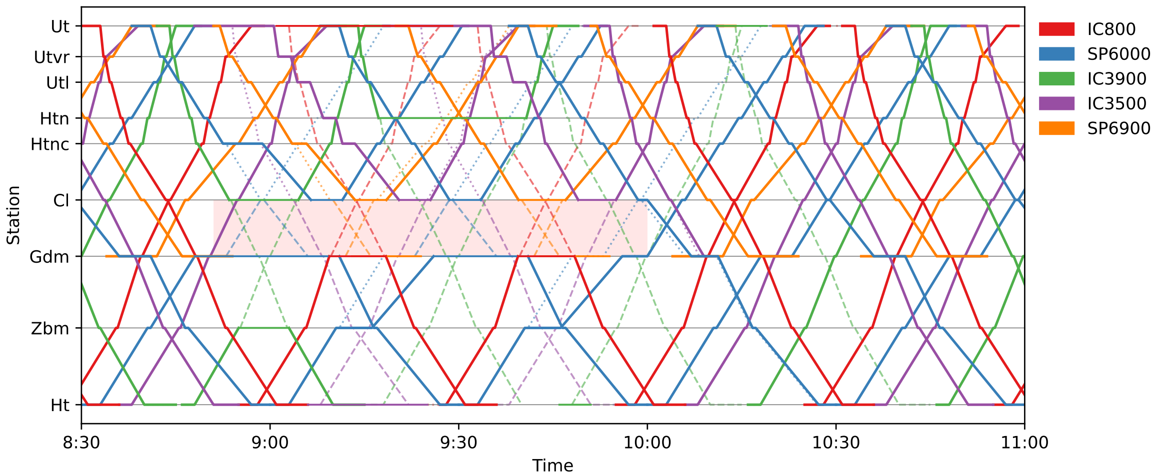


(a) Generated timetable with infinite rolling stock.

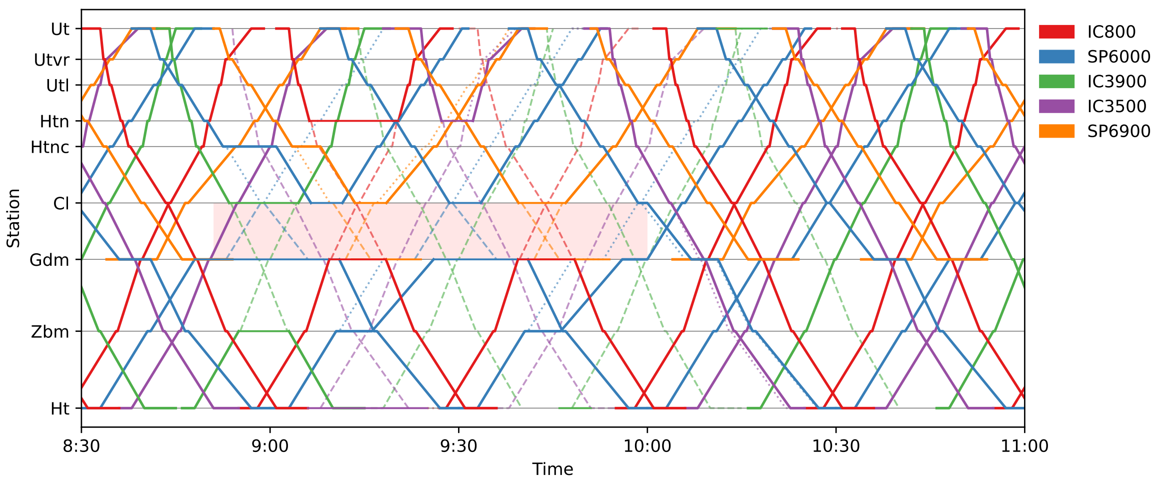


(b) Practical implementation of timetable with infinite rolling stock.

Figure 6-3: Generated timetables with infinite rolling stock.



(a) Generated timetable with rolling stock circulation.



(b) Generated timetable with rolling stock circulation and intermediate shunting actions.

Figure 6-4: Generated timetables with rolling stock circulation.

remarkable and is mainly due to the ability to store rolling stock at the Ut at the begin of the disruption. As seen in Fig. 6-4a, the IC3500 running in the down direction between 9:00 and 9:30 is run with a significant delay. The model without intermediate shunting can only short-turn or delay the train until after the disruption, as the train has already departed from its origin station and, therefore, cannot be cancelled. Instead, the model with intermediate shunting is able to store this rolling stock at the station Ut for later usage and avoids running the train with significant delays.

6-3 Case study B - Performance of passenger-centric model

To solve the MILP problems of (5-3) and (5-4), the passenger OD-demands are required. The passenger-demands in this case study are based on the OD-demands as shown in Table 6-4. The passenger OD data are generated based on publicly available data on the number of boarding passengers per station in 2019[1]. The morning travel peak is estimated at 20% of the total number of passengers per day. Between several stations, the OD-demand is set to zero because a faster route exits outside the scope of this network; e.g., between station Ac and Shl, there is a direct line that provides a faster route that is not part of this network. To calculate the congestion level, the passenger capacity at each station should be known. The station information is shown in Table 6-5, and the station capacity is generated based on the size of the station.

Table 6-4: Origin-destination demands per day.

Stations	Ac	Asa	Shl	Az	Asb	Ut	Utvr	Utl	Htn	Htnc	Cl	Gdm	Zbm	Ht	Tp	Tl	Ec
Ac		589	0	0	0	3394	677	326	687	435	780	454	337	2480	122	144	2287
Asa	589		0	0	0	596	119	57	120	76	137	80	59	435	21	25	401
Shl	0	0		503	236	1629	325	157	329	209	374	218	162	1190	59	69	1097
Az	0	0	503		153	1060	211	102	214	136	244	142	105	774	38	45	714
Asb	0	0	236	153		497	99	48	100	64	114	66	49	363	18	21	334
Ut	3394	596	1629	1060	497		685	330	695	440	789	459	341	2509	124	146	2313
Utvr	677	119	325	211	99	685		66	138	88	157	92	68	500	25	29	461
Utl	326	57	157	102	48	330	66		67	42	76	44	33	241	12	14	222
Htn	687	120	329	214	100	695	138	67		89	160	93	69	508	25	29	468
Htnc	435	76	209	136	64	440	88	42	89		101	59	44	322	16	19	297
Cl	780	137	374	244	114	789	157	76	160	101		105	78	577	28	33	532
Gdm	454	80	218	142	66	459	92	44	93	59	105		46	335	17	19	309
Zbm	337	59	162	105	49	341	68	33	69	44	78	46		249	12	14	230
Ht	2480	435	1190	774	363	2509	500	241	508	322	577	335	249		90	106	1690
Tp	122	21	59	38	18	124	25	12	25	16	28	17	12	90		5	83
Tl	144	25	69	45	21	146	29	14	29	19	33	19	14	106	5		98
Ec	2287	401	1097	714	334	2313	461	222	468	297	532	309	230	1690	83	98	

The passenger optimization is conducted on the small-scale case study as the large-scale case study was unable to solve within an acceptable time. By reducing the number of trains, the complexity is significantly reduced and will still give a good evaluation of the effectiveness of the overcrowdedness.

