Master's thesis

# Simultaneous optimization of rolling stock maintenance scheduling and rolling stock maintenance location choice

**Jordi Zomer**

August 2020

| | |
|---|---|
| **Author** | Jordi Zomer |
| **Student number** | 4631390 |
| | |
| **Date of defense** | August 18, 2020 |
| **Institution** | Delft University of Technology, Delft, Netherlands *(TU Delft)* |
| **Master's programme** | MSc Civil Engineering |
| **Master's track** | Transport & Planning |
| **Graduation committee** | Prof. dr. R.M.P. Goverde *(chair)* |
| | Dr. N. Bešinović *(daily supervision)* |
| | Dr. M.M. de Weerdt *(co-assessor)* |
| | Drs. J.J.H.M. Holtzer *(Nederlandse Spoorwegen)* |
| | Ir. W. Oldenziel *(SMA)* |

# Thanks

Looking back at the entire process that has led to the current master's thesis, there are numerous reasons to say thanks.

I am thankful for the growth that I have experienced in the process of writing my master's thesis. Not only have I seen the research itself mature, become more to the point, improve in applicability, augment in versatility - which has been a joy to observe -, also I have watched myself develop, in critical thinking, in systematic problem solving, in clear writing, in solid argumentation. Witnessing it has been fascinating and humbling at the same time.

Also, I am thankful for all support that I have received - from very many people in various ways along the road, but in particular from my daily supervisors. I am amazed by the proportion of their time that they have given me. Experiencing their commitment to me and the research has been very special. Their encouragement during harder times and their in-depth, critical feedback has been of crucial significance.

And I am thankful for all family and friends that were there during the writing of the project. The coffee breaks in the library, the video calls from abroad, the balcony sessions in the evening sun - they have been such a relief. Having people near whose appreciation and love I need not doubt, is inexpressibly valuable.

And this is only a small fragment out of the countless reasons for gratitude.

Thank You.

*Jordi Zomer*
*Zwolle, August 2020*

# Contents

# Executive summary


### Trends

The current research addresses a problem found in the area of railway operations regarding the maintenance of rolling stock units. It focuses on the situation in The Netherlands and approaches the problem from the perspective of its main railway operator *N.V. Nederlandse Spoorwegen* (NS).

In The Netherlands, railway transport is an increasingly important mode of transport. The proportion of all passenger transport via rail has increased with 2.2 percentage points between 2008 and 2017, and the number of passenger kilometers via rail with 30% between 1996 and 2017. These trends induce two developments: the number of rolling stock units that operate on railway networks grows, and simultaneously the proportion of time these rolling stock units are utilized for passenger trips increases. As a result, the use of the capacity of the railway network increases.

To sustain a reliable and comfortable railway service, rolling stock units need to undergo small maintenance activities at regular intervals. Examples of such maintenance activities are the *Technical B-check*, which takes approximately 10 minutes and needs to be performed in general every 48 hours, and *Interior cleaning*, which takes approximately 60 minutes and needs to be performed in general every 24 hours. Maintenance activities like these need to be performed at one of the 35 maintenance locations in The Netherlands.

### Problem and current research

The increasing use of the capacity of the railway network leads to two issues.

The first of these issues relates to the scheduling of maintenance activities, which is traditionally performed mainly manually. Due to the increased utilization of rolling stock units and the fact that increasingly many rolling stock units operate on the network, the complexity of this scheduling process is increasing. This raises the need for tools that automate the maintenance scheduling process.

The second issue concerns maintenance location choice. Rolling stock units are traditionally maintained during nighttime. Due to the increasing number of rolling stock units operating on the railway network, the use of the capacity of maintenance locations during nighttime is under pressure. As a result, NS is considering to perform more maintenance activities during daytime. This raises the question at which locations maintenance teams needs to be stationed to perform daytime maintenance, referred to as the maintenance location choice.

The two issues mentioned above are interrelated. On the one hand, the maintenance schedule depends on the locations that can be used for daytime maintenance. On the other hand, the locations that can best be used for daytime maintenance depend on the schedule that can be created for each choice of locations. Due to this mutual dependency, both aspects need to be addressed simultaneously.

The current research aims to find an optimal maintenance schedule and a maintenance location choice in an automated way. A maintenance schedule and maintenance location choice are considered to be optimal if they achieve the most important goal NS is currently facing: reducing the amount of work that needs to be performed during nighttime. The method should be efficient, meaning that a solution can be found within several hours of computation time.

**Model development**

The model development in the current research can be understood as a three-stage framework.

The first stage regards the problem that lies at the core of the research: the *Maintenance Scheduling and Location Choice Problem* (**MSLCP**). Given a rolling stock circulation, its goal is to find an optimal maintenance schedule and maintenance location choice - optimal in the sense of minimizing the total number of nighttime maintenance activities. A schematical representation of the **MSLCP** is given in Figure 1.


*Figure 1: The characterizing input and output of the **MSLCP**: using a rolling stock circulation, an optimal maintenance location choice and an optimal maintenance schedule are determined.*

The second stage introduces the *Activity Planning Problem* (**APP**). It is acknowledged that the **MSLCP** does not take into account the capacity of maintenance locations, that is, it is not able to provide any information on the number of maintenance teams required. Moreover, it does not provide an activity planning, determining exactly which maintenance activity needs to be performed when and by which maintenance team. The developed **APP** model aims to address these issues by providing both the required capacity as the activity planning. This is useful in various ways. First, it can be used to post-process any **MSLCP** solution determine the corresponding required capacity. Moreover, it can determine an activity planning that is useful for operations. The input and output of the **APP** and its relation to the **MSLCP** is schematically depicted in Figure 2.

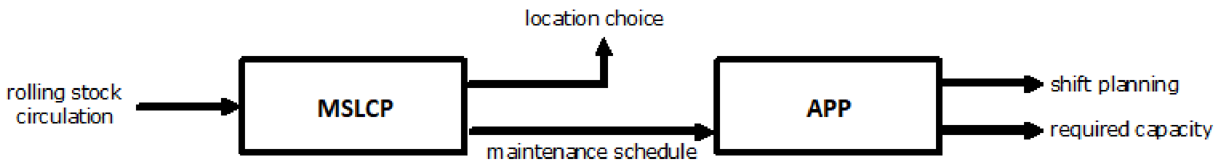



*Figure 2: Graphical representation of the input and output of the **APP** and how this functions in conjunction with the **MSLCP**.*

The third stage integrates the first and second stage by proposing the *Capacitated Maintenance Scheduling and Location Choice Problem* (**CMSLCP**). Observe that the introduction of the **APP** enables to assess the required capacity of any **MSLCP** solution. However, in many practical situations, a solution to the **MSLCP** is required that already takes into account the available capacity of maintenance locations. The **CMSLCP** is an extension to the **MSLCP**, aiming to find a solution to the **MSLCP** that satisfies some predetermined constraints on the capacity of maintenance locations. A schematical representation of this method is presented in

Figure 3, allowing for comparison with the first two stages. An approach called *Logic-Based Benders' Decomposition* (LBBD) is used, separating the problem into a master problem (the **MSLCP**) and a sub-problem (the **APP**), which are iteratively solved. An **MSLCP** solution is created, its capacity is assessed using the **APP** and this information is used to obtain a new **MSLCP** solution, until a **MSLCP** solution is reached in which maintenance location capacity constraints are satisfied. An important sub-routine of the **CMSLCP** is the cut generation process, for which four variants are provided: the naive, Basic Heuristic, Binary Search Heuristic and min-cut cut generation processes.

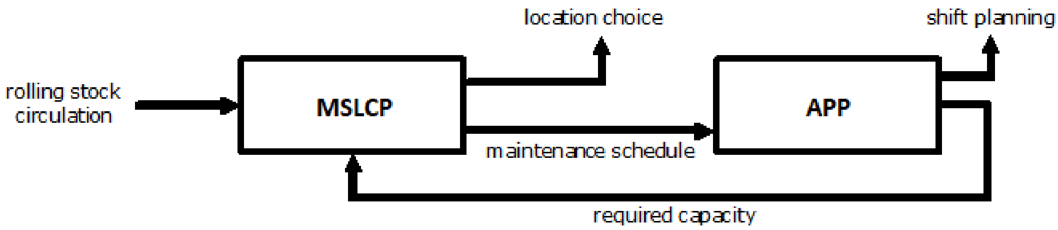



*Figure 3: Graphical representation of the **CMSLCP**, demonstrating how it combines the **MSLCP** and the **APP**.*

**Results**

The **MSLCP** model is applied to many different scenarios. Based on these scenarios, a vast amount of experiments has been conducted, addressing various aspects of the problem. Three of the key results are highlighted. First, it is shown that increasing the number of maintenance locations for daytime maintenance also increases the *day share*, a figure representing the proportion of all hours of maintenance activity that is performed during daytime. For a scenario with 5 maintenance locations for daytime maintenance, the day share increases up to 22.3% and for 20 maintenance locations for daytime maintenance up to 42.0%, for the scenarios with all rolling stock units of types VIRM4 and VIRM6. Second, the location choice is compared for various scenarios and it is shown that the location choice is consistent for different lengths of planning horizons, for different input data sets, and for different maintenance durations. Third, the **MSLCP** has been applied to a large scenario including all rolling stock units for intercity services. It is shown that the day share in this case is 22.2% when 10 locations are opened for daytime maintenance and 30.1% when 20 locations are opened for daytime maintenance. The largest encountered running time was 3 hours and 12 minutes.

Also results for the **CMSLCP** (and the **APP**, which is a subroutine of the former) are generated. A realistic problem instance is considered that serves to demonstrate the workings of the **CMSLCP** and to provide insight in its efficiency. The model has been applied to all variants of the cut generation process. It is shown that the heuristic cut generation process with 15 cuts per iteration converges quickest on a particularly hard instance, outperforming the naive, min-cut and other heuristic cut generation processes. On the other hand, on a larger instance, focusing on capacity violations of multiple maintenance shifts, the min-cut cut generation process has shown to be the best method to reach a reasonable solution quickly: in the considered problem instance, the number of maintenance shifts for which the required capacity exceeds the available capacity is reduced from 21 to 5 in 7.6 minutes, compared to 44.2 minutes when the the binary search heuristic cut generation method is used.

In addition to the aforementioned results on the **MSLCP** and **CMSLCP**, a more practical approach is taken by constructing a small use case, demonstrating how the current research can be applied in practical situations. To this end, *Viriato* is used, by which various visualizations of maintenance schedules created by the **MSLCP** can be provided. An example of a visualization that can be obtained using Viriato is given in Figure 4, clearly representing both a rolling stock circulation as any assigned maintenance activities that it contains.

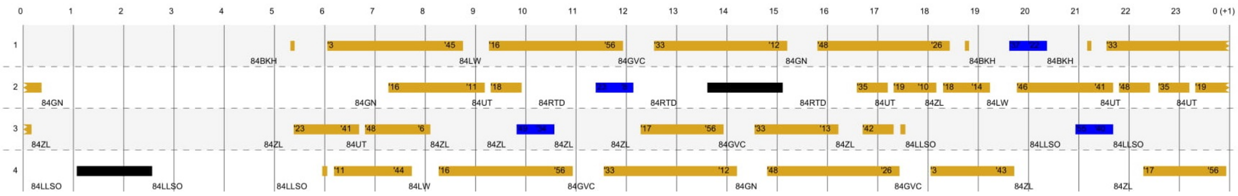


*Figure 4: Maintenance schedule for a given rolling stock circulation, computed using the **MSLCP** and visualized using Viriato.*

### Discussion

The current research is relevant from both a scientific as well as a practical perspective in multiple ways. The three developed all form a contribution to the scientific literature. This is especially true for the **CMSLCP**, which integrates complex capacity constraints in the **MSLCP** context by using the LBBD method, opening up new interesting research areas. Moreover, the models can be applied in practice for multiple objectives in practice. First and foremost, it is intended for use on a tactical level, relating to the planning a few weeks or months before operations. However, its relevance extends also to the operational level; especially the developed **APP** can be used in that sense to determine a feasible maintenance shift planning. Moreover, the models in the research can be used on a strategic level by analysing various scenarios on long-term developments.

In addition, the current research is relevant to other fields of application. The models and findings in the current research are expected to be able to be easily adapted for application to other countries or to other contexts with different objectives. Moreover, the use of the current research may extend to other areas in which planning problems prevail, such as aviation, postal delivery or more general problem classes that relate to the scheduling of activities with maximum intervals lengths between these activities.

It must be noted that the current research assumes that the rolling stock circulation is given. Therefore, it cannot be used for applications where the rolling stock circulation is required to vary. An example of such a case is the scheduling of less frequent heavy maintenance activities: in the scheduling of these activities, the rolling stock circulation is usually adapted during operations in order to be able to plan the maintenance activity, a feature not supported by the current models.

Also, the use of Viriato in research has been discussed. It is indicated that Viriato is especially relevant in bridging the gap between research and practice since models developed in research can be easily conveyed to day-to-day railway operations. Moreover, Viriato is able to quickly provide visualizations of solutions, from which researchers may also benefit. As a downside, it can be mentioned that Viriato is not intended for the generation of large batches of scenarios, an application often used in research.

### Conclusion

The current research has provided multiple models that all contribute to the same goal: the scheduling of maintenance activities. The **MSLCP** has been shown to deliver a proper maintenance schedule and to provide a consistent location choice. Moreover, using this model it has been shown that the current pressure on maintenance locations can be reduced up to 30.1% when considering all rolling stock units used for intercity services. The **APP** has shown operational usability and performs within seconds of computation time. The **CMSLCP** is a relevant contribution to the literature, and various cut generation methods have been designed for it to improve the computation time. To date, for one problem instance it has been shown that it is able to find an **MSLCP** solution that decreases the number of maintenance shifts from 21 to 5 within 7.6 minutes.

# Chapter 1

# Introduction

The current research addresses a problem found in the area of maintenance planning in the railway industry. This chapter aims to provide context for the problem that is considered and motivate its relevance. Section 1.1 exposes some trends that indicate increasing usage of railway networks, after which Section 1.2 indicates what railway networks are in general composed of. Sections 1.3, 1.4 and 1.5 explain the maintenance process in general, and Section 1.6 compares these maintenance processes for different countries. Section 1.7 identifies the problem and Section 1.8 explains how the current work addresses the identified problem. An outline of the current research is given in Section 1.9.

## 1.1 Trends in the railway industry

For many countries, rail transport is an increasingly important mode of transport. This can be deduced from, for example, the *modal split*: the relative percentage of all passenger transport that is performed by train, relative to other modes of transport. Table 1.1 indicates the development of the modal split for train transport over several years and shows that train transport becomes relatively more important. Several European countries have been considered. [1]

| | 2008 | 2017 | Δ (% pts) |
|---|---|---|---|
| European Union | 7.4 | 7.9 | +0.5 |
| France | 10.1 | 10.8 | +0.7 |
| Germany | 8.1 | 8.7 | +0.6 |
| United Kingdom | 6.9 | 8.8 | +1.9 |
| Italy | 6.0 | 5.9 | -0.1 |
| Spain | 5.5 | 7.0 | +1.6 |
| Switzerland | 17.1 | 19.7 | +2.6 |
| Netherlands | 9.2 | 11.4 | +2.2 |
| Poland | 8.2 | 7.7 | -0.5 |
| Austria | 11.1 | 11.9 | +0.8 |
| Belgium | 7.5 | 7.8 | +0.3 |

*Table 1.1: Modal split for train transport in % of all passenger transport in several European countries.*

Together with the gradual shift from other transport modalities to rail transport, the demand for railway transport is currently increasing. The demand for railway transport can be

[1] The list includes the 10 countries with the highest number of passenger kilometers in 2014, according to International Union of Railways (2018), that participate on the European Single Market, according to gov.uk (2020), in descending order of the number of passenger kilometers.

expressed in the number of passenger-kilometres. The development of this indicator is shown in Figure 1.1. For most countries, a clear positive trend is observed.

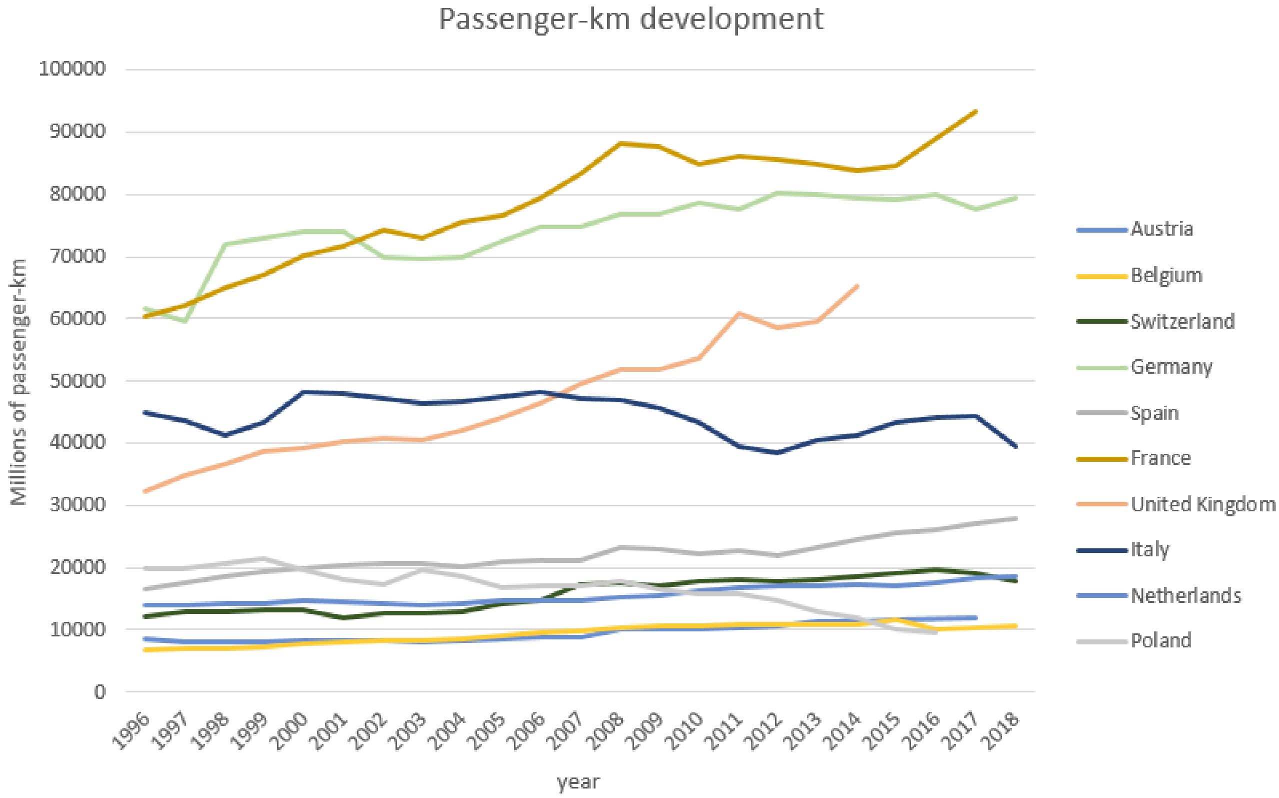



*Figure 1.1: Development of passenger-kilometres in several European countries from 1995 to 2018 (International Union of Railways, 2018).*

A comparison between 1996 and 2018 is made in Table 1.2, indicating that for almost all countries demand for railway transport has increased significantly in the last decades.

To accommodate increasing travel demand, two possible developments may be observed. Firstly, the utilization of the available rolling stock units may increase. This is the case in The Netherlands: Figure 1.2 demonstrates that the rolling stock utilization of the most important Dutch railway operator *Nederlandse Spoorwegen N.V.* (NS) has increased over the past years. Secondly, more rolling stock units may be added to the network. This is observed in for example The Netherlands (NS, 2018a, p. 5) and Germany (Deutsche Bahn, 2018, p. 9).