To compare the effectiveness of the passenger-centric model, two optimizations are carried out with different weighted objective functions (5-3). As a benchmark, the weights are set to

[1] https://dashboards.nsjaarverslag.nl/reizigersgedrag/

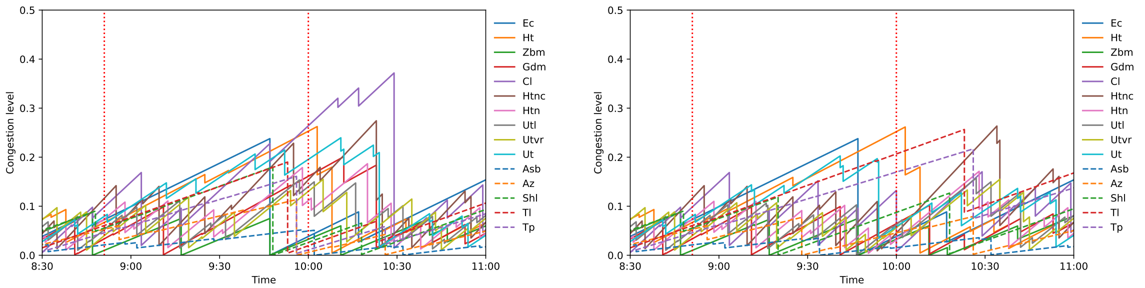
Table 6-5: Station information.

Stations	Passenger capacity	Short-turning	Shuting-yard
Ac	5000	Yes	Yes
Asa	5000	Yes	No
Shl	5000	Yes	Yes
Az	5000	Yes	No
Asb	5000	Yes	No
Ut	5000	Yes	Yes
Utvr	1000	No	No
Utl	500	No	No
Htn	1000	Yes	No
Htnc	500	No	No
Cl	1000	Yes	No
Gdm	1000	Yes	No
Zbm	1000	Yes	No
Ht	5000	Yes	Yes
Tp	500	Yes	No
Tl	500	Yes	No
Ec	5000	Yes	Yes

$\Psi_{\text{cancel}} = 100$, $\Psi_{\text{delay}} = 1$ and $\Psi_{\text{sc}} = 0$. This objective function shows the congestion levels at stations when only train delays and cancellations are included in the objective function. The weights of the second objective function are set to $\Psi_{\text{cancel}} = 100$, $\Psi_{\text{delay}} = 1$ and $\Psi_{\text{sc}} = 50000$, ensuring that the congestion levels at stations are considered.

In Fig. 6-5 the congestion levels at each station in both cases are shown. Each line represents the congestion level at a station over time. The vertical dotted lines denote the start and end of the disruption. The figures clearly show an increase in the congestion level during the disruption, as many passengers cannot reach their destination because their destination is on the other side of the disruption, e.g., passengers waiting at Ec with a destination at Ut are not able to travel until the disruption is over. Table 6-7 shows the maximum congestion level at each station. Here we can see that most of the stations have a reduced maximum congestion level. Especially, at the stations Gdm and Cl a significant reduction in maximum congestion level can be seen, while at Tp and Tl a significant increase can be seen.

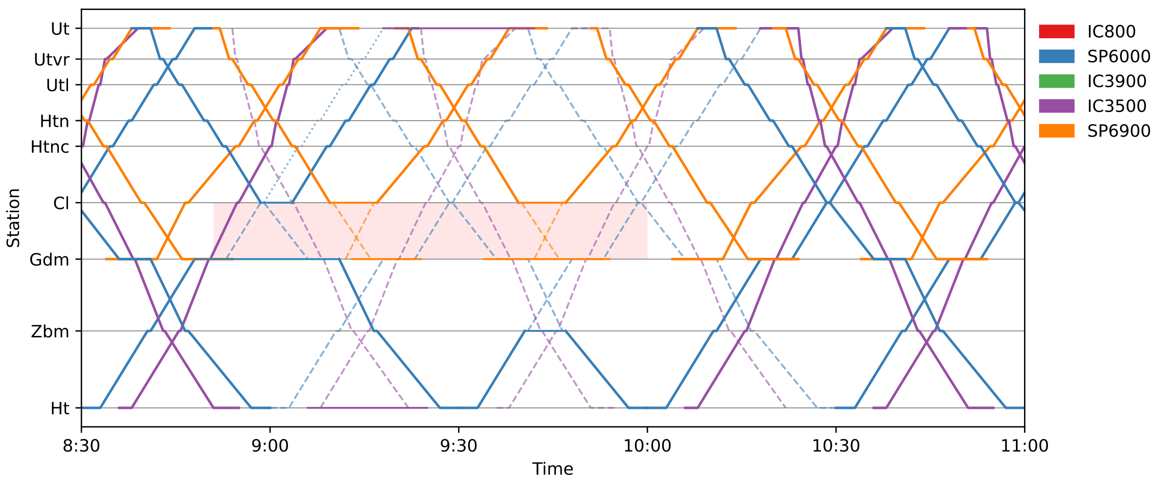
In Fig. 6-6 the disruption timetables for both optimizations are shown. The reduced congestion level of $\Psi_{\text{sc}} = 50000$ can be explained by the extra SP6000 (blue) sprinters running between stations Ut-Cl and Gdm-Ht. One extra sprinter is operated between station Ut-Cl and Gdm-Ht is extended to short-turn at a later station. After the disruption one IC3500 and one SP6000 are moved to operate directly after the disruption, with a slight delay. This ensures that the spike of passengers at the end is reduced as soon as possible. As shown in the Table 6-6, the arrival delay has increased by operating these extra sprinters, however, the reduction in congestion level at the stations is more valued. By considering the congestion level while optimizing, a balance between deviating from the timetable and the congestion level can be made. The passenger-centric model ($\Psi_{\text{sc}} = 50000$) is more practical than the operator-centric model ($\Psi_{\text{sc}} = 0$), as it can value stations with many waiting passengers over deviations from the original timetable, while the operator-centric model will only consider minimizing deviations from the original timetable.