Hence, more rolling stock units may be added to the network with possibly higher utilization to meet higher passenger demand - a phenomenon at least observed in the Netherlands. As a result, railway networks become more heavily used and the flexibility to deliver a feasible plan decreases.

## 1.2 Railway networks and rolling stock

Rail transport makes use of railway networks composed of stations that are connected by steel railway tracks. Figure 1.3 indicates the number of stations and the total track length for the railway networks of European countries. It reveals that European railway networks vary widely by the number of stations and the total track length.

| Country | Pax-km (%Δ) |
|---|---|
| France | +55 |
| Germany | +62 |
| United Kingdom | +102* |
| Italy | -1 |
| Spain | +62 |
| Switzerland | +59 |
| Netherlands | +30 |
| Poland | -52* |
| Austria | +42 |
| Belgium | +53 |

*Table 1.2: Increase in passenger-kilometres (pax-km) over several years between 1996 and 2017. Due to incomplete data for the United Kingdom and Poland, a comparison between 1996 and 2014 is given for the United Kingdom and a comparison between 1996 and 2016 is given for Poland.*

On railway networks, rolling stock operates. Rolling stock includes includes all units that move over the steel railway tracks, such as locomotives, passenger wagons and freight wagons. Some types of rolling stock are intended for passenger transport and other types of rolling stock are intended for freight transport

Rolling stock intended for passenger transport stops at stations allowing passengers to embark and disembark. Rolling stock units (sometimes also referred to as trains, train sets or railway vehicles) are intended for the transportation of multiple passengers at the same time. A rolling stock unit are fixed compositions that consist of various carriages. Multiple rolling stock units may be connected to each other forming a rolling stock combination. In most cases, various types of rolling stock operate on the network, with different characteristics, such as acceleration, seating capacity, power supply and train protection systems. Figure 1.4 shows a rolling stock combination that is used by NS. It pictures two rolling stock units of type VIRM4, each of them having four carriages. Rolling stock units can be coupled and decoupled during operations in several minutes and hence routed individually if desired.

## 1.3 Rolling stock maintenance

In order for a railway network to function properly, the rolling stock that operates on the railway network needs to receive maintenance on a regular basis. The aim of maintenance is to ensure that the rolling stock that operates on the network remains available (to ensure a reliable train service), safe and comfortable for passengers (Dinmohammadi et al., 2016). To this end, various types of rolling stock maintenance activities exist. Some of these maintenance types, such as technical checks, are intended to ensure rolling stock safety and reliability, whereas other maintenance types, such as cleaning activities, are intended to ensure passenger comfort.

Maintenance can be divided into two categories: regular maintenance (also referred to as low-level maintenance) corresponding to the maintenance activities with higher frequencies (every 1 to 14 days) and shorter duration (1-3 hours), and heavy maintenance (also referred to as high-level maintenance), corresponding to maintenance types with lower frequencies (every several months or less) and longer duration (up to several days) - see for example Andrés et al. (2015). Since regular maintenance and heavy maintenance have such a different nature, the way they are planned may also differ. Regular maintenance may be performed whenever a rolling stock unit has a planned standstill; heavy maintenance, however, requires a rolling

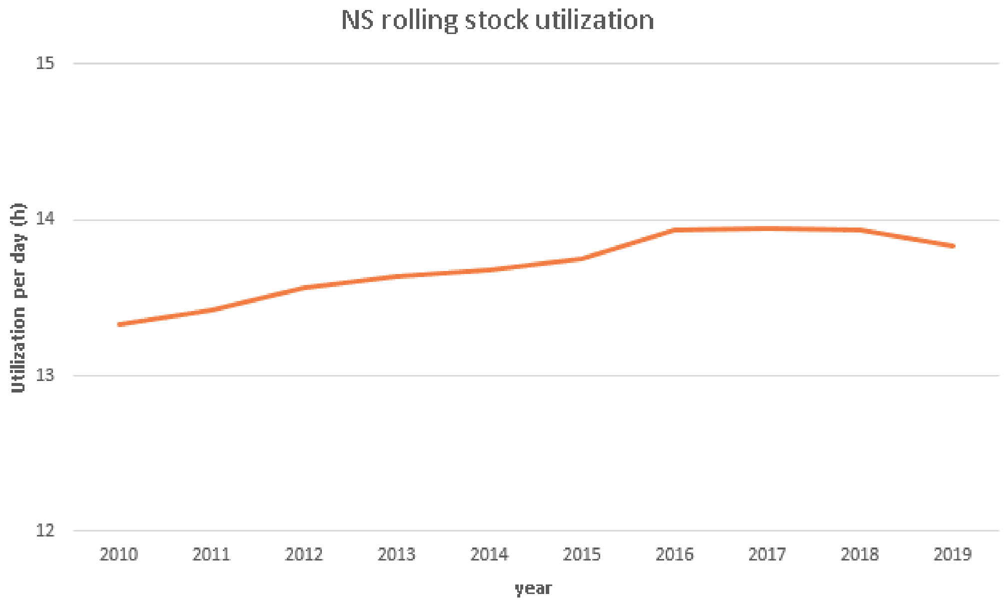



*Figure 1.2: Rolling stock utilization for Dutch Railways in 2010-2019 in hours per day. Rolling stock is considered utilized when it is used for either a commercial or empty trip.*

stock unit to be completely out of regular service. Appointments for heavy maintenance activities may be made in advance; dispatchers then need to make sure that the rolling stock unit arrives in time at the maintenance location to receive heavy maintenance (Bonne, 2020).

Table 1.3 gives an overview of the types of regular rolling stock maintenance that apply at the main railway operator in The Netherlands, *Dutch Railways* (Nederlandse Spoorwegen, NS). The Technical B-check is a check regarding, among others, the brakes and driver cabin inventory. The Technical A-check is a more extensive technical check containing the contents of the Technical B-check with in addition thorough checks of the electronic equipment. Interior cleaning regards passenger comfort and includes toilet cleaning, the emptying of garbage bins and the cleaning of the floors. Exterior cleaning which entails routing the rolling stock through a washing unit at low speed (comparable to a car wash system).

| Maintenance type | Average duration | Maximum interval |
|---|---|---|
| Technical A-check | 60 minutes | 12 days |
| Technical B-check | 10 minutes | 48 hours |
| Interior cleaning | 60 minutes | 24 hours |
| Exterior cleaning | 30 minutes | 14 days |

*Table 1.3: Different types of regular maintenance at NS (NS, 2017).*

In general, the maximum interval between consecutive maintenance activities is governed by strict rules that are imposed by railway authorities. These rules result in deadlines for each maintenance activity. If an operator fails to carry out a maintenance activity on a rolling stock unit before the deadline, this may lead to the rolling stock unit being taken out of service. As a result, an operator may not be capable of offering all train services promised in the timetable. It is therefore essential that maintenance activities are carried out on time.

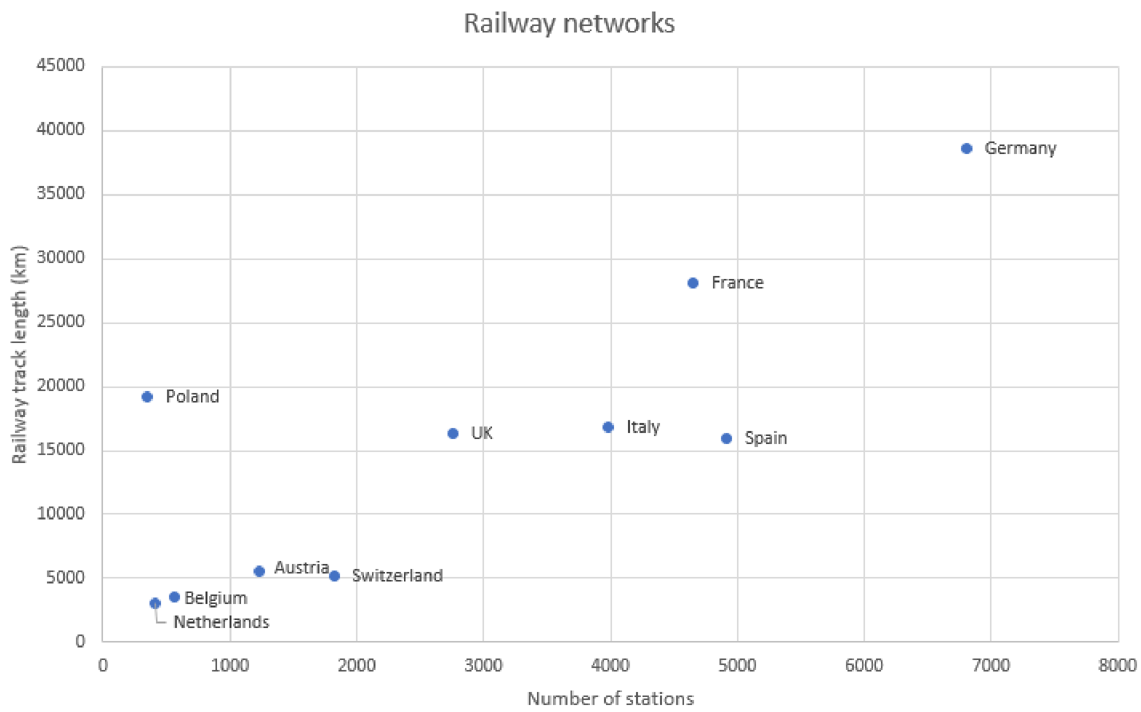



*Figure 1.3: Railway network sizes for various European countries. Data for track length comes from Eurostat (2020) and corresponds to total track length in a country in 2017, except for Belgium (2009), Switzerland (2015) and Italy (2016). Data for the number of stations comes from opendatasoft.com (2020).*

*Figure 1.4: Rolling stock combination operating on the Dutch network, consisting of two rolling units of type VIRM4, each consisting of four carriages (Treinposities.nl, 2020).*

## 1.4 Maintenance locations

Maintenance activities are carried out at so-called maintenance locations, which are railway yards with maintenance facilities. Maintenance locations are spread over the network. In The Netherlands, 35 maintenance locations exist (NS, 2019). A map indicating the maintenance locations is found in Figure 1.5.

Maintenance locations have different layouts, depending on, among other things, its size and the local geography. As a result, capacities vary widely over various maintenance locations.

An important difference among several maintenance locations is that some have more track length than others. Also, there may exist differences in the types of maintenance that can be performed at maintenance locations (Bonne, 2020; Kuhlmann, 2020; Blanc, 2020). For example, in practice it happens that not all locations are suitable for exterior cleaning, since this requires that the maintenance location is equipped with a washing installation (NS, 2019). In addition, the layout of maintenance locations may differ (Huizingh, 2018). For example, some maintenance locations are designed with the intention that all maintenance activities are performed at the place where a rolling stock unit is positioned. In this design, the position of the train remains fixed and the resources needed for maintenance activities are moved to this position. Other maintenance locations are designed with the intention that rolling stock

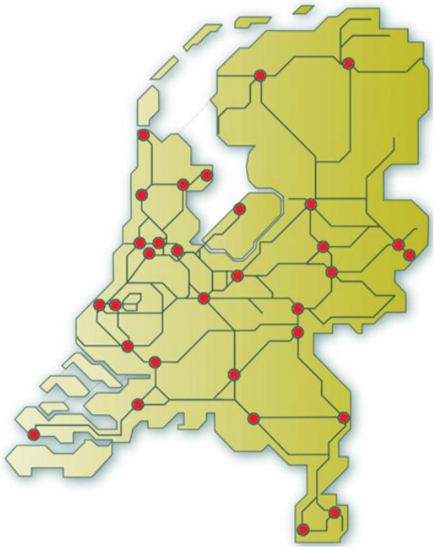


*Figure 1.5: Maintenance locations in The Netherlands (Janssens, 2017). Some maintenance locations that are close together are indicated by the same node; therefore the number of nodes indicated in this figure is somewhat lower than the actual number of distinct maintenance locations. Maintenance locations are used mainly by NS, but the other (smaller) operators on the network also use these facilities.*

units are moved to locations where specific maintenance activities can be carried out. In this design, the positions of the resources needed for maintenance activities are fixed, and the rollin g is moved to these positions. The advantage of the latter form over the former is that the maintenance resources can be located at a single spot only, since rolling stock units will move to this position to receive maintenance. This reduces the total amount of maintenance resources necessary and avoiding the necessity of moving maintenance resources during operations. However, in maintenance locations with this kind of design, usually many shunting movements are necessary to route rolling stock units to the positions where maintenance can be carried out, which is usually deemed undesirable.

When a rolling stock unit needs to be maintained, it is routed from a nearby station to the maintenance location. To this end, a track needs to be available that the rolling stock unit can use to go to the maintenance location. This availability is not straightforward, especially during daytime, since other rolling stock units (usually with passengers) need to use the track as well and get priority over empty rolling stock units.

For a maintenance location to be operational, it is necessary that personnel is stationed at a maintenance location. Often, specific types of maintenance are performed by specific types of personnel. For example, in general the type of personnel that performs interior cleaning is not the same type of personnel that performs technical checks. The number of personnel stationed at each location is the operator's decision and determines, together with the maintenance location design, the capacity of a maintenance location. This capacity need not be constant over the day: an operator can choose to open a facility at certain moments of the day or night by stationing personnel at this location. In particular, a distinction can be made between daytime operations (which means that a location is opened during the day) and nighttime operations (which means that a location is opened during the night). This distinction is clearly visible in The Netherlands where maintenance is usually carried out during nighttime.

The choice of which maintenance locations are open is not straightforward. This entails the stationing of personnel at maintenance locations. In this way, the operator has control over the opening times of maintenance locations.

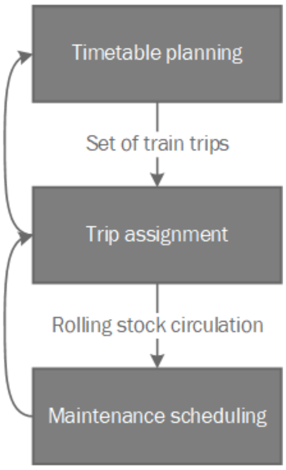



*Figure 1.6: Simplified representation of the rolling stock planning process.*

## 1.5 Railway planning

The scheduling of maintenance activities is part of the larger framework of railway planning. To understand the nature of maintenance scheduling, it is necessary to consider the broader framework.

To give a general understanding of the planning process, the process may be categorized in three steps, indicated in Figure 1.6. These three steps are a simplification of reality. First, based on the (predicted) travel demand and the available resources a railway operator has at its disposal, a timetable is designed. This timetable specifies all trips that are to be offered to passengers in the coming period. These trips need to be connected in such a way that feasible sequences of trips are obtained that can be performed by individual rolling stock units. This is done in the second stage: available resources are assigned to trips. This stage results in feasible trips for each rolling stock unit such that all trips in the timetable are covered and is referred to as the *rolling stock circulation*. When the rolling stock circulation is determined, the third planning step can be executed, where maintenance activities are scheduled in the rolling stock circulation. There are some feedback loops between the various steps: for example, if problems show up in the trip assignment phase, adjustments can be made in the timetable, or if maintenance scheduling problems are identified, the trip assignment phase can be revisited. Observe that the dispatching of specific rolling stock units to train trips is done after the planning process of Figure 1.6 and described above. The above process yields trips for 'some' rolling stock unit, but not determines which exact rolling stock unit will be used.

In practice, many disruptions take place that may disturb the rolling stock circulation. For example, Zomer (2019) concludes that for NS, the exact locations of rolling stock units is - despite the existence of a precise schedule - highly unreliable up to even half a day, due to disruptions of different kinds. Examples of disruptions are rolling stock defects, track defects, wayside equipment defects, collisions or personnel shortages. These disruptions influence the rolling stock circulation of the affected rolling stock unit. However, as a result of such a disruption, the rolling stock circulation is often adjusted, for example by assigning different rolling stock unit to some train trips than according to the original planning. Hence, a disruption influences the rolling stock circulation of multiple rolling stock units and this, in turn, influences the maintenance schedule of these rolling stock units. Therefore, in the planning process it is important to incorporate robustness of the plan under disruptions.

In practice, the maintenance schedule is often created manually. Typically, a lot of personnel capacity is required for the planning department. At NS, for example, approximately 150 full-time jobs are concerned with planning (Voet, 2020), which is costly. As a result, there is an increasing demand for automated decision-support tools to assist planning personnel.

## 1.6 International comparison

Different countries have different maintenance policies. The current section provides some insights in how the maintenance policies of different European railway operators compare. It presents a comparison between railway operator NS (The Netherlands), DB Regio Bavaria (Germany), NMBS (Belgium) and SNCF Voyages (France).

The information for this comparison originates from the following sources: Kuhlmann (2020) for DB, Bonne (2020) for NMBS, Blanc (2020) for SNCF. Kuhlmann (2020); Bonne (2020); Blanc (2020) are personal interviews for which summaries can be found in Appendix A.

The current analysis gives an overview of the most important and prominent maintenance types in each country under investigation. Heavy maintenance, which typically occurs with long intervals and takes multiple days, is not included in the current analysis.

Table 1.4 gives some general characteristics of the current maintenance process in the countries under consideration. It shows that in the NS, SNCF Voyages and DB Regio Bavaria capacity issues are reported. Also, it shows per country whether maintenance is focused on the day or on the night. Moreover it gives an overview of the types of maintenance locations that exist and how many of them are currently found in the network. This excludes those locations that can only be used for heavy maintenance.

| Railway operator | Capacity issues | Daytime/nighttime | Maintenance locations |
|---|---|---|---|
| NS (Netherlands) | yes | Nighttime | Regular maintenance locations (RML, 23); maintenance locations with washing equipment (RMLW, 12) |
| NMBS (Belgium) | no | Traditionally daytime, now shifts to combination of daytime and nighttime | Traction locations (TL, 9); Regular maintenance locations (RML, 19) |
| SNCF Voyages (France) | yes | Daytime and nighttime | Large maintenance locations near Paris (P, 4); somewhat smaller maintenance locations near other cities (NP, 3) |
| DB Regio Bavaria (Germany) | yes | Daytime and nighttime; varies strongly between maintenance locations | Yards (Y, 30-40), maintenance locations (ML, 6-7) |

*Table 1.4: Maintenance policies in European countries.*

First, it can be noted that three out of four investigated countries report capacity problems at maintenance locations. Second, it shows that it varies from country to country whether maintenance is focused on daytime or nighttime. Although NS has a tradition of nighttime maintenance and makes a shift to include more daytime maintenance, for other railway operators (such as NMBS) an opposite situation applies: they have a tradition of daytime maintenance and make a shift towards more nighttime maintenance. SNCF Voyages and DB Regio Bavaria have a practice of both daytime and nighttime maintenance, depending on the maintenance location. The origin of this lies in the fact that both companies face peak demands due to commuting traffic: many rolling stock entering a city at the start of the day and leaving the city at the end of the day (Kuhlmann, 2020; Blanc, 2020). This results in many rolling stock units available for maintenance during the day. Third, it can be noted that different levels of maintenance location types apply in each country.

Table 1.5 lists for each country the following things: the maintenance type, the interval and duration, the location where a maintenance type can be performed, and whether this maintenance type is part of the planning process. Since the actual duration and interval often vary for each rolling stock type, an approximated value applicable to most rolling stock types is given. The location type corresponds to the locations listed in Table 1.4. The column 'incorporated in planning' means that the rolling stock circulation takes into account maintenance time windows for this type of maintenance activity.

| | Type | Interval | Duration | Location | Planned |
|---|---|---|---|---|---|
| **NS** | Cleaning | 1 d | 60 min | RML, RMLW | no |
| | Technical B-check | 2 d | 10 min | RML, RMLW | no |
| | Technical A-check | 12 d | 60 min | RML, RMLW | no |
| **DB Regio Bavaria** | Technical inspections | 24 hrs | 10 min | Stations | no |
| | Cleaning checks | 2 d | 1-2 hrs | Y | no |
| | Larger techn. checks | 10 days | 8-12 hrs | ML | yes |
| **NMBS** | Cleaning, lowest level | 6 hrs | 30 min | RML | no |
| | Daily investigation | 1 d | 10 in | Stations | no |
| | Limited investigation | 7 d | 1 hrs | Stations | no |
| | Thorough check | 30 d | 8 hrs | TL | yes |
| **SNCF Voyages** | Interior cleaning | After each trip | 5-30 min | Stations | no |
| | Toilet cleaning | 2-3 d | 1 hrs | P, NP | yes |
| | Level 1 | 3-4 d | 2-3 hrs | P, NP | yes |
| | Level 2 | 30 d | 12-24 hrs | P | yes |

*Table 1.5: Overview of maintenance activities in European countries.*

The following observations can be made. First, checks at different levels exist. Usually shorter checks have to be performed regularly and longer checks less regularly. Second, the intervals for most maintenance checks are based on time. An exception is the Level 1 maintenance check in France, which is based on mileage. Third, not all maintenance activities can be performed at all locations. In general, some locations may be suitable for one type of maintenance whereas other locations may be suitable for other types of maintenance locations. Fourth, many maintenance activities are not incorporated in the planning process. In such a case, the adherence of maintenance intervals is monitored. When a deadline is almost exceeded, the dispatcher is responsible that the rolling stock unit receives the necessary maintenance.

## 1.7 Problem identification

The problem at hand can be viewed from two perspectives. Section 1.7.1 describes the problem from a general perspective, applicable to most railway operators. Section 1.7.2 slightly changes the standpoint and further examines the problem from the perspective of Dutch Railway operator NS.

### 1.7.1 General problem

The maintenance scheduling phase described in Section 1.5 is highly complex, especially in the case of The Netherlands, but also for operators in other countries (Bonne, 2020; Kuhlmann, 2020; Blanc, 2020). There are various types of maintenance that all have a different periodicity. Moreover, there are many rolling stock units that all need maintenance, but the locations and moments when maintenance can be carried out are not straightforward. Also, these rolling stock units may operate on the entirety of the network, such that a local planning approach does not yield desirable results: instead, a network-approach has to be taken. In this complicated context, operators want to prevent doing too much maintenance as this is costly. Furthermore, as a consequence of the trends mentioned in Section 1.1, the complexity of the railway planning process is increasing. The demand for rail transport is increasing in most European countries. Due to the rolling stock purchases there are more rolling stock units that need to be maintained. At the same time, due to the increased utilization of rolling stock units for passenger trips, the rolling stock units are out-of-service less often and hence, there are less opportunities to carry out this maintenance. As a result, the complexity of the maintenance planning process is accumulating.

In addition, the desirability of the maintenance schedule depends on the locations that are open for maintenance. The desirability of a maintenance schedule may be measured in for example total cost or the total number of maintenance activities. Some choices for open maintenance locations result in more desirable maintenance schedules than others. Typically, the choice concerning when each location is opened is given, but these choices need not be optimal for the current rolling stock circulation. In other words, the optimal maintenance schedule is influenced by the maintenance locations that can be used, and vice versa, the optimal maintenance locations are influenced by the maintenance schedule that can be created.

Although complicated, the maintenance scheduling process and the maintenance location choice are utterly relevant. Maintenance is performed very often and consumes a large part of the budget of railway operators. Also, maintenance is essential for the driving of rolling stock units: without technical checks, rolling stock units usually are not allowed to drive. Moreover, since the pressure on the railway network is increasing and the planning flexibility is decreasing correspondingly, it is becoming increasingly difficult and time-consuming to deliver a plan since the planning is usually created manually. Besides, the choice where maintenance needs to be carried out is not straightforward. Hence, there is an increasing need for intelligent, automated tools to support the planning process.

### 1.7.2 Problem in the context of NS

At NS, most rolling stock maintenance is performed during nights. This finds its origin in the fact that rolling stock units usually have a long planned standstill during the night at some location. Consequently, the capacity pressure on rolling stock maintenance locations during nights is increasing. As a result, NS is considering to perform more maintenance activities during daytime. This potential policy switch, however, comes with two major complicating factors.

First, performing maintenance during daytime as well as during nighttime adds a considerable complication to the maintenance planning process. The rolling stock circulation of NS is complicated, since rolling stock units usually do not follow a recurring pattern every day but spread out over the entire network instead. When all maintenance activities are performed during nighttime, the maintenance scheduling process can at least to some extent be held under control, since in general for each rolling stock unit there exists a possibility to maintain it during the night and hence each rolling stock unit is automatically 'seen' every 24 hours. The case for daytime maintenance, however, is different, since rolling stock units do not have a long planned standstill every day during daytime and as such it cannot be guaranteed that rolling stock units can be maintained during daytime always. An optimal maintenance schedule increasing daytime and nighttime maintenance is therefore not straightforward to make and difficult to create manually. Therefore, the possibility of daytime maintenance raises the desire for an automated maintenance scheduling tool.

Second, it is not desirable to open all potential maintenance locations during daytime, since the capacity use of many of these locations during daytime would often be too low. The choice on which location or set of locations to open for daytime maintenance, however, is not straightforward. The optimal locations to open for daytime maintenance depends on the optimal schedule that can be created with these locations opened. Similarly, the optimal maintenance schedule depends on the locations that are opened for daytime maintenance. Hence, the location choice and the optimal maintenance schedule are interrelated.

The problem NS is facing is to achieve a decrease in the pressure on capacity of maintenance locations during nighttime, by performing more activities during daytime. To this end, a method is required to automatically determine an optimal maintenance schedule and find an optimal maintenance location choice.

## 1.8 Objectives of the current research

The current section outlines the objective of the current research. It starts by describing how the problems identified in Section 1.7 relate to the objectives of railway operators, then it concretizes the criteria by which the quality of solutions can be assesed, after which the identified problems are translated into concrete objectives for the current research.

#### Railway operators' objectives

The previous section has indicated that the problem at hand in the current research regards maintenance processes of railway operators. First, it has been indicated that the maintenance scheduling process, which is usually performed manually, is becoming increasingly complex. Second, the maintenance location choice is not straightforward and interrelated with the maintenance scheduling process. In particular, these two problems prevail for NS, that is currently considering to perform more daytime maintenance in order to reduce the pressure on capacity of maintenance locations during the night. However, performing more daytime maintenance further complicates the planning process. Furthermore, the possibility of daytime maintenance raises the question which maintenance locations should be opened for daytime maintenance.

The objective of the current research is closely related to the goal of railway operators in general. This goal is to comply with formal and informal agreements with stakeholders to guarantee business continuity. It results in three concrete business objectives: profitability, meaning that railway operators strive for more revenues compared to the expenditures, performance, signifying that railway operators intend to deliver an attractive level and quality of service to its customers, and compliance, reflecting the fact that railway operators need to abide by certain rules and regulations (Holtzer, 2020).

The identified problems directly touch upon the business objectives of railway operators, and of NS in particular. In the first place, they impact profitability. Note that the maintenance scheduling process is usually performed manually. Due to its increasing complexity, more planning personnel is required, leading to higher costs. Especially the introduction of daytime maintenance would add considerably to the complexity of the scheduling process and hence to the required planning personnel and costs. Moreover, the contents of the schedule itself also differ in costs: for example, maintenance activities that are performed during daytime are in general less costly than maintenance activities performed during nighttime. Also, the locations opened for maintenance impact profitability. Infrastructural costs apply when a location is used for maintenance, and also it is important to open a location only if a considerable amount of work can be performed on it, to avoid the situation of stationing maintenance staff on maintenance locations while there is not enough work available to keep all maintenance staff deployed.

In the second place, the identified problems impact the performance and compliance. Maintenance schedules assure that maintenance is performed on time. Due to the increasing complexity, it becomes increasingly hard for manual planners to find a schedule such that each rolling stock unit is maintained on time. As a result, the performance may decrease, since rolling stock units may not be cleaned on time or rolling stock units may not have undergone technical checks due to which they cannot be used for train services, leading to a lower level of service. Moreover, the compliance to rules and regulations may decrease, since it becomes harder to find a maintenance schedule in such a way that all rolling stock units are maintained on time. This problem is becoming even more relevant when considering that due to disruptions the maintenance activities may not be performed according to the maintenance schedule. This leads to the fact that even if the initial maintenance schedule would lead to compliance and good level of service, in operations this may not be the case anymore if no rescheduling opportunities are valid.

Figure 1.7 gives a schematic overview of how various factors impact the achievement of the overall goal of railway operations, used by NS. The brown-coloured fields indicate how maintenance influences these goals. It shows that the number of available maintenance locations, the number of maintenance staff and the number of maintenance activities affect the expenditures for rolling stock maintenance, personnel and infrastructure costs. Moreover it demonstrates that the quality of maintenance impacts number of passengers (and hence the revenues generated by these passengers) due to its relation with passenger experience.

#### Criteria

The current research aims to address both the issue of finding a maintenance schedule and the issue of choosing maintenance locations. It has been explained above that the prevailing problems negatively impact the overall objective of railway operators.

The goal of the present research is to find a method to schedule maintenance activities and to find a location choice. Observe that these two are interrelated: the schedule depends on the available maintenance locations and the maintenance location choice depends on the schedule of maintenance activities. Hence, the two aspects need to be addressed simultaneously.

Recall that the objective of NS is to reduce the pressure on capacity of maintenance loca-

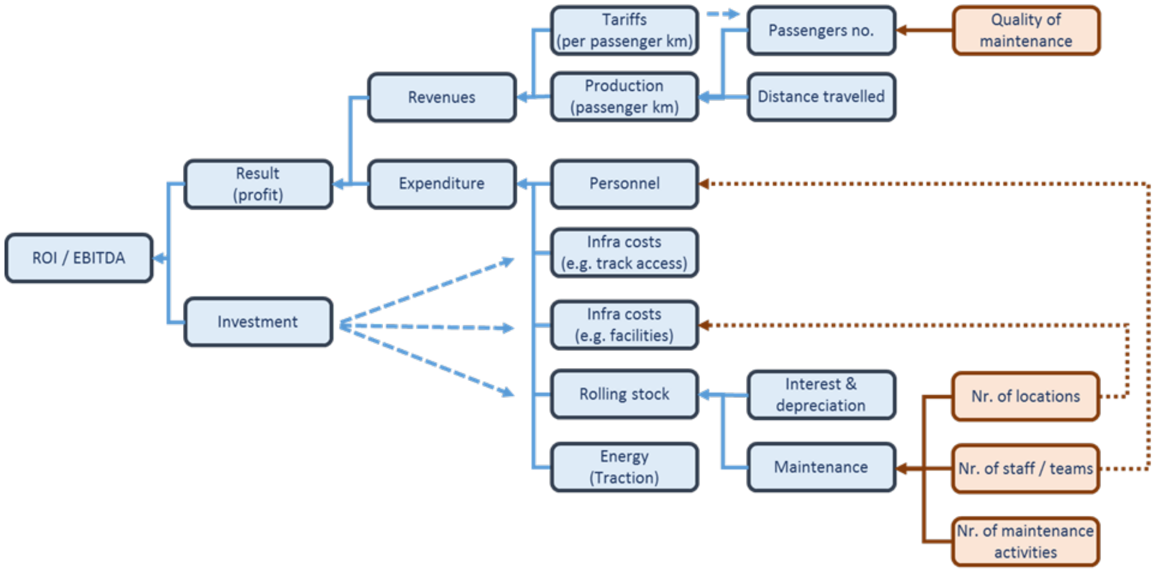



*Figure 1.7: Schematic overview of railway operators' objectives used by NS. The main objective is expressed in terms of costs, reflected by the Return on Investment (ROI) or Earnings Before Interest, Taxes, Depreciation and Amortization (EBITDA) (NS, 2020).*

tions during nighttime. Hence, a maintenance schedule and a location choice are considered to be optimal if they minimize the number of nighttime maintenance activities.

As indicated, the increasing complexity of the maintenance scheduling process leads to more planning personnel required, leading to higher costs. Hence, it is required to find the maintenance schedule in an automated way to reduce these costs. This automated method can then be used as a planning support tool to support human planners. To enable easy use by planners, the method should be efficient, meaning that a solution needs to be found within several hours. If, in addition, the method is to be used during operations to obtain an updated schedule, the requirements for efficiency are even higher: then, the method should be able to find solutions preferably within seconds, and at most within minutes.

### Research questions

The goal of the current research can be summarized by the following main research question:

*Given the rolling stock circulation, how to find efficiently*
*an optimal rolling stock maintenance schedule and simultaneously optimize*
*the choice regarding which locations are opened during daytime and during nighttime?*

To this end, the research starts by investigating relevant literature to determine how the present problem and related problems have been addressed before, and what aspects have not been addressed yet, identifying the research gap. Then, as far as no other methods are available in the currently existing literature, new models are devised that solve the problem at hand. Then, results are generated using the devised models to assess whether they meet the desired criteria. This methodology can be summarized in the following three research questions.

1. What literature regarding rolling stock maintenance scheduling and rolling stock maintenance location choice is available?

2. How can rolling stock maintenance scheduling and rolling stock maintenance location choice be efficiently modeled simultaneously?

3. What results can be obtained by applying the model to various rolling stock circulations?

In the development of the planning methods devised in the current research, their interaction with the existing planning environment Viriato, developed by *SMA und Partner A.G.* (headquartered in Zürich, Switzerland) is addressed. Especially its *Rostering Interface* is under consideration, since this enables to use the scheduling capabilities of Viriato in conjunction with privately developed external models.

The use of Viriato is benefical since it is an acknowledged product in the industry, and the potential compatibility of the current research with Viriato would support its practical applicability. Moreover, the involvement of SMA, the developer of Viriato, in the current research allows for specific feedback and a critical second opinion in the development of the research. In addition, TU Delft is interested in the use of Viriato in the research in the railway field and requested to investigate its applicability in research projects like these. To this end, also the following, fourth research question is addressed in the current research.

4. Is the Rostering Interface to the planning software Viriato an effective and efficient tool to model this kind of problems?

## 1.9 Outline

The current research is structured as follows.

Chapter 2 gives an overview of the relevant existing literature, answering the first research questions. It addresses maintenance scheduling and maintenance location choice and positions the research in the scientific field and identifies interesting research gap.

Chapters 3-6 are dedicated to the development of models, confronting the second research question. Chapter 3 defines the core problem of the current research, the *Maintenance Scheduling and Location Choice Problem* (**MSLCP**), and Chapter 4 develops a model to solve it. Given a rolling stock circulation, the goal of the **MSLCP** is to find an optimal maintenance schedule and an optimal maintenance location choice. Chapter 5 formulates the *Activity Planning Problem*. It recognizes that the **MSLCP** does not aim to provide a measure for the capacity of maintenance locations nor a feasible planning of maintenance teams at maintenance locations, and it addresses this issue by defining the **APP** model that, given a solution of the **MSLCP**, is able to provide both. Chapter 6 formulates the *Capacitated Maintenance Scheduling and Location Problem* (**CMSLCP**). It aims to provide a solution to the **MSLCP** that takes into account the capacity of maintenance locations. To this end, it designs a framework that integrates the **MSLCP** and **APP**.

Chapters 7-9 provide results for the developed models, addressing the third research question. Chapter 7 thoroughly investigates the behavior of the **MSLCP** by testing it on many different scenarios. Chapter 8 concretises the **CMSLCP** (and implicitly the **APP**, which is contained in the **CMSLCP** framework) by showing its correct functioning for a demonstrative example and by investigating the effect of its most important sub process on its performance. Chapter 9 takes a different perspective and applies the current research in an actual, practical situation.

A discussion of the current research is found in Chapter 10, critically assessing the current research in a broad perspective and debating its limitations. It also provides a discussion of the usefulness of Viriato in research and in practice, answering the fourth research question. Chapter 11 presents the main conclusions and gives recommendations for future research.

# Chapter 2

# Literature review

The current work considers rolling stock maintenance scheduling as well as rolling stock maintenance location choice. This chapter aims to identify the contributions of the current work to the literature and to obtain insights in the methodologies and techniques used in comparable research.

Section 2.1 first discusses relevant scientific literature on rolling stock maintenance scheduling. This literature usually focuses on the allocation of rolling stock units to train trips (also called the rolling stock circulation), under the assumption of a given timetable. In general, maintenance is included in these papers as additional constraints. Then, in Section 2.2, some papers on rolling stock maintenance location choice will be discussed. Section 2.3 will indicate the contribution of the current research.

At the end of both Sections 2.1 as 2.2 some corresponding literature from the field of aviation will be discussed. Much of the research on maintenance scheduling in the aviation industry is relevant, since it exhibits systematic similarities with rolling stock maintenance scheduling. Both cases entail a network with a limited number of nodes (airports and stations) on which mobile units operate (aircraft and rolling stock), governed by a given timetable. In general, these mobile units need to be maintained in regular intervals at immobile maintenance locations. It is therefore worthwhile to consider literature in the field of aviation as well.

## 2.1 Maintenance scheduling

Herr et al. (2017) consider a problem in which rolling stock units need to be assigned to train trips such that maintenance constraints are satisfied. They assume that with each trip, the rolling stock quality degrades with a degradation rate proportional to the trip duration. They propose a MIP model and the objective that they use is to schedule maintenance as late as possible, thereby making optimal use of the total allowable interval between maintenance activities. They effectuate this by a max-min objective function, maximizing the minimum degradation of rolling stock units just before the application of maintenance (i.e. the degradation of the rolling stock unit in the planning when maintenance was least necessary). The problem is non-linear and in spite of the application of linearization techniques, the problem could only be solved for small instances.

Just as Herr et al. (2017), Andrés et al. (2015) consider the problem of assigning rolling stock units to train trips. They use an aggregated space-time network in which the nodes are trip arrival times or trip departure times with the corresponding location. A MIP model that minimizes total operating costs is designed. The MIP model is computationally expensive to solve, and hence a column generation approach is used to solve the problem in reasonable time.

Maróti and Kroon (2007) consider a problem regarding heavy maintenance. They acknowledge that, in order to route a rolling stock unit to a maintenance location, it is often necessary to adjust the rolling stock circulation. They propose a model to make modifications to the regular plan to route rolling stock units to maintenance locations. In doing so, one rolling stock unit takes over the role of another rolling stock unit and vice versa so that a rolling stock unit can be routed to a maintenance location. The model is proposed for a one- to three-day time horizon. They propose an integer programming formulation which is successful if one rolling stock unit needs to be rerouted. However, for situations when multiple rolling stock units need to be rerouted, a heuristic approach is proposed that iteratively applies the integer programming formulation.

Wagenaar and Kroon (2015) consider the problem in which rolling stock needs to be rescheduled after disruptions. Most existing models cannot be used for this problem since these models are only applicable to the planning phase. In the operational phase, rolling stock units are may be scheduled for maintenance. Rescheduling the rolling stock circulation breaks the maintenance planning and this is undesirable. Wagenaar and Kroon (2015) come up with a model that reschedules rolling stock circulation taking into account the current maintenance planning. They base their models on the composition model, which assigns rolling stock units to train trips. They come up with three models that have comparable performance, dependent on the problem size.

Another relevant work is the research by Van Hövell (2019), focusing on the situation in The Netherlands. Spare rolling stock is available at daytime, which can be maintained. However, Van Hövell provides a model to make adjustments to the rolling stock schedule in order to carry out more maintenance at daytime. This leads to a decreased workload during nighttime. For a small case study, considering the rolling stock operating on one trajectory, it is shown that the proposed model obtains a solution in which all rolling stock units can be maintained at daytime instead of at nighttime.