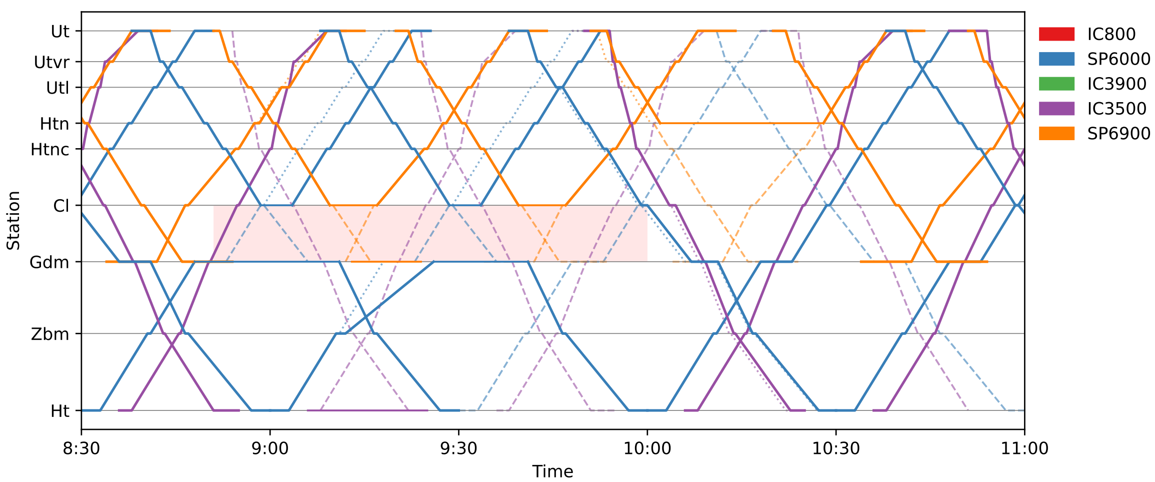
(a) $\Psi_{\text{sc}} = 0$.

(b) $\Psi_{\text{sc}} = 50000$

Figure 6-5: Congestion level of the stations over time.



(a) $\Psi_{\text{sc}} = 0$



(b) $\Psi_{\text{sc}} = 50000$

Figure 6-6: Disruption timetables of case study B.

Table 6-6: Train cancellations and delays.

	Cancellations	Delays
$\Psi_{\text{sc}} = 0$	51	1895
$\Psi_{\text{sc}} = 50000$	51	5004

Table 6-7: Maximum congestion level at each station

Weights	Az	Asb	Ut	Utvr	Utl	Htn	Htnc	Cl	Gdm	Zbm	Ht	Tp	Tl	Ec
$\Psi_{\text{sc}} = 0$	0.114	0.052	0.240	0.155	0.154	0.187	0.274	0.372	0.201	0.102	0.263	0.161	0.193	0.298
$\Psi_{\text{sc}} = 50000$	0.102 $\downarrow$	0.038 $\downarrow$	0.203 $\downarrow$	0.155 $\approx$	0.157 $\uparrow$	0.171$\downarrow$	0.264$\downarrow$	0.115$\downarrow$	0.115$\downarrow$	0.123$\uparrow$	0.262$\downarrow$	0.212$\uparrow$	0.257$\uparrow$	0.298 $\approx$

6-4 Case study C - Comparison of MILP and ALNS

To evaluate the effectiveness of the ALNS algorithm, the MILP approach, the LNS algorithm, and the ALNS algorithm are applied to a small-scale case study and a large-scale study. The weights of the objective function are set to $\Psi_{\text{cancel}} = 100$, $\Psi_{\text{delay}} = 1$ and $\Psi_{\text{sc}} = 1$. Due to the reduced complexity of the small-scale study, we are able to calculate the optimal solution with the MILP approach in an acceptable amount of time. The relaxed objective function of (5-29) is used such that ALNS can avoid infeasibility and use the transition stage. The maximum iteration time of LNS and ALNS is set to 400 seconds.

In Table 6-8 the results are shown. The MILP approach is able to find the optimal solution of 6698 in 426 seconds. ALNS is able to find the optimal solution in 8 seconds and 1 iteration after the initial solution. It shows that ALNS is significantly more efficient in the small-scale case study while maintaining the same performance as the MILP approach. LNS has a comparable solution time with ALNS, however the optimum was not found. The main advantage of ALNS is that the train order of the optimal solution does not differ much from the initial order; therefore, not many iterations are required to reach the train order of the optimal solution.

In Table 6-9 the results of the large-scale case study are shown. The MILP approach was unable to find a feasible solution within the 10 hour (36000s) time limit, only a best bound of 12521 was found. The ALNS algorithm was set to a limit of 100 iterations and was able to find a solution of 13387. LNS has a comparable solution time with ALNS, however the solution quality of LNS is worse. In Fig. 6-7a the best objective value for each iteration. In the first 7 iterations, the ALNS algorithm remains in the transition stage, i.e., the repair operator is not able to find a feasible solution without using the request bank. Several iterations are required before a new event order can be found without cancelling past events. The 7 iterations take 453 seconds.

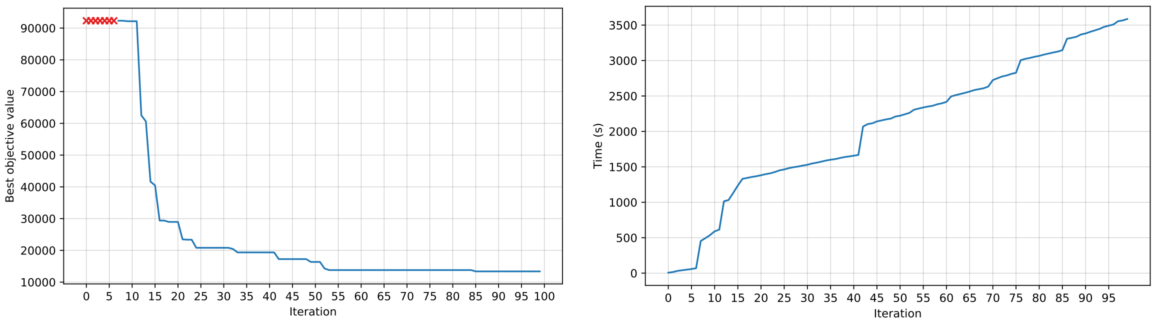
Table 6-8: Small-scale case study results.

	Performance	CPU time (s)
MILP	6998	426
LNS	7167	16
ALNS	6998	8

Table 6-9: Large-scale case study results.

	Performance	CPU time (s)
MILP	-	36000
LNS	14052	3152
ALNS	13387	3586

In Fig. 6-7a the cumulative computational time is shown during the iterations. Iterations 12 and 42 have computational times that reach the time limit of 400 seconds. Both iterations correspond to the destroy event order in a random station operator, the random station being Ut. Iteration 7 has a computational time of 384 seconds and is the destroy event order in a random station operator, with random station being Cl. The other iterations have computational times that are below 200 s.



(a) Objective function. **(b)** Computational time.

Figure 6-7: ALNS best objective and computational time during iterations.

The poor performance of the MILP approach for the large-scale case study compared to the MILP approach for the small-scale case study is that the large-scale case study has a significant increase in variables. In the small-scale case study, the number of binary variables is 9177, while the large-scale has 16963. In Fig. 6-8 and Fig. 6-9 the disruption timetable of both case studies are shown. In Appendix A-2 the whole disruption timetable is shown.