**Corresponding literature in aviation**

In the area of aircraft maintenance scheduling, Clarke et al. (1997) introduced the so-called aircraft rotation problem, which aims to assign specific aircraft to each flight from a given set of flights. This highly resembles the problem in the railway area that tries to assign specific rolling stock units to each train trip from a given set of train trips. Clarke et al. (1997) use a network formulation where the nodes are the airports and the arcs are the flights. A path in this graph is a sequence of trips for one aircraft. Since the timetable is recurring perpetually, Clarke et al. (1997) aim to find a cycle, such that the sequence of trips is recurrent. They aim to find this cycle optimally and such that maintenance constraints are satisfied. The maintenance constraints add heavily to the computational complexity. Therefore, the model is solved using Lagrangian Relaxation, thereby finding a feasible but not necessarily optimal solution in reasonable computation time.

Gopalan and Talluri (1998) build upon the work of Clarke et al. (1997) and use a similar objective and similar constraints. However, the method they apply is different, since Gopalan and Talluri (1998) propose algorithms to find a solution instead of a MIP model. They adopt two different models: the first is static with an infinite horizon, meaning that the flight schedule is the same for each day and repetitive; the second is dynamic with a finite horizon, meaning that a flight schedule for some given number of days is given. They use a three-step procedure: first they find a feasible aircraft assignment without maintenance constraints, then they make adjustments so that the maintenance requirements can be met, and lastly they find a maintenance rotation satisfying maintenance constraints using polynomial-time algorithms that they propose.

Sarac et al. (2006) acknowledge that long-term aircraft routing plans are often ignored in operations, due to the high number of disruptions that take place. Hence they develop a model that solves the aircraft maintenance scheduling problem including maintenance constraints in an operational context. The planning horizon is one day and not more, since this would be too optimistic to be applicable in an industry with as many disruptions as the aviation industry. Sarac et al. (2006) propose a connection network where the nodes represent flight legs and the arcs represent feasible connections among the flights. They show the problem is NP-hard and hence apply column generation to achieve feasible solutions in reasonable time.

## 2.2 Maintenance location choice

Tönissen et al. (2019) acknowledge that the maintenance routing problem is inseparably connected to the problem that considers the positions in the network where to open maintenance locations. Therefore they aim at locating the maintenance facilities in the railway network. However, since the maintenance location choice is dependent on line and fleet planning, they come up with models that determine optimal maintenance locations under line and fleet planning that is subject to uncertainty or change. They propose two models: one to optimize the average-case, intended for risk-neutral decision makers, and one to optimize the worst-case scenario, intended for risk-averse decision makers. The models are two-stage stochastic mixed integer programming models, in which the first stage is to open a facility, and in the second stage to minimize the routing cost for the first-stage location decision for each line plan scenario. Their model is an aggregate model on an annual basis ignoring the details of the day-to-day rolling stock planning.

Tönissen and Arts (2018) build on Tönissen et al. (2019). Their work is similar since both works consider the choice for rolling stock maintenance locations on a longer time scale. Compared to Tönissen et al. (2019), this work includes the recovery costs of maintenance location decisions, unplanned maintenance, multiple facility sizes and economies of scale (providing that a location twice as big is not twice as expensive). However, as a downside, the second-stage problem that could also be observed in Tönissen et al. (2019) becomes, as a result, NP-hard. Therefore they provide an algorithm with the aim to avoid having to solve the second stage for every scenario.

Canca and Barrena (2018) consider the simultaneous rolling stock allocation to lines and choice for depot locations in a rail-rapid transit context. They propose a MIP formulation which appears hard to solve. Therefore they propose a three-step heuristic approach with an algorithm to determine the minimum number of vehicles needed for each line, a MIP formulation to determine the routes of rolling stock on each line, and a Genetic Algorithm to determine the circulation of rolling stock on lines over multiple days together with the depot choice. Although the interaction between depot choice and rolling stock allocation is relevant to the current research, there are also fundamental differences. Canca and Barrena (2018) assume that rolling stock units cannot be exchanged between lines on the same day, but only during the night at depot locations. This is an important difference compared to the regular railway context. Moreover, they do not explicitly model maintenance.

Zomer (2019) considers the railway network in The Netherlands. Traditionally, most maintenance in The Netherlands is performed at nighttime. However, as a result of the growing capacity issues, also daytime maintenance is considered by the country's largest railway operator NS. This raises the question which locations need to be opened for daytime maintenance.

Zomer (2019) took historical data as a starting point and designed a simulation model. This simulation model was used to estimate the expected effects of carrying out daytime maintenance at given maintenance locations. Although the use of historical rolling stock data has advantages, since it accurately describes reality (as opposed to rolling stock planning data), it cannot be used for situations in the future for which only planning data is available. Also, although simulation offers the opportunity to investigate effects for various scenarios of daytime maintenance locations, it cannot be used to systematically optimize the maintenance location choice.

**Corresponding literature in aviation**

Feo and Bard (1989) introduce the problem of assigning aircraft to given flights and simultaneously optimizing the number of maintenance facilities. They use an infinite horizon and a simplified timetable structure. They solve the problem as a minimum cost multi-commodity flow problem. The proposed MIP formulation is difficult to solve, so they come up with a two-phase heuristic approach. In the first phase, many possible trip patterns for individual aircraft are computed. Then in the second phase, the most promising trip patterns are combined in such a way that it solves the global problem. Their heuristic yields good results in a reasonable amount of computation time.

Gopalan (2014) is closely related to the work of Feo and Bard (1989), although there are differences in the details of the addressed problem and solution approach. Gopalan (2014) assumes routes during the day are given for each aircraft, although these routes are not yet assigned to specific aircraft. Each route needs to be connected to a route on the next day in such a way that the routing passes through a maintenance location with some given periodicity. The objective is to minimize the number of maintenance locations (one of the differences from Feo and Bard (1989), that considers cost minimization). To this end, they first provide MIP formulation and solve the dual to obtain a lower bound of the objective value. Then, they propose four heuristics in a similar way as Feo  Bard: they first select an arbitrary connections of routes and and obtain the minimum number of facilities for this arbitrary routing to obtain a feasible solution. This is done multiple times to obtain increasingly better solutions.

## 2.3 Contribution to the literature

In Table 2.1, the discussed literature is classified in several categories. It shows for each paper whether it is written in the aviation (A) or in the railway (R) context, whether it considers the allocation of mobile units (MU, i.e. rolling stock units or aircraft) to trips, whether it considers maintenance constraints, whether it creates an explicit maintenance schedule for every (relevant) MU and whether it considers facility location choice optimization.

Some more explanation may be necessary on the column indicating whether an explicit maintenance schedule for every (relevant) MU is created. A paper that considers maintenance does not necessarily create an explicit maintenance schedule. An example is the work by Clarke et al. (1997). They do consider maintenance by requiring that each trip path may not exceed some specified length, however they do not explicitly determine at what moment in time and at what location maintenance is carried out.

| | A/R | MU alloc. | Maint. considered | Explicit maint. sched. | Location choice |
|---|---|---|---|---|---|
| Herr et al. (2017) | R | x | x | x | |
| Andrés et al. (2015) | R | x | x | x | |
| Maróti and Kroon (2007) | R | x | x | x | |
| Wagenaar and Kroon (2015) | R | x | x | x | |
| Clarke et al. (1997) | A | x | x | | |
| Gopalan and Talluri (1998) | A | x | x | | |
| Sarac et al. (2006) | A | x | x | | |
| Tönissen et al. (2019) | R | | x | | x |
| Tönissen and Arts (2018) | R | | x | | |
| Canca and Barrena (2018) | R | x | | | x |
| Feo and Bard (1989) | A | x | x | | x |
| Gopalan (2014) | A | | x | | x |
| Van Hövell (2019) | R | | x | x | |
| Zomer (2019) | R | | x | | x |
| *Current* | R | | x | x | x |

*Table 2.1: Overview of the literature discussed in Chapter 2.*

The current research is a contribution to the scientific literature since in the following ways.

1. It offers a simultaneous optimization of rolling stock maintenance scheduling and maintenance location choice. The current paper is unique since it is, to the author's best knowledge, the only paper that can be classified in both of the last two columns of Table 2.1.

2. It explicitly distinguishes between daytime maintenance and nighttime maintenance. This is relevant for at least the situation in The Netherlands, where nighttime maintenance is standard and recent developments have led the company to investigate daytime maintenance as well.

3. It considers maintenance location choice for The Netherlands, comparable to the work of Zomer (2019), but approaches the problem from an optimization perspective instead of from a simulation perspective.

# Chapter 3

# Problem description

The problem that is addressed in the current research is defined as the *Maintenance Scheduling and Location Choice Problem* (**MSLCP**). The most important input of the **MSLCP** is a rolling stock circulation containing all planned rolling stock movements of a given set of rolling stock units. In addition, a set of potential maintenance locations where maintenance activities can be carried out, and a set of maintenance activities that need to be scheduled for all rolling stock units, with a given duration of each maintenance activity and given intervals between consecutive maintenance activities, needs to be provided. The goal of the **MSLCP** is to simultaneously find an optimal maintenance schedule and an optimal choice of locations used for maintenance (from the set of potential maintenance locations).

The input and output of the **MSLCP** is graphically presented in Figure 3.1.


*Figure 3.1: The characterizing input and output of the **MSLCP**: using a rolling stock circulation, an optimal maintenance location choice and an optimal maintenance schedule is to be determined. It must be noted that in addition, also the set of maintenance locations and the set of maintenance types needs to be provided as an input, but these have more the nature of a user setting and are therefore not provided in the current figure.*

Section 3.1 gives a detailed overview of the most important problem characteristics of the **MSLCP** and Section 3.2 summarizes the most important assumptions for this problem and

## 3.1 Problem characteristics of the MSLCP

**Rolling stock circulation** It is assumed that a rolling stock circulation is given. As indicated in Chapter 1, the rolling stock circulation contains a list of trips that a rolling stock unit is scheduled to perform, for each rolling stock unit. These trips include an origin and destination position (which are often train stations), and the corresponding planned departure and arrival times. An example of a rolling stock circulation is found in Figure 3.2. Squares represent arrivals or departures of rolling stock. Inside the squares, the station abbreviation and the departure or arrival time at this station is given. Solid lines represent time intervals where a rolling stock unit is used for a train service. For example, from 07.09 to 10.41, the depicted rolling stock unit is planned to be used for a train service between Ekz (Enkhuizen) and Hrl (Heerlen). Dashed lines represent time intervals where a rolling stock unit is not in service.

For example, between 10.41 and 16.19, the depicted rolling stock unit is not in service and will be standing still at Hrl.

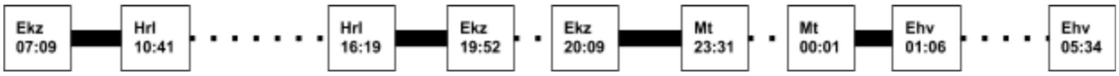



*Figure 3.2: Example of a rolling stock circulation*

**Maintenance opportunities** It is assumed that maintenance can be carried out if and only if a rolling stock unit stands still. These moments are referred to as *maintenance opportuntities* (MOs). Table 3.1 indicates the MOs corresponding to the rolling stock circulation from Figure 3.2. All MOs (even short ones) can potentially be used for maintenance activities. When a maintenance activity is assigned to an MO, this implies that the maintenance activity needs to be performed between the start and end time of this MO. The exact scheduled time of the maintenance activity is not determined in the scope of this research.

| MO | day | location | start time | end time |
|---|---|---|---|---|
| 1 | 1 | Hrl | 10:41 | 16:19 |
| 2 | 1 | Ekz | 19:52 | 20:09 |
| 3 | 1 | Mt | 23:31 | 0:01 |
| 4 | 2 | Ehv | 01:06 | 05:34 |

*Table 3.1: Example of the MOs corresponding to the rolling stock circulation example from Figure 3.2.*

Two time windows are considered: daytime and nighttime. An MO can be during daytime or nighttime. This division is especially relevant for NS, since it is currently considering a transition from performing maintenance during nighttime only to performing maintenance during both nighttime and daytime (Zomer, 2019; Van Hövell, 2019). The time window for daytime maintenance is set at 07.00-19.00, the time window for nighttime maintenance from 19.00-07.00. Some MOs may be partly during daytime and partly during nighttime (this occurs, for example, when an MO lasts from 18.00 to 20.00). For these MOs it is not straightforward whether maintenance would be carried out during the day or during the night (i.e. whether the MO should be considered to be during daytime or during nighttime). For these reasons, the following simplification is applied in the current problem: an MO is marked to be *during daytime* if and only if both its start time and its end time are between 07.00 and 19.00 of the same day; an MO is marked to be *during nighttime* in all other cases. Note that, although this assumption is reasonable in most cases, there are some occasions where it is not realistic: for example, an MO starting at 11.00 (during daytime) and ending at 19.01 (just after the stat of the nighttime time window) would be classified to be a nighttime MO whereas maintenance scheduled in it can probably be performed during daytime.

**Maintenance activities** It is assumed that a set of regular maintenance types of arbitrary size is known, and that for each maintenance type a fixed duration and a fixed maximum time interval between subsequent maintenance activities is given. Heavy maintenance is not included in the scope of the research: this type of activities is usually not planned in the rolling stock circulation, but accounted for separately by dispatchers whenever a heavy maintenance activity needs to be performed in the near future.

Maintenance activities are assigned to maintenance opportunities, which is referred to as the *maintenance scheduling*. It is assumed that maintenance activities are carried out in a subsequent manner and cannot overlap. Moreover, it is assumed that the number of hours at the start of the planning horizon since the last maintenance activity for each rolling stock unit is given.

**Maintenance locations** Maintenance can be carried out at a potential maintenance location. The set of potential maintenance locations is given. For each of these locations, it can be decided whether a location should be opened or not. This decision is referred to as *location choice*. Each location can be opened during nighttime or during daytime, that is, for each maintenance location, there are four possible outcomes: the location is not used at all, the location is used for daytime maintenance, the location is used for nighttime maintenance, or the location is used for both.

Observe that it is assumed that all maintenance activities can and will be performed at a potential maintenance location, and that there is no subdivision in the types of maintenance that can be performed at specific locations. This is reasonable for the Dutch case, since all regular maintenance activities considered in the current research need to be performed at maintenance locations. For other countries, however, this may not be suitable since some maintenance activities need to be performed at specific locations (see also Section 1.6.

The reachability of maintenance locations is incorporated implicitly since the current research takes the given rolling stock circulation as an input. Hence, the movements to and from maintenance locations need to be taken into account in the rolling stock circulation. Maintenance activities can take place only at those locations where rolling stock units are located according to the rolling stock circulation. This allows to discard the reachability of maintenance locations in the current research.

**Planning horizon** The planning horizon in the current research is equal to the planning horizon in the rolling stock circulation. In other words, a maintenance schedule is determined for the entire time horizon of the rolling stock circulation, and as a result, the optimal location choice is valid for the length of this time horizon as well. In The Netherlands, rolling stock circulations are available for periods of eight weeks. This implies that the planning horizon in the current research is also fixed at eight weeks.

**Objective** The objective of the **MSLCP** is to assign maintenance activities to MOs and determine for each location (1) if it is open during daytime and (2) whether it is open during nighttime, satisfying the intervals between maintenance activities and the other constraints, in such a way that the number of nighttime maintenance activities is minimized. This goal may be relevant to NS, since the capacity of maintenance locations at nighttime is under pressure. As a technical aside, a small penalty needs to apply for any maintenance activity, to avoid the situation that more daytime maintenance activities are planned than necessary.

## 3.2 List of assumptions

To summarize, all assumptions of the **MSLCP** are given in Table 3.3. For each assumption, it is indicated whether this assumption reflects a **model** choice, meaning it is fixed and cannot be altered without changing the model, or whether the assumption concerns expected **input**.

| | | assumption | model | input |
|---|---|---|---|---|
| **RS circulation** | 1. | The rolling stock circulation is given. | | x |
| **Maintenance locations** | 2. | The set of nighttime maintenance locations is given. | | x |
| | 3. | All nighttime locations are open for maintenance during nighttime. | x | |
| | 4. | The set of potential daytime maintenance locations is given. | | x |
| | 5. | The maximum number of daytime locations that can be opened is given. | | x |
| **Maintenance types** | 6. | The number of maintenance types are given. For each maintenance type, the required interval between consecutive maintenance activities of this type and the duration of maintenance activities of this type are given. | | x |
| | 7. | The required interval between consecutive maintenance activities of the same type is measured in time (as opposed to, for instance, the total distance covered). | x | |
| | 8. | The specifications of maintenance types are identical for all rolling stock units and for all maintenance locations. | x | |
| | 9. | The initial conditions for each rolling stock unit, for each maintenance type are given (measured as the total number of hours since the last maintenance activity of a specific maintenance type at the start of the time horizon). | | x |
| **Scheduling** | 10. | Maintenance activities are assigned to MOs. An MO occurs whenever a rolling stock unit has a planned standstill. | x | |
| | 11. | Any MO is classified to be during daytime if its end time is during daytime. Else, it is classified to be during nighttime. | x | |
| | 12. | The time between two consecutive maintenance activities of the same type may not exceed the specified maximum interval between consecutive maintenance activities. This time interval is measured from the end of the MO in which the first maintenance activity is performed to the start of the MO in which the second, subsequent maintenance activity is performed. | x | |
| | 13. | The sum of the duration of all maintenance activities performed in an MO may not exceed the total time available in that MO. | x | |
| | 14. | The number of activities that can be performed at a maintenance location is not restricted (i.e. maintenance locations have infinite capacity) | x | |
| **Time windows** | 15. | The hours of each day are divided into two time windows: daytime and nighttime | x | |
| | 16. | The start of the daytime time window and the start of the nighttime time window are given. | | x |

*Table 3.3: Assumptions of the **MSLCP** model.*

# Chapter 4

# Base model development

The current chapter develops models for the **MSLCP** and for some related problems. Section 4.1 introduces all necessary concepts in mathematical notation. In Section 4.2 the **MSLCP** is mathematically formulated. Section 4.3 defines two other models that can be considered to be extensions of the **MSLCP**. For reference, Table 4.4 gives an overview of all mathematical notation used in the **MSLCP** model with its meaning.

## 4.1 Mathematical notation for MSLCP

Let $I$ be the set of rolling stock units considered in the current problem and let $i \in I$ be the index used to indicate a specific rolling stock unit. Let $T \in \mathbb{R}$ be the length of the planning horizon in hours. Let $L$ denote the set of potential maintenance locations.

#### Maintenance opportunities

From the rolling stock circulation, *maintenance opportunities* (MOs) can be identified. An MO occurs when a rolling stock unit is standing still at a potential maintenance location. Let $J_i \equiv \{1, ..., \overline{J_i}\}$ denote the the set of MOs for rolling stock unit $i \in I$. The location of a rolling stock unit $i$ at MO $j \in J_i$ is denoted by $l_{ij} \in L$. The start time of MO $j \in J_i$ is denoted by $s_{ij} \in \mathbb{R}$ and the end time is denoted by $e_{ij} \in \mathbb{R}$, where time is given as the number of hours that passed since midnight of the first day in the planning horizon, unless stated otherwise.