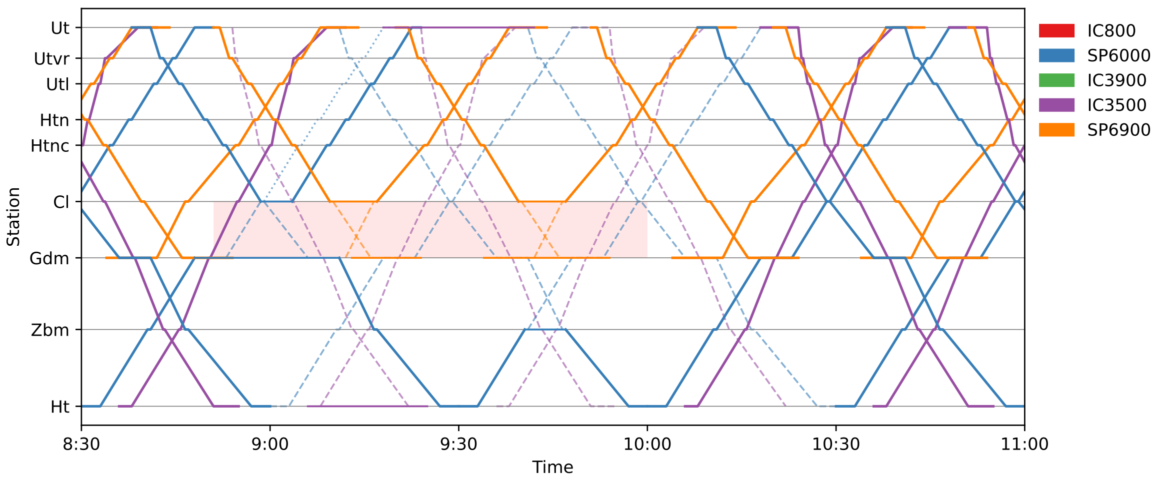


Figure 6-8: Best found disruption timetable of ALNS for the small-scale case study.

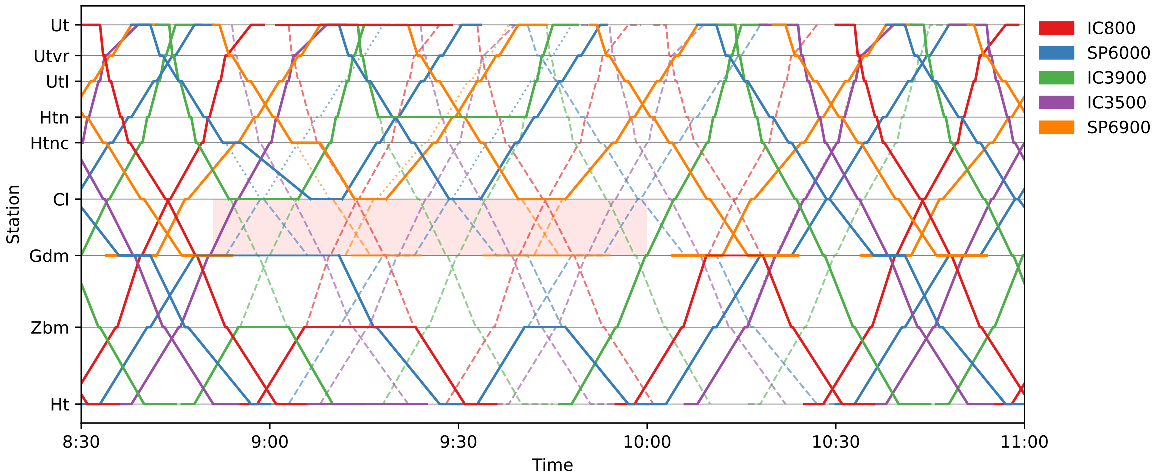


Figure 6-9: Best found disruption timetable of ALNS for the large-scale case study.

6-5 Conclusions

Case study A evaluates the effectiveness of rolling stock circulation. A comparison is made between a model with infinite rolling stock and a practical model. The results show that the infinite rolling stock model has good solutions in theory; however, the practical model shows that the disruption timetable found by the model with infinite rolling stock requires significant additional cancellations due to insufficient rolling stock. A model is run with rolling stock circulation, outperforming the practical solution as there is a better strategy for which trains to cancel. Furthermore, a model with rolling stock circulation and intermediate shunting. The intermediate shunting further improves the performance, as trains can be stored for later use at intermediate stations instead of being run with large delays. The model is especially efficient in minimizing the total delay.

In case study B the effectiveness of the passenger-centric model is evaluated. Due to the significant increase in complexity, the optimization is conducted on the smaller-scale case study. A comparison is made between a case where the congestion level is include in the objective function and a case without including the congestion level. The results show that the model is able to lower the maximum congestion levels by slightly changing the timetable. A balanced trade-off can be made between deviating from the original timetable and the congestion level at stations.

In case study C the effectiveness of ALNS is evaluated. First, a comparison is made between MILP, LNS and ALNS in the small-scale case study. ALNS is able to find the optimal solution significantly faster than MILP, while LNS does not converge to the optimal solution. Second, a comparison is made with the large-scale case study. For the large-scale study, MILP was not able to find a feasible solution within 10 hours. ALNS and LNS were able to find a feasible solution, where ALNS was able to find a lower objective.

Chapter 7

Conclusions and discussion

In this thesis, a novel disruption management approach based on the event-activity model is developed that integrates detailed passenger-related factors into the model. The new model aims to describe the congestion level at stations while considering train capacity. The resulting passenger-centric problem is difficult to solve due to the introduction of binary variables for train orders. An ALNS algorithm is proposed to make a balanced trade-off between solution and quality. The remainder of this chapter is as follows. In Section 7-1, the conclusions are given and the research questions are answered. In Section 7-2 recommendations for future work are given.

7-1 Conclusions

The main research question on passenger-centric railway disruption management of this thesis was:

> Can a detailed number of passengers formulation be integrated in a railway traffic disruption management problem to efficiently jointly optimize train delays, cancellation, and overcrowdedness at stations?

The main research question was split into the following subquestions, that can be answered:

1. *How to build a railway management model to incorporate rolling stock circulation and detailed number of passengers in a railway network?*
   In this thesis, a novel formulation for rolling stock circulation in the event-activity network is introduced. By considering shunting actions at intermediate and terminal stations, the number of reserve rolling stock at each shunting yard is kept track of. Furthermore, a novel formulation for the event-activity network is introduced to calculate the number of waiting passengers. Instead of using a static passenger size, the arrival rate of passengers at each station is used to calculate the number of waiting passengers

at each departure event. By determining the time difference between two consecutive departure events, the number of arriving passengers is calculated; therefore, the influence of train delays on the number of waiting passengers is considered. Furthermore, the train capacity is considered by calculating the number of onboard passengers and denying boarding to passengers when a train exceeds its capacity. A formulation inspired by VRPs is introduced to determine the event orders at stations, as train orders can change due to rescheduling measures. The novel formulation is capable of describing the passenger in detail, making it possible to consider overcrowdedness at stations in detail.