Let $d_{ij}$ indicate whether an MO is classified to be during daytime or during nighttime: let $d_{ij} = 1$ if MO $j \in J_i$ for rolling stock unit $i$ is during daytime and let $d_{ij} = 0$ if a MO $j \in J_i$ for rolling stock unit $i \in I$ is nighttime. Let $\delta_d$ be the hour of the day when the daytime maintenance window starts and let $\delta_n$ be the hour of the day when the nighttime maintenance window starts. Recall that an MO is classified to be during daytime if and only if both its start and end time are during daytime of the same day. In mathematical formulation,

$$d_{ij} = \begin{cases} 1 & \text{if } \delta^D \leq e_{ij} \bmod 24 < \delta^N \\ 0 & \text{else} \end{cases} \quad .$$

The MOs for the rolling stock unit from the example of Figure 3.2 and Table 3.1 is displayed in mathematical notation in Table 4.1. Observe that the start and end time of the MOs are indicated into hours after midnight of the first day for computational convenience.

| $i$ | $j$ | $l_{ij}$ | $s_{ij}$ | $e_{ij}$ | $d_{ij}$ |
|---|---|---|---|---|---|
| 1 | 1 | Hrl | 10.68 | 16.32 | 1 |
| 1 | 2 | Ekz | 19.87 | 20.15 | 0 |
| 1 | 3 | Mt | 23.52 | 24.02 | 0 |
| 1 | 4 | Ehv | 25.10 | 29.57 | 0 |

*Table 4.1: Example of the MOs for rolling stock unit $i = 1$ derived from the rolling stock movements given in Table 3.1.*

### Maintenance types

Let $K$ be the set of maintenance types, $K \equiv \{1, ..., \bar{K}\}$. For each maintenance type $k \in K$, let $v_k \in \mathbb{R}^+$ be its duration in hours and let $o_k \in \mathbb{R}^+$ be the maximum interval between two consecutive maintenance activities of maintenance type $k$ in hours. Table 4.2 gives maintenance types that apply at NS in mathematical formulation.

| Maintenance type | $k$ | $v_k$ | $o_k$ |
|---|---|---|---|
| Technical A-check | 1 | 1.0 | 288 |
| Technical B-check | 2 | 0.17 | 48 |
| Interior cleaning | 3 | 1.0 | 24 |
| Exterior cleaning | 4 | 0.5 | 336 |

*Table 4.2: Maintenance types in The Netherlands given in Table 1.3, in mathematical notation, with maintenance duration $v_k$ and maintenance interval $o_k$ in hours.*

### Maintenance locations

A potential maintenance location can be opened during daytime, meaning it is available for maintenance from $\delta^D$ to $\delta^N$, by default from 07.00 to 19.00. Let $y_l^D \in \{0, 1\}$ be a binary variable equal to 1 if location $l \in L$ is available for daytime maintenance and 0 otherwise. Let $y_l^N \in \{0, 1\}$ be a binary variable equal to 1 if location $l \in Ł$ is available for nighttime maintenance and 0 otherwise.

The number of potential maintenance locations that can be opened is usually restricted. This is for example the case at NS, since it considers a gradual shift to rolling stock maintenance during daytime and does not want to open all locations at once. The current research addresses the case where the number of daytime maintenance locations is restricted and assumes that the number of nighttime maintenance locations is unconstrained. To this end, let $L_{max}^D$ denote the maximum number of potential maintenance locations that can be opened during daytime.

### Maintenance schedule

Maintenance activities are assigned to maintenance opportunities. Let $x_{ijk} \in \{0, 1\}$ be a binary variable equal to 1 if maintenance of type $k$ is performed to rolling stock unit $i \in I$ at MO $j \in J_i$, and 0 otherwise. It is required that the total time available at MO $j$ is not exceeded: $\sum_{k \in K} x_{ijk} v_k \leq e_{ij} - s_{ij}$.

Furthermore, an MO $j$ can only be used if the corresponding location is open at the moment of the MO. Therefore:

- $(d_{ij} = 0, y_{l_{ij}}^N = 0) \implies x_{ijk} = 0 \quad \forall k \in K$

- $(d_{ij} = 1, y^D_{l_{ij}} = 0) \implies x_{ijk} = 0 \quad \forall k \in K$

Moreover, the intervals between successive maintenance activities of the same type should satisfy the given criteria. The interval between two MOs $j, j' \in J_i$, $j \neq j'$, is measured from the end time of the first MO to the start time of another MO: $s_{ij'} - e_{ij}$. If activity $k \in K$ is scheduled for rolling stock unit $i \in I$ in MO $j$, then the next maintenance activity should be scheduled such that the interval constraints are satisfied. Let $V_{ijk} \subset J_i$ denote the set of maintenance opportunities for rolling stock unit $i \in I$ that start after the end of MO $j \in J_i$ but earlier than $o_k$ hours after the end of MO $j \in J_i$. This set is $V_{ijk} = \{p \in J_i : e_{ij} < s_{ip} \leq e_{ij} + o_k\}$ for $j \in J_i$. It is then required that for all $i \in I, j \in J_i$, the following implication holds:

$$x_{ijk} = 1 \implies \exists j' \in V_{ijk} : x_{ij'k} = 1$$

.

Observe that a next maintenance activity only needs to be scheduled if maintenance needs to be carried out within the current planning horizon, that is, if $e_{ij} + o_k \leq T$.

**Initial conditions**

Let $b_{ik}$ be the number of hours since the last maintenance activity of type $k$ for rolling stock unit $i$ at midnight of the first day. Then let $V_{i0k} = \{p \in J_i : s_{ip} \leq o_k + b_{ik}\}$.

**Planning horizon**

Firstly, the start of the planning horizon is chosen in such a way that 0 is the midnight of the first day. Observe that inevitably, some MOs are 'invisible', since there may exist some MO $j' \notin J_i$ with $s_{ij'} < 0$ and $e_{ij'} > 0$. The set $J_i$ contains only MOs that start *after* midnight of the first day. Secondly, the end of the planning horizon ($T$) is arbitrary, but should be chosen not later than the end of the time period for which the rolling stock circulation (which is input to the **MSLCP**) is defined.

## 4.2 MSLCP model

The **MSLCP** model aims to find $x_{ijk}$ and $y_l$ satisfying the above described constraints that minimize number of maintenance activities during the day. In this model, the decision variables are $x_{ijk}$ $(i \in I, j \in J_i, k \in K)$ and $y^D_l$ $(l \in L)$. $y^N_L$ are considered to be given in the input.

The objective is to minimize the number of maintenance activities during the night. The model can then be formulated as follows.

$$\min \sum_{i \in I} \sum_{j \in J_i} \sum_{k \in K} x_{ijk}(1 - d_{ij}) + \varepsilon \sum_{i \in I} \sum_{j \in J_i} \sum_{k \in K} x_{ijk} \tag{4.1}$$

subject to

$$1 \leq \sum_{p \in V_{i0k}} x_{ipk} \qquad \forall i \in I, k \in K \tag{4.2}$$

$$x_{ijk} \leq \sum_{p \in V_{ijk}} x_{ipk} \qquad \forall i \in I, j \in J_i, k \in K : e_{ij} + o_k \leq T \tag{4.3}$$

$$x_{ijk} \leq y^D_{l_{ij}} \cdot d_{ij} + y^N_{l_{ij}} \cdot (1 - d_{ij}) \qquad \forall i \in I, j \in J_i, k \in K \tag{4.4}$$

$$\sum_{k \in K} x_{ijk} v_k \leq e_{ij} - s_{ij} \qquad \forall i \in I, j \in J_i \tag{4.5}$$

$$\sum_{l \in L} y^D_l \leq L^D_{max} \tag{4.6}$$

$$x_{ijk} \in \{0,1\}, \;\; y^D_l \in \{0,1\} \tag{4.7}$$

The objective function (4.1) minimizes the number of nighttime maintenance activities. The second term penalizes every maintenance activity with an arbitrarily small penalty cost $\varepsilon$ in order to avoid unnecessary maintenance activities being performed. Constraints (4.2) and (4.3) enforce that intervals between successive maintenance activities are satisfied. Constraints (4.4) ensure that maintenance can only be executed at a location that is opened. Constraints (4.5) take account of the requirement that the duration of maintenance may not exceed the total time of an MO. The number of locations for daytime maintenance is restricted by constraint (4.6). Constraints (4.7) ensure that the integer decision variables are also binary.

## 4.3 Related models

The **MSLCP** model is the core of the current research. Nonetheless, in the course of the research process, the following two alternatives to the **MSLCP** were developed as well. For these models, no results have been generated. However, to account for their development, their formulations are provided in the current section. It entails the following two models:

1. **MSLCP-P**: A disadvantage of the **MSLCP** is that it may not result in a feasible solution. After all, the given rolling stock circulation may be such that it is impossible to find a maintenance schedule that matches the interval constraints. This occurs for example if the time between two MOs of sufficient length to perform maintenance activities exceeds the maximum interval between consecutive maintenance activities. To overcome this problem, the **MSLCP** is extended to take into account penalties. The corresponding problem is called the **MSLCP-P**.
2. **MSLCP-ND**: The **MSLCP** considers the daytime maintenance locations to be variable and assumes the nighttime maintenance locations are fixed. In the **MSLCP-ND**, the nighttime locations are considered to be variable as well.

### 4.3.1 MSLCP-P model

The **MSLCP** treats the maintenance interval constraints as hard constraints. This implies that no interval constraint violations are allowed. However, in some cases, a violation of this interval constraint is desired. An example of such a case is when there are no suitable maintenance opportunities within the desired horizon. In such a case, the standard model would find no solutions. To still find solutions in such a case, a model is developed that incurs penalties for constraint violations. More specifically, intervals between consecutive maintenance opportunities that exceed the maximum admissible interval can be accepted by the model at a cost. The **MSLCP-P** is an extension to the **MSLCP**, allowing for violations of interval constraints at a penalty cost.

Let $h_k$ be the extra interval length in which maintenance opportunities can be scheduled at a cost. This means that, for a given planned maintenance activity, the next maintenance activity may be at most $o_k + h_k$ hours later (whereas in in the former model, this was restricted to $o_k$ hours). To incur penalty costs, the set $V_{ijk}$ needs to be redesigned. Let $V^+_{ijk}$ be the analogue to $V_{ijk}$ and let it be defined as follows: $V^+_{ijk} = \{p \in J_i : e_{ij} < s_{ip} \leq e_{ij} + o_k + h_k\}$. Observe that $V_{ijk} \subseteq V^+_{ijk}$. Let $V^+_{i0k} = \{p \in J_i : s_{ip} \leq o_k + b_i k + h_k\}$. Let $r$ be an index to identify the elements of $V^+_{ijk}$, $r \in \{1, ..., |V^+_{ijk}|\}$. Let $c_{ijkr}$ be the penalty cost incurred when the $r$th element of $V^+_{ijk}$ is to be scheduled after the $j$th MO for rolling stock unit $i$ and maintenance type $k$. Moreover, let $p_{ijkr} \in J_i$ be the index to refer to the specific MO $j \in J_i$ corresponding to the $r$th element of the set $V^+_{ijk}$. Also, let $z_{ijkr} \in \{0, 1\}$ be an artificial decision variable that takes the value 1 if and only if the $r$th element of $V_{ijk}$ is to be chosen.

Below follows the model formulation of the **MSLCP-P**.

$$\min \sum_{i\in I}\sum_{j\in J_i}\sum_{k\in K} x_{ijk}(1-d_{ij}) + \sum_{i\in I}\sum_{j\in J_i}\sum_{k\in K}\sum_{r\in V^+_{ijk}} z_{ijkr}c_{ijkr} + \varepsilon \sum_{i\in I}\sum_{j\in J_i}\sum_{k\in K} x_{ijk} \tag{4.8}$$

subject to

$$1 \leq \sum_{p\in V^+_{i0k}} x_{ipk} \qquad \forall i \in I, k \in K \tag{4.9}$$

$$x_{ijk} \leq \sum_{p\in V^+_{ijk}} x_{ipk} \qquad \forall i \in I, j \in J_i\backslash\{1\}, k \in K : e_{ij} + o_k + h_k \leq T \tag{4.10}$$

$$x_{ijk} \leq y^D_{l_{ij}} \cdot d_{ij} + y^N_{l_{ij}} \cdot (1-d_{ij}) \qquad \forall i \in I, j \in J_i, k \in K \tag{4.11}$$

$$\sum_{k\in K} x_{ijk}v_k \leq e_{ij} - s_{ij} \qquad \forall i \in I, j \in J_i \tag{4.12}$$

$$\sum_{l\in L} y^D_l \leq L^D_{max} \tag{4.13}$$

$$\sum_{r\in V_{ijk^+}} z_{ijkr} = x_{ijk} \qquad \forall i \in I, j \in J_i, k \in K \tag{4.14}$$

$$z_{ijkr} \leq x_{ip_{ijkr}k} \qquad \forall i \in I, j \in J_i, k \in K, r \in V^+_{ijk} \tag{4.15}$$

$$x_{ijk},\, y^D_l,\; z_{ijkr} \in \{0,1\} \tag{4.16}$$

Compared to the objective of the **MSLCP** (4.1), this objective function (4.8) has extra terms to incorporate penalties for each maintenance interval exceedance: more specifically, the penalty cost associated to the first element of the set $V^+_{ijk}$ which was scheduled is incurred, on the condition that MO $j$ for rolling stock unit $i$ was scheduled for maintenance type $k$.[1] Constraints (4.9) and (4.10) are similar to Constraints (4.2) and (4.3) from the **MSLCP**, but in the current model the set of admissible maintenance opportunities is extended to $V^+_{ijk}$.

[1] It may occur that for some scheduled MO $j$ for rolling stock unit $i$ for maintenance type $k$, multiple 'next' MOs in $V^+_{ijk}$ were scheduled. In that case, only the cheapest penalty cost should be incurred. Typically, this will be the penalty cost associated to the *first* MO following the MO $j$ for rolling stock unit $i$ for maintenance type $k$ needs to be incurred. This behaviour is guaranteed since Constraints (4.14) ensure that only one penalty cost is incurrend and the objective function (4.8) ensures that the cheapest penalty cost is incurred.

Constraints (4.11)-(4.13) are equal to Constraints (4.4)-(4.6) from the former model. Constraints (4.14) and (4.15) ensure that, if MO $j$ is scheduled for rolling stock unit $i$ for maintenance type $k$, the variable $z_{ijkr}$ is set correctly to identify which next MO is scheduled as well. Observe that $z_{ijkr} = 1$ for the MO in the set $V^+_{ijk}$ that follows MO $j$ for rolling stock unit $i$ of type $k$. Constraints (4.16) ensure that the integer decision variables are also binary.

### 4.3.2 MSLCP-ND model

In the **MSLCP**, the opened locations during nighttime were fixed. However, nighttime maintenance is costly; hence, it is desirable to minimize the number of nighttime locations. The **MSLCP-ND** addresses this problem, treating both the daytime and nighttime locations as decision variables. For that purpose, the parameter $y_l^N$ from the **MSLCP** becomes a decision variable in the **MSLCP-ND**.

To incorporate the objective of minimizing the number of nighttime maintenance locations, the model can be formulated as follows.

$$\min \sum_{l \in L} y_l^N \tag{4.17}$$

$$1 \leq \sum_{p \in V_{i0k}} x_{ipk} \qquad \forall i \in I, k \in K \tag{4.18}$$

$$x_{ijk} \leq \sum_{p \in V_{ijk}} x_{ipk} \qquad \forall i \in I, j \in J_i, k \in K : e_{ij} + o_k \leq T \tag{4.19}$$

$$x_{ijk} \leq y^D_{l_{ij}} \cdot d_{ij} + y^N_{l_{ij}} \cdot (1 - d_{ij}) \qquad \forall i \in I, j \in J_i, k \in K \tag{4.20}$$

$$\sum_{k \in K} x_{ijk} v_k \leq e_{ij} - s_{ij} \qquad \forall i \in I, j \in J_i \tag{4.21}$$

$$\sum_{l \in L} y_l^D \leq L^D_{max} \tag{4.22}$$

$$x_{ijk},\, y_l^D,\, y_l^N \in \{0, 1\} \tag{4.23}$$

Compared to the **MSLCP**, the objective has changed into minimizing the number of nighttime maintenance locations. The constraints (4.18)-(4.22) are identical to Constraints (4.2)-(4.6) from the **MSLCP**.

This model is still expected to result in a solution that tries to minimize the number of nighttime maintenance activities, since the minimization of nighttime maintenance activities contributes to the goal of minimizing nighttime maintenance locations. As such, the objective of the **MSLCP** is still pursued at least partially in the **MSLCP-ND**. However, unlike the **MSLCP**, the **MSLCP-ND** does not have a specific incentive to minimize nighttime maintenance activities: if a nighttime maintenance location needs to be opened anyway, the model has no incentive to further minimize the number of nighttime maintenance activities.

| | variable | significance |
|---|---|---|
| **Sets** | $I$ | The set of rolling stock units considered in the current problem. |
| | $J_i$ | The set of all MOs for rolling stock unit $i$ |
| | $K$ | The set of maintenance activity types. |
| | $L$ | The set of potential maintenance locations. |
| | $V_{ijk} \subseteq J_i$ | The set of MOs of which at least one should be used for maintenance type $k$ for rolling stock unit $i$ if maintenance type $k$ was performed in MO $j$ for rolling stock unit $i$. |
| | $V^+_{ijk} \subseteq J_i$ | The analogue to $V_{ijk}$, but allowing for sequences of maintenance activities that exceed the maximum interval |
| **Indices** | $i \in I$ | Index used to identify any rolling stock unit |
| | $j \in J_i$ | Index used to identify any MO |
| | $p_{ijkr} \in J_i$ | Index used to refer to the MO $j \in J_i$ that is the $r$th element of the set $V^+_{ijk}$ |
| | $k \in K$ | Index used to identify any maintenance activity type. |
| | $l \in L$ | Index used to identify any potential maintenance location. |
| | $l_{ij} \in L$ | Index used to identify the location corresponding to MO $j$ for rolling stock unit $i$. |
| **Parameters** | $c_{ijkr}$ | Penalty cost incurred when the $r$th element of $V^+_{ijk}$ is to be scheduled after the $j$th MO. |
| | $d_{ij} \in \{0,1\}$ | Binary input parameter used to indicate whether MO $i$ for rolling stock unit $j$ is during the day ($d_{ij} = 1$) or during the night ($d_{ij} = 0$). |
| | $e_{ij} \in \mathbb{R}$ | The end time of MO $j$ for rolling stock unit $i$. |
| | $o_k \in \mathbb{R}^+$ | The maximum interval between two subsequent maintenance activities of type $k$. |
| | $s_{ij} \in \mathbb{R}$ | The start time of MO $j$ for rolling stock unit $i$ |
| | $\delta^D \in (0,24)$ | The hour when the daytime maintenance window starts. |
| | $\delta^N \in (0,24)$ | The hour when the nighttime maintenance window starts. |
| | $v_k \in \mathbb{R}^+$ | The duration of maintenance activity of type $k$ in hours. |
| | $y^N_l$ | Binary input variable to 1 if location $l \in L$ is available for nighttime maintenance and 0 otherwise (except for **MSLCP-ND**). |
| **Decision variables** | $x_{ijk} \in \{0,1\}$ | Binary decision variable equal to 1 if maintenance of typekis performed to rollingstock unit $i \in I$ at MO $j \in J_i$, and 0 otherwise. |
| | $y^D_l$ | Binary decision variable to 1 if location $l \in L$ is available for daytime maintenance and 0 otherwise. |
| | $y^N_l$ | Binary decision variable to 1 if location $l \in L$ is available for nighttime maintenance and 0 otherwise (only in **MSLCP-ND**). |
| | $z_{ijkr}$ | Binary artificial variable equal to 1 if MO $j$ for rolling stock $i$ for type $k$ is chosen, and the following MO used by this rolling stock unit for this maintenance type is the $r$th element from the set $V^+_{ijk}$ |

*Table 4.4: Mathematical notation in the **MSLCP** model, for reference.*

# Chapter 5

# Activity planning in maintenance shifts

Although the previous chapter has resulted in a model to schedule maintenance activities by assigning them to maintenance opportunities, it has not addressed the actual planning of these maintenance activities on the level of the maintenance locations. Neither has it confronted the associated required capacity at maintenance locations to execute a given maintenance schedule. The current chapter addresses these aspects by introducing the *Activity Planning Problem* (**APP**) and proposing a model for it.

This chapter is organized as follows. Section 5.1 discusses the motivation for the current chapter and introduces the **APP**. Section 5.2 discusses the characteristics of the **APP** in detail. Then, Section 5.3 describes how the **APP** can be used to analyse solutions of the **MSLCP**, and more specifically, how solutions of the **MSLCP** need to be prepared so that they can be handled by the **APP** model. Section 5.4 gives a formal mathematical definition of a model for the **APP**. For reference, Table 5.6 gives an overview of all mathematical notation used in the **APP** model with its meaning.

## 5.1 Lead-in

The **MSLCP** assigns maintenance activities to maintenance opportunities in such a way that intervals between maintenance activities are satisfied. Maintenance activities are not scheduled accurate to the minute, but rather they are assigned to an MO and the maintenance activity has to be performed anywhere in the MO. This is considered a justified choice: in the first place, railway operators tend to work in a similar way in the planning of maintenance activities, especially when the considered time horizon is multiple weeks (as in the current case) and the activities spread out over the entire network; in the second place, scheduling activities accurate to the minute would add considerably to the computational tractability of the problem.

However, from the perspective of a specific location on a specific day, the assignment of a maintenance activity to a maintenance opportunity often does not provide sufficient information to create a feasible planning for a *maintenance shift* (referred to as the *(maintenance) shift planning*). A maintenance shift is a fixed period of the day for which a planning of resources is made. Usually the maintenance teams change before and after shifts, but stay the same during shifts. A maintenance shift planning assigns work to maintenance teams. In order to create a good maintenance shift planning, two problems play a role that are not addressed in the **MSLCP**.

- For a specific maintenance shift, dispatchers of every maintenance location need to determine for each activity when it is to be performed, taking into account the time it is planned to arrive at the maintenance location and the time it needs to depart from the maintenance location again.

- The number of maintenance teams depends on the actual planning that can be made. This number is not straightforward to determine, though very relevant since it is an important measure of the required capacity.

To address the two problems identified above, the *Activity Planning Problem* (**APP**) is defined. The input of the **APP** is a set of jobs that need to be performed and a maximum number of maintenance teams available. A job represents the activities that need to be performed to one rolling stock unit during a specified maintenance opportunity. A job contains one maintenance activity of a specific maintenance type, but can also contain multiple maintenance activities of different maintenance types. The set of jobs can (but need not necessarily) originate from the **MSLCP**, since the **MSLCP** presents a schedule of maintenance activities, from which the maintenance jobs that need to be performed on each maintenance location can be inferred. The output of the **APP** is twofold: first, it gives the minimum number of teams necessary to perform the given set of jobs, and second it gives the corresponding optimal activity planning, defining the start and end times of each job. The **APP** shows similarities with the class of *Parallel Machine Scheduling Problems*, as addressed by for example Kravchenko and Werner (2009).

The functioning of the **APP** in conjunction with the **MSLCP** is graphically presented in Figure 5.1. This clearly shows that the **APP** is an addition to the **MSLCP**. Note the similarities with Figure 3.1 from Chapter 3, which introduced the inputs and outputs of the **MSLCP**.

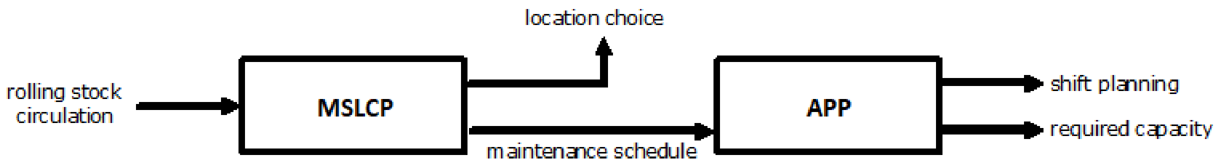



*Figure 5.1: Graphical representation of the input and output of the **APP** and how this functions in conjunction with the **MSLCP**. Although it must be noted that the **APP** takes some additional parameters as input which are omitted in this figure, the characterizing input is a set of jobs which, in case of cooperation with the **MSLCP**, is the maintenance schedule for a specific maintennce location.*

## 5.2 APP problem characteristics

This section defines the most important notions relevant in the **APP**.

### Jobs

The **APP** takes a set of jobs as input. For each job in this set, the following things need to be specified.

- The release time, indicating when the job becomes available. In the current research, this corresponds to the moment a rolling stock unit arrives at a maintenance location and hence becomes available for maintenance.

- The deadline time, indicating when the job needs to be finished. In the current research, this corresponds to the moment a rolling stock unit departs from a maintenance location and hence to the moment when all maintenance activities need to be finished.

- The duration of the job, equal to the time required for all maintenance activities contained in the job.

It is assumed that for each job, the time between the release and deadline of the job is larger than or equal to the job duration. If there would exist any job for which this were not the case, solving the **APP** would make no sense because from the start it could already be identified that no feasible solution exists. This is a general assumption that is automatically satisfied in the context of the **MSLCP**, since the **MSLCP** does not provide any solutions in which the total time necessary for maintenance exceeds the total time available for maintenance.

#### Maintenance teams

Each job needs to be performed by one and only one maintenance team. The team works on this job uninterruptedly, i.e. the job cannot be split into multiple separate parts (meaning preemption is not allowed). Also, it is assumed that for each job, one team is necessary. The total number of maintenance teams available is specified as an input parameter.

#### Scheduling

The maintenance jobs are assigned to maintenance teams, and the start time of each maintenance job is determined. The end time of the job is then automatically determined by adding the job duration to the start of the maintenance job. The start time should be such that it is after the release time of a job, and such that the end time is before the deadline of a job.

#### Objective

The objective is to minimize the number of available maintenance teams necessary. Note that the maximum number of maintenance teams available is specified as an input parameter. This implies that, if a given set of jobs requires a number of maintenance teams that exceeds the specified maximum number of maintenance teams, the **APP** results in an infeasible solution. It may however also happen that a given set of jobs requires a number of maintenance teams that is lower than the specified maximum number of maintenance teams. In that case, still a solution is provided that requires the minimum number of maintenance teams, and the 'extra' maintenance teams are idle (i.e. no jobs assigned to these maintenance teams).

## 5.3 From MSLCP output to APP input

Although the **APP** can be used separately, with a generic set of jobs as input, it can also determine the set of jobs based on the maintenance schedule that the **MSLCP** provides and determine a maintenance shift planning and capacity requirements for the given **MSLCP** solution. To this end, the output of the **MSLCP** needs to be prepared to be used as input for the **APP**.

#### Maintenance shifts

The output of the **MSLCP** is an assignment of maintenance activities to maintenance opportunities over the entire time horizon. The planning of maintenance activities, however, is usually performed per maintenance shift. A maintenance shift is a fixed period of the day for which a maintenance schedule is made. The current research assumes two maintenance shifts: the daytime maintenance shift between 7.00 and 19.00 and the nighttime maintenance shift between 19.00 and 7.00.

Unique maintenance shifts are characterised by the following three variables:

- *Maintenance location.* Maintenance shifts are defined for any maintenance location from the set of potential maintenance locations. For example 'Utrecht'.
- *Time window.* There are two types of maintenance shifts: daytime maintenance shifts and nighttime maintenance shifts.
- *Reference day.* The reference day is the day when the maintenance shift *starts*. This is necessary since the night shift covers two days: the time interval between 19.00 and 24.00 is a different day than the time interval between 24.00 and 19.00 on the next day, although these time intervals correspond to the same maintenance shift.

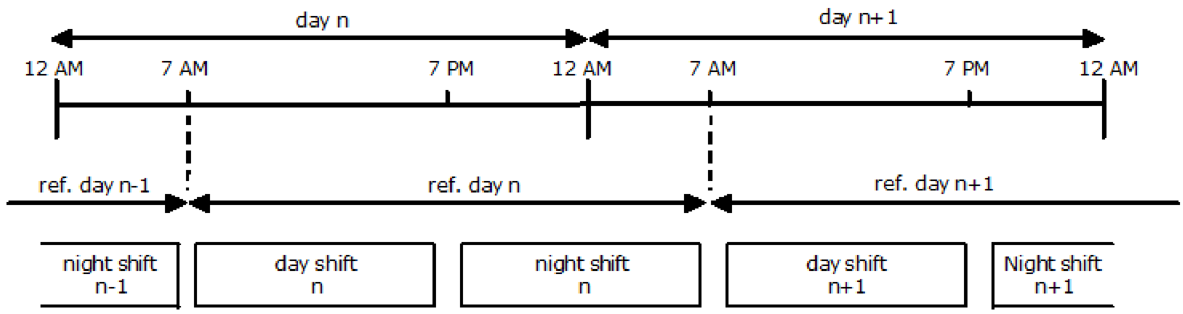



*Figure 5.2: Division of the day into maintenance shifts. In this figure the daytime time window interval is from 07.00 to 19.00 and the nighttime time window is from 19.00 to 07.00. This figure shows two days, $n$ and $n + 1$. It is indicated how the reference day is constructed, and how this reference day is used to classify the various maintenance shifts.*

Figure 5.2 indicates how two days are divided into daytime and nighttime maintenance shifts. It shows that the reference day is constructed in such a way that reference days start at 07.00 and end at 07.00 the next morning, and it is shown that this allows to refer to daytime and nighttime maintenance shifts by the reference day.

An example of a unique maintenance shift would be *the night shift in Amsterdam on day 3*, meaning the shift that starts in Amsterdam at 19.00 on day 3 and ends in Amsterdam at 07.00 on day 4. It is important to note that maintenance shifts are location-specific. Hence, the nighttime maintenance shift on reference day $n$ in Amsterdam and the nighttime shift on reference day $n$ in Utrecht are not the same.

### Assignment of maintenance activities to maintenance shifts

The **APP** can be used to determine the planning of maintenance activities and the minimum number of required teams for some given shift, for a given a solution of the **MSLCP**. Recall that the **APP** requires as an input a list of maintenance jobs. To this end, for a given shift and solution of the **MSLCP** a maintenance job list must be constructed.

In order to do this, for each maintenance activity it needs to be determined in which maintenance shift it is to be performed. This is not always straightforward, since maintenance activities are assigned to MOs, and one MO may cover multiple maintenance shifts (for example when it starts at 15.00 and ends at 20.00, covering a daytime and the subsequent nighttime maintenance shift).

Recall that an MO is characterized to be during daytime if and only if its start and end time are both during the same daytime time window (that is, the MO is contained within one daytime maintenance shift) and that an MO is during nighttime in all other cases.

The following procedure is used to determine to what maintenance shift an MO belongs.

- Suppose an MO is classified as a daytime MO. Then, by the definition of daytime MOs, it is clear that the entire MO is contained within the daytime time window. The reference day is therefore equal to the end time of the MO and it belongs to the daytime maintenance shift of that particular day.
- Suppose an MO is classified as a nighttime MO. Note that this does not necessarily mean that the end time is during nighttime (for example, an MO starting during nighttime and ending during daytime is still classified as a nighttime MO). For nighttime MOs, it seems reasonable to assign these MOs to the last nighttime maintenance shift that it was in. In other words, if the end time is between 0.00 and 19.00, it is classified as an MO during the nighttime shift with a reference day at the *previous day*; if, on the other hand, the end time is between 19.00 and 0.00, this last maintenance shift is the nighttime maintenance shift with reference day on the *current day*.

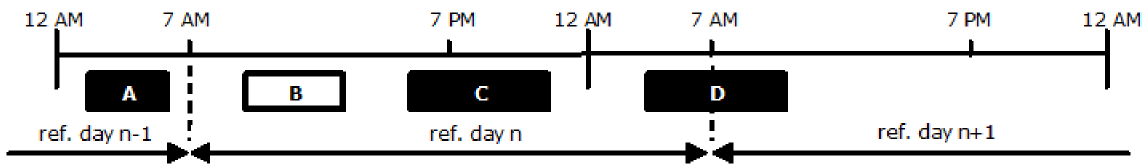



*Figure 5.3: Assignment of MOs to shifts. This figure presents four MOs (A, B, C and D), of which B is classified as daytime and MO and A, C and D are classified as nighttime MOs in the* ***MSLCP****.*

An example is found in Figure 5.3. This figure presents four MOs. Based on the above described procedure, these MOs can be assigned to maintenance shifts.

- MO $A$ is a nighttime maintenance shift with end time on day $n$ before 19.00. Hence, it is assigned to the nighttime shift of the previous day: night $n-1$.
- MO $B$ is a daytime maintenance shift with end time on day $n$. Hence, it is assigned to the daytime shift of the current day, day $n$.
- MO $C$ is a nighttime maintenance shift with end time on day $n$ after 19.00. Hence, it is assigned to the nighttime shift of the current day: night $n$.
- MO $D$ is a nighttime maintenance shift with end time on day $n+1$ before 19.00. Hence, it is assigned to the nighttime shift of the previous day: night $n$.

### Release and deadline times

For each of the maintenance jobs considered by the **APP**, a release time and a deadline time need to be specified.

For most maintenance jobs considered in the current context, the release time and deadline time can be easily inferred from the corresponding MO. As described above, all maintenance activities are assigned to shifts, and the maintenance activity needs to be performed in this specific shift. In many cases the MO is fully contained in the shift. Take for example MO $A$ and $B$ in Figure 5.3. The start time of $A$ is after the start of nighttime maintenance $n-1$ and the end time of $A$ is before the end of nighttime maintenance shift $n-1$. Similarly, for MO $B$, its start is after the start of daytime maintenance shift $n$ and its end is before the end of daytime maintenance shift $n$. For these cases, the release and deadline time of the corresponding maintenance job is straightforward: the release time is equal to the start time of the MO and the deadline time is equal to the end time of the MO.

However, there are also MOs that are not fully contained in the corresponding maintenance shift, such as MOs $C$ and $D$ in Figure 5.3. Still, they are assigned to nighttime maintenance shift $n$ and therefore need to be performed in this shift. (Note that there may be cases where maintenance activities in MOs like $C$ and $D$ could be performed during daytime, but this requires that the maintenance locations of MOs $C$ and $D$ are opened during the night in the corresponding **MSLCP** solution, which is not guaranteed.)

In order to make sure that maintenance activities are performed as much as possible in the maintenance shift that they were assigned to, the following rules are used to determine the release times.

- If an MO is a daytime MO, then the release time of the corresponding maintenance job is equal to the start of the MO.
- If an MO is a nighttime MO and the start of the MO is after the start of the maintenance shift, then the release time of the corresponding maintenance job is equal to the start of the MO.
- If an MO is a nighttime MO and the start of the MO is before the start of the maintenance shift, then the release time of the corresponding maintenance job is set to the start of the maintenance shift (usually 19.00). There is one exception to this rule: when, by setting the release time to 19.00, the time available for maintenance (i.e. between the end of the MO and 19.00) is less than the duration of the maintenance, then the release time is set to end time minus the total duration of maintenance in this job. Suppose that the maximum duration of maintenance in a job is 90 minutes, than this means that in the extreme case maintenance activities in the nighttime maintenance shift may already be planned to start at 17.30 (in the extreme case that the end of the MO would be exactly at 19.00).

A similar, symmetric set of rules prevails for the determination of the deadline moment.

- If an MO is a daytime MO, then the deadline time of the corresponding maintenance job is equal to the end of the MO.
- If an MO is a nighttime MO and the end of the MO is before the end of the maintenance shift, then the release time of the corresponding maintenance job is equal to the start of the MO.
- If an MO is a nighttime MO and the end of the MO is after the end of the maintenance shift, then the deadline time of the corresponding maintenance job is set to the end of the maintenance shift (usually 07.00). There is one exception to this rule: when, by setting the deadline time to 07.00, the time available for maintenance (i.e. between the start of the MO and 07.00) is less than the duration of the maintenance, then the deadline time is set to start time plus the total duration of maintenance in this job. Suppose that the maximum duration of maintenance in a job is 90 minutes, than this means that in the extreme case maintenance activities in the nighttime maintenance shift may be planned to end at 8.30 (in the extreme case that the start of the MO would be exactly at 7.00).

It must be noted that in most cases, MOs are contained in either the daytime shift or the nighttime shift. In these cases, the release time is equal to the start of the MO and the deadline time is equal to the end of the MO. The other cases relate to MOs that start before the peak and end after the peak (i.e. that contain a peak), and this does not occur often since during peak hours most rolling stock units are in use.

#### Job durations

Each job corresponds to the maintenance activities in one MO. It is assumed that all maintenance activities in an MO are performed sequentially without interruption. Hence, a maintenance job does not distinguish between activities of different maintenance types. The duration of a job is therefore equal to the sum of the durations of all maintenance activities scheduled by the **MSLCP** in an MO.

## 5.4 Mathematical formulation APP

### 5.4.1 Mathematical notation

#### Jobs

Let $J$ be a given set of jobs that need to be scheduled, and for each job $j \in J$ let the release time $r_j \in \mathbb{R}$, the deadline time $t_j \in \mathbb{R}$ and the duration $v_j \in \mathbb{R}$ be given.[1]

#### Teams

Let $\overline{N}$ be the maximum number of available maintenance teams and define $N = \{1, ..., \overline{N}\}$ to be the set of maintenance teams.

#### Moments

The current formulation of the **APP** uses so-called *moments*. A moment represents the opportunity of a maintenance team to start a job. This is a construct used to model the **APP** as a linear problem. Each team has a set of moments available, corresponding to the maximum number of jobs that they can perform. To any moment, a job can be assigned. If a job is assigned to a moment, the start time of this particular moment is associated to the start time of the corresponding maintenance job. The introduction of the concept of moments allows to model a sequential planning, by requiring that if a job is assigned to moment $m$, moment $m+1$ can start only after the job assigned to moment $m$ is finished.

Let $\overline{M}$ be the number of moments available per team. Note that the maximum number of moments used by a team occurs when a team is continually occupied with maintenance activities of the shortest duration for the entire length of the maintenance shift. A sufficiently large $\overline{M}$ can thus be obtained by dividing the total time available in a maintenance shift over the minimum time required for each maintenance job. Moreover, the number of moments necessary never exceeds the total number of jobs. Based on these two indications, Equation (5.1) gives an appropriate value for $\overline{M}$ that is used in the current research.

$$\overline{M} = \min\left(\left\lceil \frac{\delta^N - \delta^D}{\min_{k\in K} v_k} \right\rceil, |J|\right) \tag{5.1}$$

Define $M = \{1, ..., \overline{M}\}$ to be the set of moments.

[1] Note that the set of jobs $J$ is not the same set $J$ as used in Chapter 4 to denote the set of MOs.

#### Objective

The objective is to minimize the number of maintenance teams.

### 5.4.2 APP model formulation

Let $x_{nmj} \in \{0,1\}$ be a binary variable that signifies whether moment $m$ for team $n$ is associated to job $j$: let $x_{nmj} = 1$ if and only if team $n$ at moment $m$ processes job $j$. Let $s_{nm} \in \mathbb{R}$ be the start time of the moment $m$ for team $n$. Let $y_n \in \{0,1\}$ be a binary variable that signifies whether team $n$ is active or not: let $y_n = 1$ if team $n$ is used for this schedule.

The **APP** model can then be formulated as follows.

$$\min \sum_{n \in N} y_n \tag{5.2}$$

subject to

$$\sum_{j \in J} x_{nmj} r_j \leq s_{nm} \leq \sum_{j \in j} x_{nmj}(t_j - v_j) \qquad \forall n \in N, m \in M \tag{5.3}$$

$$s_{n,m+1} \geq s_{nm} + \sum_{j \in J} x_{nmj} v_j \qquad \forall n \in N, m \in \{1, ..., \overline{M} - 1\} \tag{5.4}$$

$$\sum_{n \in N} \sum_{m \in M} x_{nmj} = 1 \qquad \forall j \in J \tag{5.5}$$

$$\sum_{j \in J} x_{nmj} \leq 1 \qquad \forall n \in N, m \in M \tag{5.6}$$

$$\sum_{m \in M} \sum_{j \in J} (y_n - x_{nmj}) \geq 0 \qquad \forall n \in N \tag{5.7}$$

$$x_{nmj} \in \{0,1\},\ y_{nm} \in \{0,1\},\ \ s_{nm} \in \mathbb{R} \tag{5.8}$$

The objective (5.2) minimizes the number of teams necessary. Constraints (5.3) guarantee that the start moment is after the release time of the corresponding job and before the latest start moment for the corresponding job (i.e. the deadline minus the duration). Constraints (5.4) enforce that the start moments for one team are sufficiently far apart so that maintenance activities do not overlap. Constraints (5.5) ensure that every job is assigned to exactly one moment. Constraints (5.6) make sure that each moment is used for at most one job. Constraints (5.7) establish that a team can only be used if it is 'active'.