2. *How to design an approach to effectively solve the passenger-centric railway disruption management problem?*
   The case studies show that the MILP approach for the passenger-centric problem results in long computation times for small networks, which implies that MILP may not be suitable for real-time timetable rescheduling. Alternative solution methodologies are required to effectively solve the railway disruption management problem. Adaptive large neighborhood search (ALNS) is a well-researched method in VRP. As the proposed passenger-centric railway disruption management model has certain similarities with VRPs, and ALNS has been proven to be efficient for these types of problems. ALNS has multiple components that should be designed carefully to make the algorithm efficient. First, the algorithm uses destroy operators to destroy part of the solution to explore the solution space. Then, the repair operator solves the destroyed solution, finding the optimal neighborhood solution. Second, ALNS requires an initial solution. For the initial solution, the event ordering of the original timetable is selected, as part of the objective is to minimize the deviation from the original timetable. Third, a relaxation of the MILP formulation is required as the problem may be infeasible due to the fixed event order. Finally, an adaptive layer is introduced that uses the roulette wheel principle to select the destroy operators based on the performance of the destroy operators over a span of multiple iterations. As shown in the case studies, ALNS is able to obtain the optimal solution for the small-scale case study significantly faster than the MILP approach. For the large-scale case study, ALNS was able to find a solution within 453 seconds, while the MILP approach could not find a feasible solution in 10 hours. Compared to LNS, ALNS has better solution quality with a solution time similar to LNS.

To answer the main research question: A detailed number of passengers formulation can be integrated with a railway traffic disruption management model to efficiently reduce overcrowding at stations and trains. The combined results of case studies B and C show that the congestion levels at stations can be reduced, and with ALNS can be solved in an efficient way. The results of case study B show that the maximum congestion levels at stations can be reduced by slightly changing the disruption timetable. Case study C shows that the resulting detailed number of passengers formulation can be efficiently solved with an ALNS while outperforming the MILP approach and the LNS algorithm.

From the case studies, it becomes clear that the passenger-centric MILP problem has become very hard to solve due to the binary variables for the train orders. There are many train lines considered in this study, and especially around larger stations, e.g., Ut, the number of events increases significantly, resulting in many options for the event ordering. The power of the ALNS algorithm lies in that the optimal solution does not necessarily deviate much from the original planned timetable, as deviation from the original timetable is penalized, i.e., an objective function that includes minimization of train delays and cancellations. ALNS uses the original timetable as a starting point and can slowly deviate with the destroy and repair algorithm.

7-2 Future work

In this section, several recommendations are given for future research.

Time-varying passenger arrival rate

The passenger arrival rates are fixed in this thesis. An extension could be made by introducing arrival rate changes based on historical data. Wang et al. (2018) have successfully applied arrival rate changes for an urban railway network. To integrate arrival rate changes, their work can be transformed to fit the event-activity network and extend the passenger-centric model of this thesis.

Flexible coupling of trains

In this thesis, each rolling stock has the same passenger capacity. In practice, many operators are able to change train compositions by coupling and decoupling train, and can select the rolling stock based on the passenger demand. From the results of this thesis, it can be seen that there is a surge of passengers around the end of the disruption, and the selection of train capacity could be further integrated to reduce overcrowdedness during disruption.

Passenger routes

Passenger routes are assumed static in this thesis, i.e., the routes of passengers remain the same despite the rescheduling measures that are applied. In practice, passengers may take different routes based on the implemented disruption timetable. In further work, multiple predefined routes could be integrated into the model so that passengers can select based on preferences and availability. Another option could be to apply passenger reassignment like in the work of Zhu and Goverde (2020a), however, their model already has significant complexity, which could lead to problems with the performance of the model.

Passenger travel time

In this thesis, the focus has been on minimizing crowdedness at stations and trains to ensure safe operations for passengers. Although passenger travel time is implicitly taken into account by minimizing train delays and cancellations, further extensions can be made by taking passenger travel time into account. The passenger travel time can be determined by calculating the difference between the arrival time at their destination and the departure time at the origin; however, problems arise when the passenger travel time is minimized. As the passenger group size is variable in this thesis and the passenger travel time is also variable, the resulting objective function would be non-linear. To avoid non-linearity, an option would be to approximate the passenger travel time with a piecewise affine function.

Close the loop

The optimization in this thesis is classified as open-loop control, i.e., optimization is run once without information updates over time. An interesting research direction would be to implement the model with closed-loop control, e.g., model predictive control (MPC). Improvements should be made to represent current states of the model, e.g., departure times of trains, number of passengers on board of a train, and rolling stock inventory at the shunting yards. A challenge could be to ensure that the problem can be solved within the limited solution time imposed by MPC. An overview of possible strategies can be found in the work of Fang et al. (2015).

Extension of ALNS

In this thesis, ALNS has been tested on a network in The Netherlands; however, it would be interesting to see how the performance of ALNS is on networks with other track configurations and on larger networks. Different destroy operators can be explored that are more suitable for the networks. Furthermore, the options to implement a heuristic repair operator can be explored to shorten the iteration solution time. It will be a challenge to define effective heuristics that can repair event orders and the disruption management problem without using a solver.

Appendix A

Supportive figures

A-1 Case study A

The complete disruption timetables that correspond with case study A.

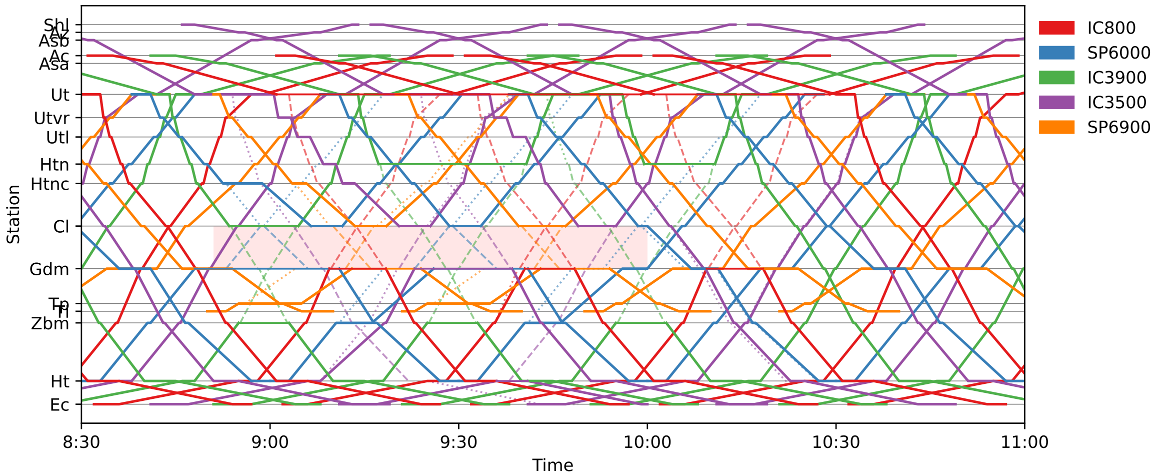


Figure A-1: Generated disruption timetable with infinite rolling stock.