## 5.5 APP example

To demonstrate the workings of the **APP**, two examples are given. Table 5.1 gives the input for Example 1. It demonstrates three jobs, the first one available between 1 and 3 (i.e. 01.00 and 03.00), the second between 5 and 10 (i.e. 05.00 and 10.00), the third between 6 and 10 (i.e. 06.00 and 10.00). The durations of all jobs are 2 hours.

| $j$ | release $r_j$ | deadline $t_j$ | duration $v_j$ |
|---|---|---|---|
| 1 | 1 | 3 | 2 |
| 2 | 5 | 10 | 2 |
| 3 | 6 | 10 | 2 |

*Table 5.1: Input for Example 1. Indicated are the various jobs $j \in J$ and their corresponding release time $r_j$, deadline time $t_j$ and duration $v_j$.*

The resulting maintenance shift planning is given in Table 5.2: the first job is performed between 1 and 3, the second job between 6 and 8, and then the third job between 8 and 10. There is only one maintenance team needed for this maintenance planning.

| $j$ | release $r_j$ | deadline $t_j$ | duration $v_j$ | team $n_j$ | moment $m_j$ | start $s_{n_j m_j}$ | end $s_{n_j m_j} + v_j$ |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 3 | 2 | 1 | 2 | 1.0 | 3.0 |
| 2 | 5 | 10 | 2 | 1 | 4 | 8.0 | 10.0 |
| 3 | 6 | 10 | 2 | 1 | 3 | 6.0 | 8.0 |

*Table 5.2: Output for Example 1. Indicated are the various jobs $j \in J$ and their corresponding release time $r_j$, deadline time $t_j$ and duration $v_j$, the team number by which it is performed $n_j$, the moment id $m_j \in M$, the start moment time $s_{n_j m_j}$ and the end time (computed by adding the duration to the start time).*

Now consider an adaptation to the problem in which the maintenance durations are increased to 4 hours for job 2 and job 3. See Table 5.3 for the new input.

| $j$ | release $r_j$ | deadline $t_j$ | duration $v_j$ |
|---|---|---|---|
| 1 | 1 | 3 | 2 |
| 2 | 5 | 10 | 4 |
| 3 | 6 | 10 | 4 |

*Table 5.3: Input for Example 2. Indicated are the various jobs $j \in J$ and their corresponding release time $r_j$, deadline time $t_j$ and duration $v_j$.*

Clearly, the previous maintenane shift planning is not feasible anymore since there is a conflict for maintenance job 2 and 3. Hence, two maintenance teams are needed. Table 5.4 gives the output for the second example and shows that indeed, two teams are used. The first activity and the second activity are both performed between 6 and 10, but by different maintenance teams (indicated by the different value for $n$).

| $j$ | release $r_j$ | deadline $t_j$ | duration $v_j$ | team $n_j$ | moment $m_j$ | start $s_{n_j m_j}$ | end $s_{n_j m_j} + v_j$ |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 3 | 2 | 1 | 3 | 1.0 | 3.0 |
| 2 | 5 | 10 | 4 | 2 | 4 | 6.0 | 10.0 |
| 3 | 6 | 10 | 4 | 1 | 4 | 6.0 | 10.0 |

*Table 5.4: Output for Example 2. Indicated are the various jobs $j \in J$ and their corresponding release time $r_j$, deadline time $t_j$ and duration $v_j$, the team number by which it is performed $n_j$, the moment id $m_j \in M$, the start moment time $s_{n_j m_j}$ and the end time (computed by adding the duration to the start time).*

| | variable | significance |
|---|---|---|
| **Sets** | $J$ | The set of maintenance jobs |
| | $N$ | The set of maintenance teams |
| | $M$ | The set of moments |
| **Indices** | $j \in J$ | Index used to identify any maintenance job |
| | $n \in N$ | Index used to identify any maintenance team |
| | $m \in M$ | Index used to identify any moment |
| **Parameters** | $r_j \in \mathbb{R}$ | The release time of maintenance job $j$ |
| | $t_j \in \mathbb{R}$ | The deadline time of maintenance job $j$ |
| | $v_j \in \mathbb{R}$ | The duration of maintenance job $j$ |
| | $\overline{M} \in \mathbb{N}^+$ | The number of moments considered for each maintenance team |
| | $\overline{N} \in \mathbb{N}^+$ | The available number of maintenance teams |
| **Decision variables** | $x_{nmj} \in \{0, 1\}$ | Binary decision variable equal to 1 if moment $m$ for maintenance team $n$ is used for maintenance job $j$ |
| | $s_{nm} \in \mathbb{R}$ | Continuous decision variable signifying the start time of moment $m$ for maintenance team $n$ |
| | $y_n \in \{0, 1\}$ | Binary decision variable equal to 1 if maintenance team $n$ is used |

*Table 5.6: Mathematical notation in the **APP** model, for reference.*

# Chapter 6

# Maintenance location capacity modelling

While Chapter 4 provided the **MSLCP** model to address the scheduling of maintenance activities and Chapter 5 proposed the **APP** model that is able to assess the capacity requirements (i.e. required number of teams) of any **MSLCP** solution, the current section integrates the two in one framework using an approach called Logic-Based Benders' Decomposition (LBBD), which is a generalization to the recognized method called Benders' Decomposition. The resulting model, the *Capacitated Maintenance Scheduling and Location Choice Problem* (**CMSLCP**) provides a method that allows to find a solution to the **MSLCP** that satisfies constraints on the capacity of maintenance locations.

Section 6.1 discusses the motivation for the current chapter and introduces the **CMSLCP**. Section 6.2 gives an introduction on Benders' decomposition in general and its logic-based version (Logic-Based Benders' Decomposition, LBBD). Then, Section 6.3 describes a solution framework to solve the **CMSLCP** using LBBD and Section 6.4 defines it formally. A crucial part in this framework is the generation of so-called *cuts*. Their quality highly influence the quality and efficiency of the **CMLSCP**. Section 6.5 proposes four methods to generate these cuts. For reference, Table 6.2 gives an overview of all mathematical notation used in the **CMSLCP** model with its meaning.

## 6.1 Lead-in

Chapter 4 provides a base model for the scheduling of maintenance tasks and the choice for maintenance locations, called the **MSLCP** model. This model does not address the capacity of maintenance locations: it assumes that the capacity of maintenance locations is unrestricted. This assumption can be justified in the light of the current situation at NS, where maintenance location capacity during nighttime is under pressure and maintenance location capacity during daytime is abundantly available. The goal of the **MSLCP** is to move as much work from nighttime to daytime, reducing the capacity usage during nighttime as much as possible, so capacity constraints on nighttime maintenance are not relevant. Moreover, since the capacity on daytime maintenance locations seems to be amply available, also capacity constraints for daytime maintenance do not seem to be necessary.

Nonetheless, taking capacity of maintenance locations into account is relevant. First, capacity constraints may actually play a role for other railway operators. Second, railway operators may want to exercise some control on the workload assigned to each maintenance location. Third, from a scientific point of view the incorporation of capacity constraints is interesting due to the extra complexity that it adds to the problem.

This extra complexity comes from the fact that capacity the capacity of any **MSLCP** solution is hard to determine. For example, for **MSLCP** solutions where many maintenance activities need to be performed at the same time within maintenance shifts (corresponding to a situation with peak hours for maintenance), more teams are necessary compared to solutions where maintenance are activities are evenly spread over maintenance shifts. Hence, the capacity depends on the optimal activity planning which is not readily available.

However, with the introduction of the **APP** model developed in Chapter 5 a method has become available to assess the complex capacity requirements of any **MSLCP** solution. In other words, the **APP** model is capable of post-processing any **MSLCP** solution to generate a measure of the capacity requirements in each maintenance shift. In this way, it can be checked for any solution of the **MSLCP** whether the available capacity is exceeded or not. Yet, if it turns out that the available capacity is exceeded, the **APP** is not capable of finding an **MSLCP** solution that is guaranteed to satisfy capacity constraints.

The purpose of this chapter is to find a solution to the **MSLCP** that does satisfy capacity constraints, where capacity constraints are restrictions to the total number of maintenance teams available at each locations and prevent the required capacity to exceed the available capacity. To this end, the current chapter introduces the *Capacitated Maintenance Scheduling and Location Choice Problem* (**CMSLCP**), which is an extension to the **MSLCP** including capacity constraints. It uses the **MSLCP** at the basis, but adds the *APP* model in the framework to be able to also consider capacity constraints. Figure 6.1 demonstrates the cooperation between the **MSLCP** and the **APP** to include capacity constraints. Note the similarities with Figure 5.1 from Chapter 5, which introduced how the **APP** can be used to post-process any **MSLCP** solution.

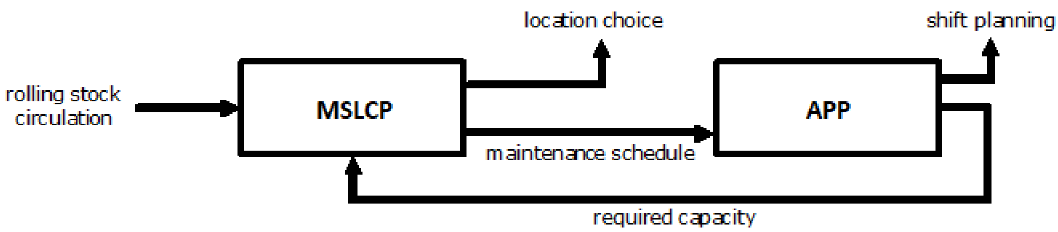



*Figure 6.1: Graphical representation of the **CMSLCP**, demonstrating how it combines the **MSLCP** and the **APP**.*

## 6.2 Benders' decomposition

Benders' decomposition (BD) is a method proposed by Benders (1962) and aims to efficiently solve large-scale linear optimization problems. The idea behind BD is to split the complete problem into a master problem, containing only a subset of the variables of the complete problem, but which is usually easier to solve than the complete problem, and a sub problem, containing the other variables. The master problem is solved first and then, given the candidate solution obtained from the master problem, the sub problem is solved. In this sub problem, the variables of the candidate solution given by the master problem are fixed. Using the dual of the sub problem, so-called *cuts* can be generated. These cuts serve to constrain the solution space of the master problem, so that solution regions in this space that are not feasible for the complete problem are disregarded. Then, the master problem is solved again, including the cuts generated in the first iteration. The master problem is now more restricted compared to the first time it was run, but the solution produced is closer to a feasible solution for the complete problem. The sub problem is run again, new cuts are generated, which are in turn added to the master problem, and so forth. Optimality is reached when the objective value of the master problem is equal to the objective value of the sub problem. In this case, the algorithm terminates.

The classical BD method requires that the sub problem be continuous and linear, so that standard linear duality can be used. The solution of the dual of the sub problem then provides efficient cuts. However, in many cases, the sub problem is not continuous or not linear. For such cases, one may resort to the *Logic-based Benders' decomposition* (LBBD), which is a generalization of the classical Benders' decomposition (Hooker, 2011). Since its introduction, it has been applied to problems in various areas, such as facility location management, radiation therapy and the dispatching of automated guided vehicles (Rahmaniani et al., 2017). The LBBD does not require that the sub problem take a specific form; it can potentially be non-linear and non-continuous. Cuts need to be generated for the sub-problem according to its specific structure.

A potential disadvantage of the LBBD compared to classical BD is that a cut generation procedure needs to be designed for every new type of sub-problem. Unlike in the classical BD, where cuts are always generated by solving the dual version of the sub problem, LBBD does not have such a standard procedure. This is often considered a disadvantage (Rahmaniani et al., 2017), although it may also offer advantages in that it allows to exploit the characteristics of the specific problem under consideration even more (Hooker, 2019).

Figure 6.2 indicates how the LBBD decomposes a problem into a master and a sub problem.

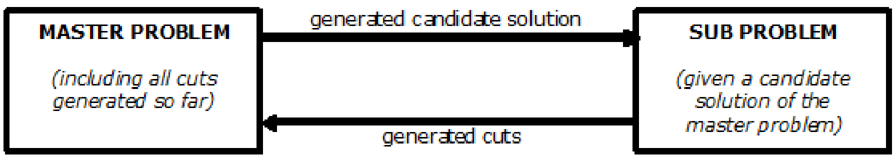



*Figure 6.2: The division of an optimization problem into a master and a sub problem, using the LBBD method. Note: the back-and-forth process between the master and sub problem terminates when the objective value of the master problem and the objective value of the sub problem are identical.*

The framework incorporates the following steps. Assume that the set of generated *cuts* is initially empty.

1. Solve the master problem subject to the current set of all generated cuts. Generate a candidate solution.

2. Solve the sub problem, given the candidate solution obtained in step (1). Generate new cuts constraining the search space in the master problem.

Terminate when the objective of the master problem is equal to the objective of the sub problem.

## 6.3 Solution framework

The current solution framework uses LBBD to solve the **CMSLCP** as a decomposition of the **MSLCP** (the master problem) and the **APP** (the sub problem). This composition is graphically represented in Figure 6.3. Notice the analogy with the general LBBD method presented in Figure 6.2.

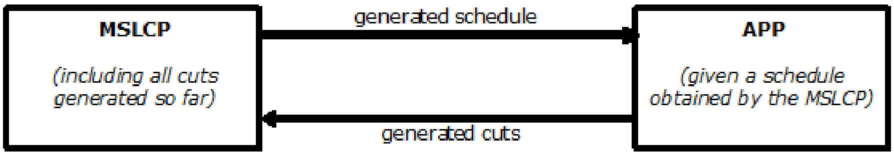



*Figure 6.3: The division of the* ***CMSLCP*** *into a master and a sub problem using the LBBD method.*

The framework for the **CMSLCP** incorporates the following steps. Assume that the set of generated *cuts* is initially empty.

1. Solve the **MSLCP** subject to the current set of all generated cuts. Generate a candidate solution, which is a maintenance schedule (i.e. an assignment of maintenance activities to MOs).

2. Given the maintenance schedule obtained by the **MSLCP**, solve the **APP** for every maintenance shift. If capacity during a maintenance shift is exceeded, generate new cuts constraining the search space of the **MSLCP**.

Terminate when the **APP** results in a feasible solution for all time shifts. In that case all constraints in the **MSLCP** and all additional constraints handled by the **APP** are satisfied and an optimal solution has been determined.

## 6.4 Mathematical formulation of the CMSLCP

### Maintenance shifts and maintenance tams

Let $S$ be the set of unique maintenance shifts (a concept defined in Chapter 5). Let $\overline{N}$ be the number of teams available at any location. Without loss of generality, in the current framework it is assumed that the available capacity (measured in the number of maintenance teams available) is equal for all maintenance shifts and that it is equal to one maintenance team ($\overline{N}$). It is usually straightforward to extend this concept to capacities of more than one maintenance team: after all, the **APP**, that is used to determine the capacity for each maintenance shift, can be run with any number of maintenance shift.[1]

### Master and sub problem solutions

The **CMSLCP** iteratively solves the master and sub problem. Let $x^k$ be the solution of the **MSLCP** after the $k$th iteration of the **CMSLCP**(i.e. this corresponds to a maintenance schedule, which is an assignment of maintenance activities to MOs). For a given solution $x^k$, the set of jobs can be determined for any shift. Let $J_{x^k}(s)$ be the set of jobs for shift $s \in S$, given the solution of the **MSLCP** $x^k$. Let **APP**$(J, \overline{N})$ be the objective value obtained after running the **APP** for set of jobs $J$. Use the notation **APP**$(J) = \infty$ if the **APP** for the set of jobs $J$ results in an infeasible solution. To describe the capacity required for a shift $s \in S$, given a solution $x^k$ of the master problem, the notation **APP** $(J_{x^k}(s))$ is used.

### Cuts

If **APP**$(J) = \infty$ for a given set $J$, it can be immediately concluded that the combination of jobs in the set $J$ results in a violation of the maintenance location capacity. In this case, based

[1] There is one exception: for the min-cut cut generation method, the generalization to multiple maintenance teams is not straightforward. A note on this is made at the end of Section 6.5.4.

on the set $J$, cuts can be generated according to one of the procedures that are described in Section 6.5.

A cut indicates a combination of jobs that results in an infeasible solution of the **APP**. Let $C(J)$ be the set of cuts based on set $J$. For any cut $A \in C(J)$ it holds that $A \subseteq J$ and **APP**$(A) = \infty$.

Each cut can be translated into a constraint of the **MSLCP**. Consider a cut $A$. Since $A \subseteq J$, every element $a \in A$ signifies a maintenance job. This maintenance job has the following characteristics: it corresponds to MO $a_j$ for rolling stock unit $a_i$ and it contains the set of maintenance activities $a_K$ (where $a_K \subseteq K$). To include this cut $A$, the constraint in Equation (6.1) needs to be added to prevent the combination of jobs in the cut to show up in a next iteration of the **CMSLCP**.

$$\sum_{a \in A} \sum_{k \in a_K} (1 - x_{a_i a_j k}) \geq 1 \tag{6.1}$$

Multiple cuts, for example the set of cuts $C(J)$, can be added by adding the constraint from Equation (6.1) to the **MSLCP** for every cut $A \in C(J)$.

Let $k$ be an index that tracks the current iteration. Let $C_k^*$ be the set of cuts generated up to and including the $k$th iteration. Let $C_0^* = \varnothing$. Let $\ell_0$ be the start time of the algorithm. Let $\ell$ be a parameter restricting the total computation time until the process terminates (if no optimal solution is found earlier).

Pseudo-code for the iterative procedure of the **CMSCLP** is given in Algorithm 1.

**Algorithm 1 CMSLCP** iterative approach

**function** CMSLCP($\ell$)
    $C_0^* \leftarrow \varnothing$
    $\ell_0 \leftarrow$ current time
    $k \leftarrow 1$
    **while** current time $- \ell_0 < \ell$ **do**
        compute **MSLCP** solution $x^k$, subject to cuts in $C_{k-1}^*$
        $C_k^* \leftarrow C_{k-1}^*$
        **for** $s \in S$ **do**
            **if APP**$(J_{x^k}(s)) = \infty$ **then**
                $C_k^* \leftarrow C_k^* \cup C(J_{x^k}(s))$
            **end if**
        **end for**
        **if** $|C_{k-1}^*| = |C_k^*|$ **then**
            **return** $x^k$ as the optimal **MSLCP** solution
        **end if**
        $k \leftarrow k + 1$
    **end while**
    **return** $x^k$ as the best found sub-optimal **MSLCP** solution
**end function**

The code starts by initializing $C_0^*$, $\ell_0$ and $k$, after which the iterative loop starts. This loop first computes a solution to the **MSLCP** subject to all cuts generated so far. Then, it starts to generate new cuts, first introducing the set $C_k^*$ that will contain these cuts (initially equal to the previous set of cuts). For all maintenance shifts $s$, it is then computed whether a feasible planning could be made for this particular shift. If that is the case, no cuts have to be generated, but if this on the other hand is not the case, cuts are added to the set $C_k^*$. These cuts are produced using any cut generation process (see also Section 6.5 for the cut generation methods used in the current research). After the evaluation of the capacity of all shifts, it is checked whether the previous set of cuts is equal in size as the present set of cuts: if this is the case, no new cuts were necessary meaning that the solution is feasible, and the process terminates. Else, the loop is run again to generate a new solution of the **MSLCP** and potentially new cuts. The process terminates if either an optimal **MSLCP** solution is found, satisfying all constraints, or if the user-defined maximum running time is exceeded.

In Appendix C.1, the iterative procedure of the **CMSLCP** is applied on a very simple toy instance for a maintenance shift of which the capacity is initially violated. It is shown that the **CMSLCP** is indeed able to find a feasible maintenance schedule.