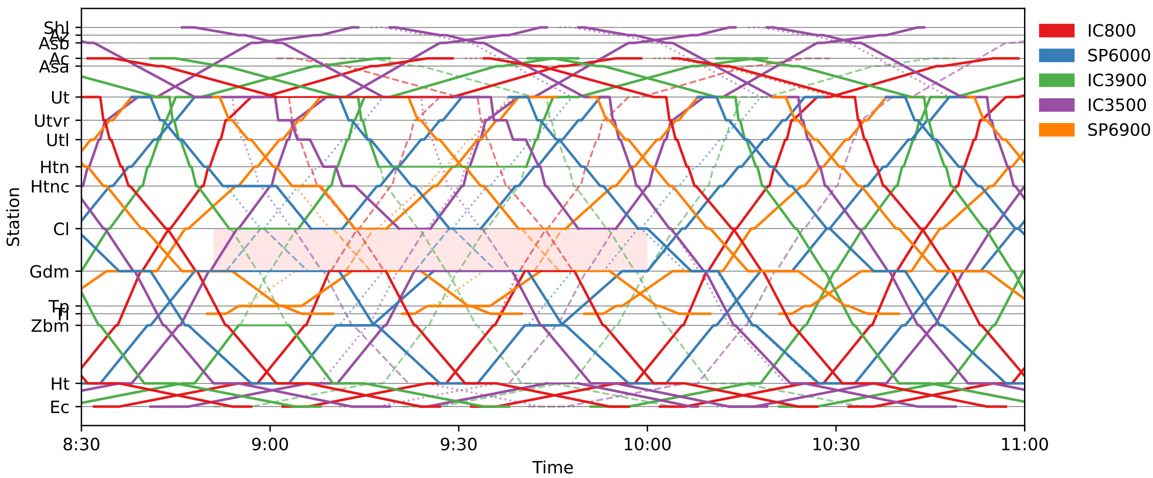


Figure A-2: Practical disruption timetable.

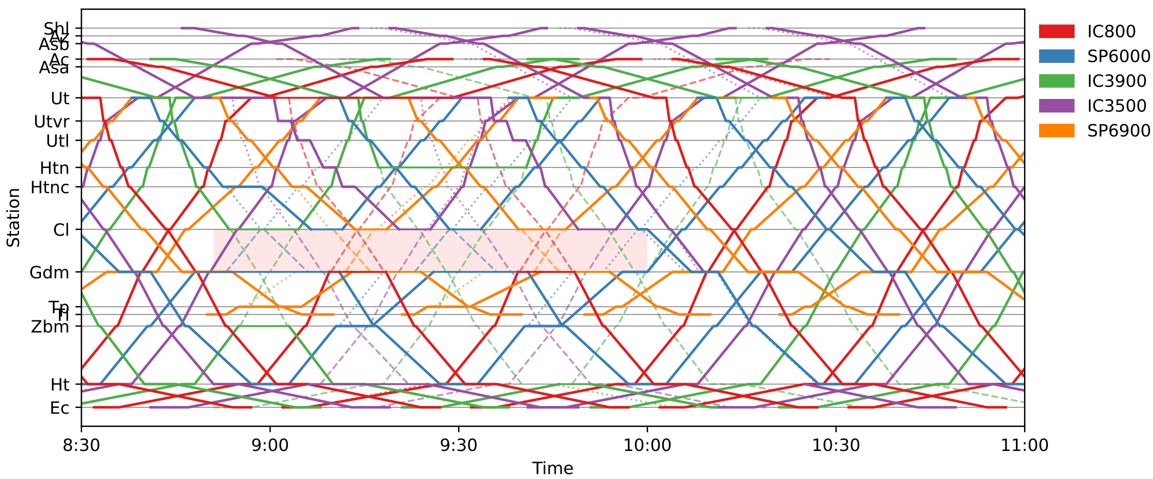


Figure A-3: Generated disruption timetable with rolling stock circulation.

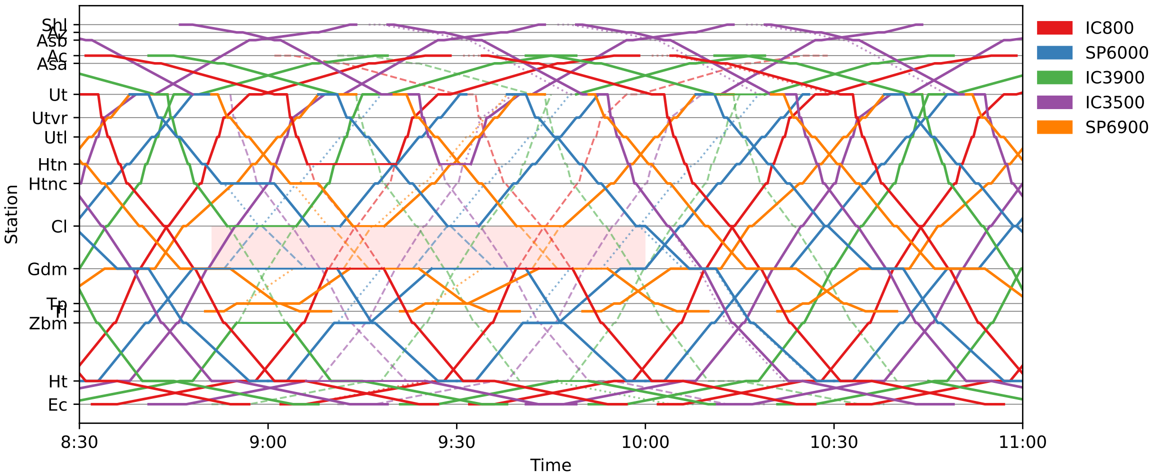


Figure A-4: Generated disruption timetable with rolling stock circulation and intermediate shunting actions.

A-2 Case study C

The complete disruption timetables that correspond with case study C.

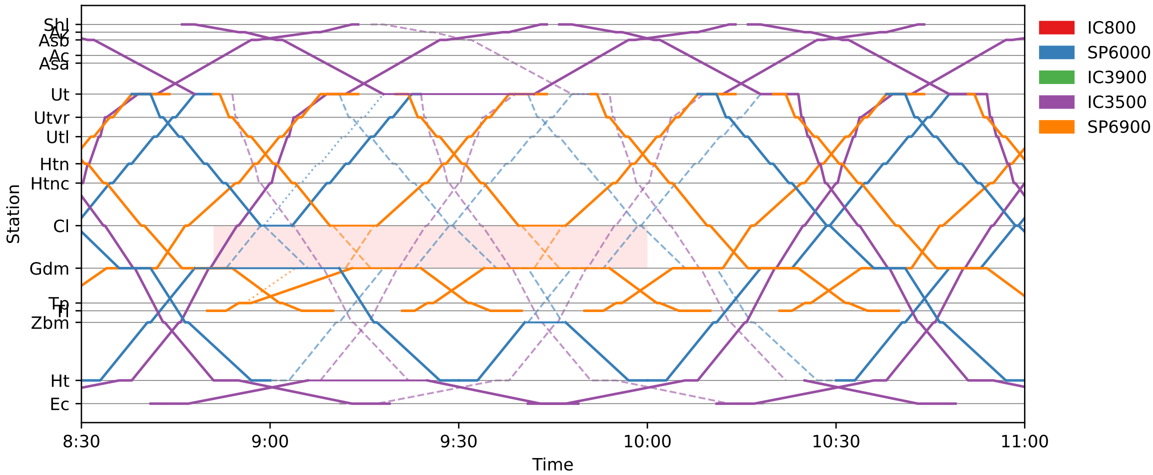


Figure A-5: Best found disruption timetable of ALNS for the small-scale case study.

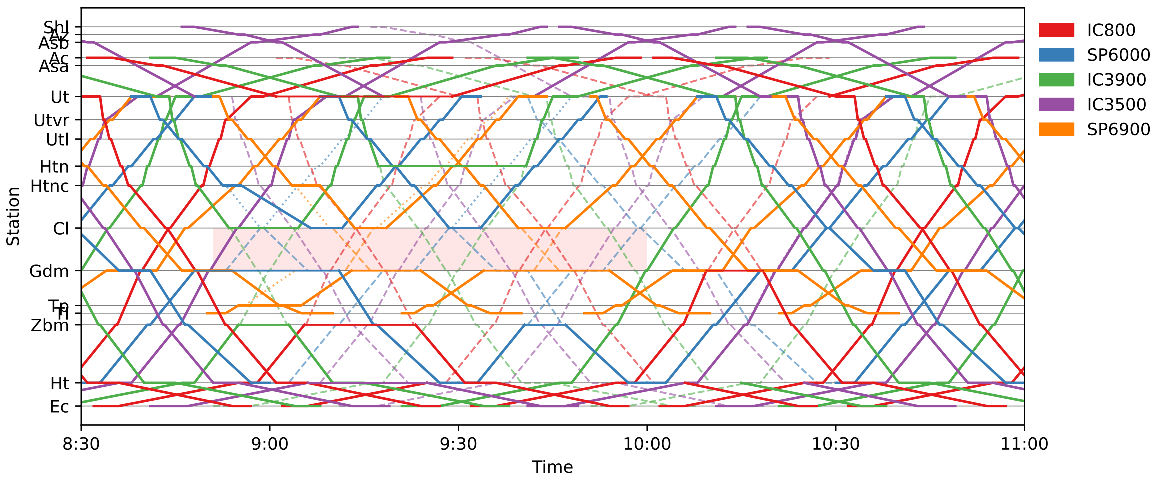


Figure A-6: Best found disruption timetable of ALNS for the large-scale case study.

Bibliography

A. O. Adewumi and O. J. Adeleke. A survey of recent advances in vehicle routing problems. *International Journal of System Assurance Engineering and Management*, 9(1):155–172, 2018.

T. Albrecht, I. A. Hansen, and J. Pachl. *Railway Timetable and Traffic: Analysis, Modelling, Simulation.* Eurailpress, 2008.

A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, 1999.

M. R. Bussieck, T. Winter, and U. T. Zimmermann. Discrete optimization in public rail transport. *Mathematical programming*, 79(1):415–444, 1997.

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

G. Cavone, T. van den Boom, L. Blenkers, M. Dotoli, C. Seatzu, and B. De Schutter. An mpc-based rescheduling algorithm for disruptions and disturbances in large-scale railway networks. *IEEE Transactions on Automation Science and Engineering*, 2020.

F. Chu and A. Oetting. Modeling capacity consumption considering disruption program characteristics and the transition phase to steady operations during disruptions. *Journal of Rail Transport Planning & Management*, 3(3):54–67, 2013.

F. Corman and L. Meng. A review of online dynamic models and algorithms for railway traffic management. *IEEE Transactions on Intelligent Transportation Systems*, 16(3):1274–1284, 2014.

X. Dong, D. Li, Y. Yin, S. Ding, and Z. Cao. Integrated optimization of train stop planning and timetabling for commuter railways with an extended adaptive large neighborhood search metaheuristic approach. *Transportation Research Part C: Emerging Technologies*, 117:102681, 2020.

W. Fang, S. Yang, and X. Yao. A survey on problem models and solution approaches to rescheduling in railway networks. *IEEE Transactions on Intelligent Transportation Systems*, 16(6):2997–3016, 2015.

N. Ghaemi, O. Cats, and R. M. Goverde. Railway disruption management challenges and possible solution directions. *Public Transport*, 9(1):343–364, 2017.

N. Ghaemi, O. Cats, and R. M. Goverde. Macroscopic multiple-station short-turning model in case of complete railway blockages. *Transportation Research Part C: Emerging Technologies*, 89:113–132, 2018.

K. Ghoseiri, F. Szidarovszky, and M. J. Asgharpour. A multi-objective train scheduling model and solution. *Transportation research part B: Methodological*, 38(10):927–952, 2004.

Y. Huang, C. Mannino, L. Yang, and T. Tang. Coupling time-indexed and big-m formulations for real-time train scheduling during metro service disruptions. *Transportation Research Part B: Methodological*, 133:38–61, 2020.

J. Jespersen-Groth, D. Potthoff, J. Clausen, D. Huisman, L. Kroon, G. Maróti, and M. N. Nielsen. *Disruption management in passenger railway transportation*, pages 399–421. Springer, 2009.

E. König. A review on railway delay management. *Public Transport*, 12(2):335–361, 2020.

I. Louwerse and D. Huisman. Adjusting a railway timetable in case of partial or complete blockades. *European journal of operational research*, 235(3):583–593, 2014.

C. E. Miller, A. W. Tucker, and R. A. Zemlin. Integer programming formulation of traveling salesman problems. *Journal of the ACM (JACM)*, 7(4):326–329, 1960.

S. Ropke and D. Pisinger. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation science*, 40(4):455–472, 2006.

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

P. Shaw. Using constraint programming and local search methods to solve vehicle routing problems. In *International conference on principles and practice of constraint programming*, pages 417–431. Springer, 1998.

L. P. Veelenturf, M. P. Kidd, V. Cacchiani, L. G. Kroon, and P. Toth. A railway timetable rescheduling approach for handling large-scale disruptions. *Transportation Science*, 50(3): 841–862, 2016.

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

Y. Wang, T. Tang, B. Ning, T. J. Van Den Boom, and B. De Schutter. Passenger-demands-oriented train scheduling for an urban rail transit network. *Transportation Research Part C: Emerging Technologies*, 60:1–23, 2015.

Y. Wang, A. D'Ariano, J. Yin, L. Meng, T. Tang, and B. Ning. Passenger demand oriented train scheduling and rolling stock circulation planning for an urban rail transit line. *Transportation Research Part B: Methodological*, 118:193–227, 2018.

X.-y. Xu, J. Liu, H.-y. Li, and J.-Q. Hu. Analysis of subway station capacity with the use of queueing theory. *Transportation research part C: emerging technologies*, 38:28–43, 2014.

J. Yin, L. Yang, T. Tang, Z. Gao, and B. Ran. Dynamic passenger demand oriented metro train scheduling with energy-efficiency and waiting time minimization: Mixed-integer linear programming approaches. *Transportation Research Part B: Methodological*, 97:182–213, 2017.

J. Yin, A. D'Ariano, Y. Wang, L. Yang, and T. Tang. Timetable coordination in a rail transit network with time-dependent passenger demand. *European Journal of Operational Research*, 295(1):183–202, 2021.

S. Zhan, L. G. Kroon, L. P. Veelenturf, and J. C. Wagenaar. Real-time high-speed train rescheduling in case of a complete blockage. *Transportation Research Part B: Methodological*, 78:182–201, 2015.

Y. Zhu and R. M. Goverde. Dynamic passenger assignment for major railway disruptions considering information interventions. *Networks and Spatial Economics*, 19(4):1249–1279, 2019.

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

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

Glossary

List of Acronyms

ALNS	Adaptive large neighborhood search
LNS	Large neighborhood search
MILP	Mixed-integer linear programming
VRP	Vehicle routing problem

List of Symbols

Activity sets

A_{boarding}	Set of boarding activities
A_{cap}	Set of capacity activities
A_{dwell}	Set of dwell activities
$A_{g,e}$	Set of activities part of route g, e
$A_{\text{head,ar,ar}}$	Set of arrival-arrival headways activities
$A_{\text{head,ar,de}}$	Set of arrival-arrival headways activities
$A_{\text{head,ar,de}}$	Set of arrival-departure headways activities
$A_{\text{head,de,de}}$	Set of departure-departure headways activities
A_{head}	Set of headways activities: $A_{\text{head}} = A_{\text{head,de,de}} \cup A_{\text{head,ar,ar}} \cup A_{\text{head,ar,de}}$
A_{odturn}	Set of short-turn activities at terminal stations, known as OD-turning
A_{pass}	Set of pass-through activities
A_{run}	Set of running activities
$A_{\text{fromshunt}}$	Set of move from shunting yard activities
A_{turn}	Set of short-turn activities
A_{shunting}	Set of shunting activities: $A_{\text{shunting}} = A_{\text{toshunt}} \cup A_{\text{fromshunt}}$
A_{station}	Set of activities related to stations: $A_{\text{station}} = A_{\text{dwell}} \cup A_{\text{pass}}$
A_{toshunt}	Set of move to shunting yard activities
A_{transfer}	Set of transfer activities
A_{waiting}	Set of waiting activities

Decision variables

$\alpha_{e,e'}$	Indicates whether event e is the previous event of e'
c_e	Cancellation of event e
$\epsilon_{e,e'}$	Event order between e and e' for station capacity
m_a	Short-turn activity selection
$\eta_{\text{can},e}$	Number of waiting passengers wanting that can board the train of departure event e
$\eta_{\text{can},g,e}$	Number of waiting passengers of passenger group g that can board the train of departure event e
$\eta_{\text{want},e}$	Number of waiting passengers wanting to board the train of departure event e
$\eta_{\text{want},g,e}$	Number of waiting passengers of passenger group g wanting to board the train of departure event e
$n_{\text{trans},g,e}$	Number of transferring passengers of group g at departure event e
$n_{\text{alight},e}$	Number of passengers alighting at departure event e
$n_{\text{remain},e}$	Remaining passenger capacity of the corresponding train of departure event e
$q_{e,e'}$	Train order between e and e' for headway activities
$u_{g,e}$	Route available for group g starting at event e
$w_{\text{after},g,e}$	Number of waiting passengers of group g immediately after departure event e

$w_{\text{before},g,e}$	Number of waiting passengers of group g immediately before departure event e
$w_{\text{before},e}$	Number of waiting passengers immediately before before event e
$w_{\text{before},g,e,e'}$	Number of waiting passenger waiting for event e after event e
x_e	Reschedule time of event e
ξ_s	Congestion level at station s

Events and activities

a	Activity
d_e	Operation direction of the train corresponding with e
e	Event
e'	Linked event
i_e	Number of rolling stock available for use at a shunting yard
κ_e	Railway track used by event e
λ_e	Related departure event of duplicate departure event e
$L_{a,\max}$	Maximum time duration of activity a
L_a	Minimum time duration of activity a
o_e	Original schedule time of event e
$s_{\text{next},e}$	Next station of event e
s_e	Station corresponding with event e
$\tau_{\text{line},e}$	Train line corresponding with event e
τ_e	Train service corresponding with event e
$\tau_{\text{type},e}$	Train type corresponding with event e

Miscellaneous

Δ	Maximum allowed difference between events for a destroy operator
D	Maximum allowed delay
L_a	Minimum time duration of activity a
Π_{s_e}	Number of duplicate departure events at station s_e
R	Maximum recovery time
s_{entry,d_e}	The entry station of the disruption in direction d_e
σ	Reward given to an operator when it is successful
t_{end}	End time of the disruption
t_{start}	Start time of the disruption
Ψ_{cancel}	Weight for cancellation of event e
Ψ_{delay}	Weight for reschedule time of event e
γ	Weight for cancellations of the request bank
Ψ_{sc}	Weight for congestion level ξ_s
w_j	Probability of selecting heuristic j
$\chi_{s,\tau_{\text{type}}}$	Station capacity of train type τ_{type}
$S_{\text{cap},s}$	Passenger capacity at station s

Passenger related

D_g	Destination station of passenger group g
g	Passenger group
O_g	Origin station of passenger group g
ζ_g	Number of passengers of passenger group g arriving per second
$N_{g,e}$	Number of activities in route g, e

Event sets

E_{arr}	Set of arrival events
E_{dep}	Set of departure events
E_{dum}	Set of dummy events
E_{dde}	Set of duplicate departure events
E_{exc}	Set of event that belong to trains that already departed from their starting station before the disruption occurs
E_{inv}	Set of inventory events
E_{train}	Set of events required to operate trains: $E_{\text{train}} = E_{\text{arr}} \cup E_{\text{dep}}$