## 6.5 Cuts generation

If a solution to the **MSLCP** is found that violates the maintenance location capacity constraints, cuts are added to the **MSCLP** in order to constrain the solution space and prevent such a solution from showing up again. The cut generation process contributes to the speed with which the **CMSLCP** model is able to find an **MSLCP** solution that satisfies the capacity constraints. The current section presents four alternative processes to generate cuts.

The generation of cuts is important for the efficiency of the **CMSLCP**. It is desirable to find cuts that are as general as possible. In other words, it is desirable to find a cut that reduces the solution space of the **MSLCP** to the largest possible extent. For example, consider a situation wit 3 maintenance jobs: $A$, $B$ and $C$. Suppose that the set of maintenance jobs $\{A, B, C\}$ results in an infeasible solution. This combination of jobs can be added as a cut to the **MSLCP** to prevent this combination of jobs from showing up in future iterations. However, now suppose that actually $A$ and $B$ are overlapping maintenance jobs, such that the occurrence of these two jobs together results in an infeasible **APP** solution. Then not only the combination of jobs $\{A, B, C\}$ results in an infeasible **APP** solution, but in fact any combination of jobs including jobs A and B. A stronger cut, therefore, would be the cut with only the combination of jobs $A$ and $B$, since this not only prevents the combination of $A$, $B$ and $C$ to show up in future solutions, but actually any combination of jobs including $A$ and $B$. The cut generation process is therefore an important factor to consider in the light of the efficiency of the solution approach.

The remainder of this section is organized as follows: first, a simple, but naive method to generate cuts is proposed. Then, a second method is devised that generates cuts in a heuristic manner, and a third method is designed that improves upon the second in terms of running time. Lastly, a more complicated method is developed that tries to find the minimal cut necessary by utilizing the problem structure.

### 6.5.1 Naive cut generation

Let $J$ be a set of jobs for that results in a capacity violation, i.e. **APP**$(J) = \infty$. Let $C(J)$ be the set of cuts generated for this set of jobs.

Since $J$ results in an infeasible solution to the **APP**, it is known that the combination of jobs in $J$ cannot occur together. In the naive cut generation method, the set of $J$ itself is added as a cut. Hence, $C(J) = \{J\}$. Note that this cut is very specific, raising the expectation that many iterations are necessary to converge to a solution of the **CMSCLP** that does not violate the capacity constraints.

### 6.5.2 Basic Heuristic cut generation

Assume that a given set of jobs $J$ violates maintenance location capacity constraints. In the naive cut generation method, all these jobs are added as a cut. However, there may be sets of jobs $J^* \subset J$ that also result in an infeasible solution. The heuristic cut generation method aims to find a set $J^*$ in such a way that $J^*$ is 'just infeasible': this means that there is at least one job $j \in J^*$ such that **APP**$(J^* \setminus \{j\}) < \infty$.

In an attempt to generate smaller cuts, the *Basic Heuristic* cut generation method is proposed. This method starts with an empty set $\tilde{J}$ and then moves random jobs iteratively from $J$ to $\tilde{J}$. It checks whether the current set of jobs $\tilde{J}$ results in a feasible solution of the **APP**. If it does, the current set $\tilde{J}$ is not yet an appropriate cut since the combination of jobs currently in $\tilde{J}$ is not infeasible: hence, another job is added in a new iteration. If, on the other hand, it does not, then the current set of jobs is added as a cut to the **MSLCP**.

Pseudo-code for this method is presented in Algorithm 2.

**Algorithm 2** Basic Heuristic cut generation

```
function HEURISTIC CUT GENERATION(J)
    J̃ ← ∅
    while APP(J̃) < ∞ do
        pick random j ∈ J
        J ← J \ j
        J̃ ← J̃ ∪ j
    end while
end function
```

The proposed method is guaranteed to terminate since at some point, all jobs from $J$ are moved to $\tilde{J}$, meaning that the contents of $\tilde{J}$ are equal to the initial contents of $J$. For this set, it is already known that **APP**$(J) = \infty$ since this was required at the start.

The heuristic cut generation method can be run multiple times to generate multiple cuts. In general, these cuts are not identical due to the fact that the choice on which job $j \in J$ to move from $J$ to $\tilde{J}$ is random.

### 6.5.3 Binary Search Heuristic cut generation

The Basic Heuristic cut generation method aims to find sets of jobs that are 'just infeasible' by iteratively and randomly adding jobs until an infeasible solution is found. The Binary Search Heuristic cut generation method uses the same idea, but improves upon the efficiency of the former by applying a method that is inspired by the principle of binary search (see for example Cormen et al. (2009, p.799)).

Recall that it is known that $J$ results in an infeasible solution. The goal is to find a subset $J^* \subset J$ that also results in an infeasible solution but is of smaller cardinality than $J$. To this end, let $A$ be an initially empty set such that at any moment in the procedure, the jobs in $A$ result in a feasible solution, i.e. **APP**$(A) < \infty$. Let $B$ be a set of candidate jobs that, when added to the jobs in $A$, at any moment in the procedure results in an infeasible solution: **APP**$(A \cup B) = \infty$. The algorithm repeatedly splits $B$ into two halves, a left half $B_L$ and a right half $B_R$, and it computes **APP**$(A \cup B_L)$. If this results in an infeasible solution, i.e. **APP**$(A \cup B_L) = \infty$, then the set $B_R$ is discarded. In the subsequent iteration of the algorithm the set B of candidate jobs is reduced to $B_L$. If this results in a feasible solution, i.e. **APP**$(A \cup B_L) < \infty$, some jobs from $B_R$ still need to be added to achieve a 'just infeasible' solution. In this case, the jobs in $B_L$ are all included in the set $A$, and the remaining candidate jobs $B$ to decide on are the jobs $B_R$."

The algorithm terminates when $|B| = 1$, meaning the set $A \cup B$ is just infeasible. This set can be added as a cut. It often has smaller cardinality than the set $J$ and therefore results in more effective cuts than in the naive method.

The following loop invariants hold (i.e. those expressions are true at the start and end of each iteration):

- **APP**$(A) < \infty$, meaning that the set of jobs in $A$ is feasible
- **APP**$(A \cup B) = \infty$, meaning that when the set of jobs in $B$ is added to the set of $A$, the resulting set of jobs is infeasible.

Pseudo code for the described procedure is given in Algorithm 3.

**Algorithm 3** Binary Search Heuristic cut generation

```
function HEURISTIC CUT GENERATION(J)
    A ← ∅
    B ← J
    while |B| > 1 do
        B_L ← ∅
        h ← ⌈(1/2)|B|⌉
        for i ← 1 to h do
            pick random j ∈ B
            B_L ← B_L ∪ {j}
            B ← B \ {j}
        end for
        B_R ← B
        if APP(A ∪ B_L) = ∞ then
            B ← B_L
        else
            A ← A ∪ B_L
            B ← B_R
        end if
    end while
    return A ∪ B
end function
```

### 6.5.4 Min-cut cut generation

In order to find more efficient cuts, the current section designs a method that aims to find cuts with a small amount of jobs, by making use of the specific structure of the problem.

To this end, the *Relaxed Activity Planning Problem* (**RAPP**) is defined. The **RAPP** is a relaxation of the **APP** in two ways. First, the **RAPP** discretizes the planning horizon to a set of *instants*, which are integer minutes, meaning that jobs can only start and end on integer minutes and job durations should be specified as integers. In the practical context of the railway industry, this is not expected to be problematic since rolling stock units are usually planned per minute. Second, the **RAPP** allows for preemption of jobs. This means that the work on a job does not need to be performed uninterruptedly, and, in case of more than one maintenance team, can be performed by multiple maintenance teams. Although this assumption may result in an activity planning that is not realistic, this assumption is useful as it supports the definition of an efficient **RAPP**, which, in turn, supports the discovery of efficient cuts.

The benefit of the definition of the **RAPP** lies in the fact that, although a feasible solution to the **RAPP** need not be a feasible solution of the **APP** (since the former is a relaxation of the latter), any infeasible solution to the **RAPP** is also an infeasible solution to the **APP**. As a result, infeasible solutions found for the **RAPP** can be used as cuts in the **APP**. Note that this also means that the **RAPP** can only be used if it is infeasible: else, the naive cut generation method is resorted to instead to generate cuts (see Section 6.5.1)[2].

In the remainder of the current section, a model for the **RAPP** is provided for the case with one maintenance team. Then it is discussed how this model can be used to generate cuts for the **MSLCP**. Thereafter, the steps in generating cuts for the **MSLCP** using the **RAPP** is summarized. At the end, a small comment is made on the use of the **RAPP** for situations with more than one maintenance team.

### RAPP definition

The **RAPP** attempts to assign jobs to as many distinct instants as its duration. This problem can be viewed as a variant of the bipartite matching problem (Cormen et al., 2009, p. 732), where jobs need to be matched to instants, with this difference that jobs in the current problem usually need to be matched to multiple instants instead of only one. The bipartite matching problem is often modeled as a maximum flow problem (Ford and Fulkerson, 1956), for which efficient solution algorithms exist (Cormen et al., 2009, p. 732-735). Following this approach, the current research defines the **RAPP** as a maximum flow problem.

Let $J$ be the set of jobs, and let $r_j, t_j, v_j$ be the release time, deadline time and duration for job $j \in J$, respectively, defined in minutes. It is assumed that the duration $v_j$ is integer. Let $P_j$ be the set of instants at which job $j$ is available. This comprises all minutes between $r_j$ and $t_j$ and can be expressed as follows: $P_j = \{x \in \mathbb{N} : \lfloor r_j \rfloor \leq x \leq \lceil t_j \rceil\}$. As mentioned above, note that this aspect of the **RAPP** does not represent a relaxation in the usual context of the railway industry, where jobs are planned per minute and hence $r_j = \lfloor r_j \rfloor$ and $t_j = \lceil t_j \rceil$. Let $P$ be the set of all time instants at which at least one job is available, $P \cup_{j \in J} P_j$.

**Maximum flow graph** The problem is formulated as a maximum flow problem. Define a source $s$ and a sink $t$ and let $E_G$ be a set of directed edges with capacity $c_e$ for edge $e \in E_G$. Let $G = (N_G, E_G)$ be a directed flow graph, where its set of nodes $N_G$ is defined by $N_G = \{s \cup J \cup P \cup t\}$ and its set of directed edges $E_G$ is constructed as follows:

- A directed edge $e \in E_G$ from node $s$ to node $j$ for all $j \in J$ with capacity $c_e = v_j$
- A directed edge $e \in E_G$ from node $j$ to $p$ for all $j \in J$ and $p \in P_j$, with unit capacity $c_e = 1$. This implies that, for each job, there is a directed edge to each instant at which it is available.
- A directed edge $e \in E_G$ from $p$ to $t$ for all $p \in P$, with unit capacity $c_e = 1$.

Determine the maximum flow through the flow graph $G$ from the source $s$ to the sink $t$ and denote the resulting flow through each edge $e \in E_G$ by $f_e$. The **RAPP** is considered to be feasible if and only if the maximum flow equals the sum of all durations, or, equivalently, equals the

[2]In fact, when the **RAPP** is feasible, one need not necessarily use the naive cut generation method instead. In fact, any other cut generation method can be resorted to.

sum of all capacities on edges departing from $s$, i.e. if and only if

$$\sum_{e \in E_G} f_e = \sum_{j \in J} v_j = \sum_{e \in \{(s,v) \in E_G : v \in J\}} c_e. \tag{6.2}$$

The satisfaction of the aforementioned condition(s) represents the fact that all jobs have been completely scheduled.

To understand the workings of the **RAPP**, an example is presented where one maintenance team has to perform four jobs: $J = \{j_1, j_2, j_3, j_4\}$. Jobs $j_1$ and $j_2$ can both be performed at instants $p_1$ and $p_2$ (i.e. $P_1 = P_2 = \{j_1, j_2\}$ and jobs $j_3$ and $j_4$ can be performed at instants $p_3$ and $p_4$ (i.e. $P_3 = P_4 = \{j_3, j_4\}$). As a result, the set of all instants $P = \{p_1, p_2, p_3, p_4\}$.

If all jobs have a duration of one instant, a feasible solution can evidently be achieved, for example by performing $j_1$ during $p_1$, $j_2$ during $p_2$, $j_3$ during $p_3$ and $j_4$ during $p_4$. If, however, the jobs have a duration of two instants, a feasible solution cannot be achieved: for example, $j_1$ and $j_2$ both need to make use of both $p_1$ and $p_2$, which is not possible.

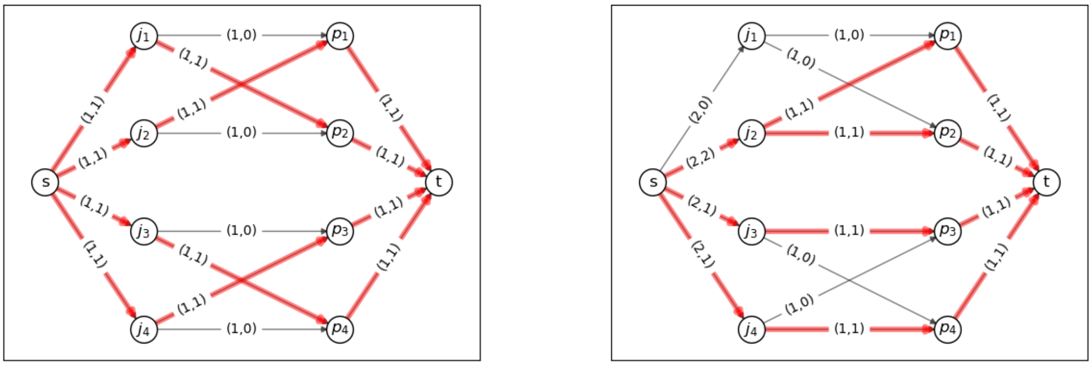



*Figure 6.4: Flow graph $G$ corresponding to the **RAPP** model, for a situation with job durations equal to 1 (left) and equal to 2 (right), the former representing a feasible **RAPP** solution and the latter representing an infeasible **RAPP** solution. Edges $e$ are annotated $(c_e, f_e)$: the first index represents the edge capacity and the second index represents the assigned edge flow. Red-colored edges represent edges through which a strictly positive flow is assigned.*

Figure 6.4 pictures the flow graph $G$ associated to this set-up, and the associated assigned flow, in the situation with job durations of 1 instant (left) and in the situation with job durations of 2 instants (right). In the first situation, the maximum flow is 4 which is equal to the sum of all durations, meaning that by Equation (6.2) the **RAPP** is feasible. The found solution assigns $j_1$ to $p_2$, $j_2$ to $p_1$, $j_3$ to $p_4$ and $j_4$ to $p_3$. In the second situation, however, the maximum flow is also 4, whereas the sum of all job durations is 8, meaning that by Equation (6.2) the **RAPP** is not feasible. The remainder of the current section discusses how this infeasible solution can be used to generate cuts for the **MSLCP**.

### Use RAPP for cut generation

If, given an **MSLCP** solution, the **RAPP** results in an infeasible solution, then the **RAPP** can be used to determine a set of cuts for the **MSLCP**.

To this end, the concept of *minimum cuts* from graph theory is used. The minimum cut is equal to the maximum flow, and gives information about the edges that form a bottleneck in the current graph (Taha, 2011, p. 269). It will become clear that the bottleneck relates to jobs that cannot be performed together; hence, from the minimum cut (in the flow network), a cut to the **MSLCP** (in the **CMSLCP** framework) can be deduced. (Be aware that the term 'cut' can relate both to the cut in the flow network, indicating a bottleneck, as to the cut in the **CMSLCP** context, constraining the solution of the **MSLCP**.)

**Residual graph** To determine the minimum cut, the concept of *residual graph* is used (Cormen et al., 2009, p. 716). It offers information on how the flow between edges can be changed and represents the amount of possible additional flow through each edge. It may also contain so-called *reverse edges*, that represent the possibility of canceling already assigned flow.

To formally define the concept of the residual graph, let $R$ be a directed graph with the same nodes as $G$ and let its set of edges be denoted by $E_R$, that is, $R = (N_G, E_R)$. Then, the set of edges $E_R$ is constructed as follows. For every edge $e \equiv (u, v) \in E_G$:

- there is an edge $e' \equiv (u, v) \in E_R$ with capacity $c_{e'} = c_e - f_e$ if and only if $c_e - f_e > 0$; and
- there is an edge $e'' \equiv (v, u) \in E_R$ with capacity $c_{e''} = f_e$ if and only if $f_e > 0$.

The nodes that are reachable from $s$ comprise the minimum cut, and constitute together the bottleneck.

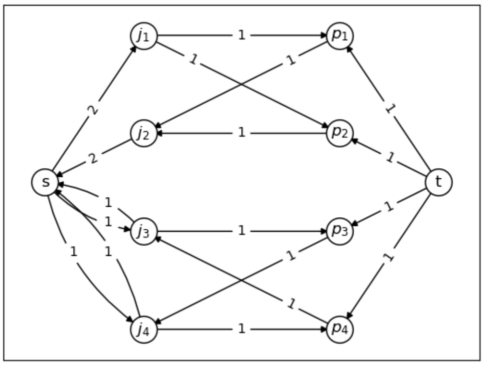



*Figure 6.5: Residual graph $R$ corresponding to the infeasible solution from Figure 6.4. Each directed edge represents the residual capacity between two nodes, if positive.*

To understand the meaning of the residual graph, return to the earlier example that resulted in the infeasible **RAPP** solution pictured on the right side of Figure 6.4. Figure 6.5 displays the residual graph $R$ corresponding to this flow graph $G$. Take, for instance the positive residual capacity of 2 from $s$ to $j_1$: this signifies that an additional flow can be assigned from $s$ to $j_1$ (corresponding to the situation in which $j_1$ is scheduled). However, in this case, the flow must continue to $p_1$ and $p_2$ (meaning that $j_1$ is scheduled during $p_1$ and $p_2$). This can only be achieved if already assigned flow to $p_1$ and $p_2$ flows back to $j_2$ (signifying that $j_2$, which was formerly scheduled at $p_1$ and $p_2$, is not scheduled anymore) and from there flow further back to the source $s$.

The fact that there apparently exists a path from $s$ via $j_1$, $p_1$ and $j_2$ back to $s$ is an important observation: it signifies that $j_1$ and $j_2$ are conflicting. This, in turn, means that $j_1$ and $j_2$ cannot be scheduled together and can be added as a cut. In fact, all jobs on every path starting from $s$ and returning to $s$ constitute an infeasible combination of jobs.

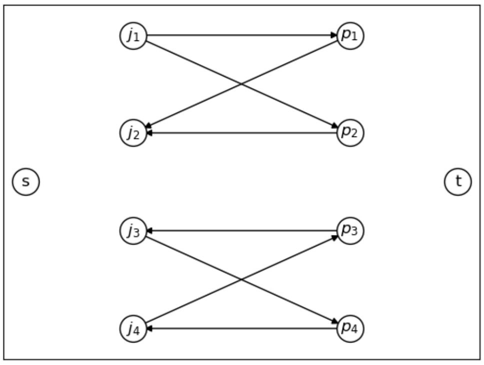



*Figure 6.6: Reachable Components graph $H$, separating the various reachable components that are reachable from $s$.*

**Reachable Components graph** This idea can be formalized by introducing the *Reachable Components graph* $H$. The aim of this graph is to separate components that define different combinations of jobs, each of which cannot occur together (i.e. result in an infeasible solution of the **RAPP**). Let $H$ be a directed graph and let it have the same nodes as $G$ and with the set of edges $E_H$, i.e. $H = (N_G, E_H)$. Let $E_H$ contain all edges in $R$ that are not connected to the source $s$ or sink $t$, that is, $E_H = \{(u,v) \in R : u \notin \{s,t\}, v \notin \{s,t\}\}$. Let $D(F,n)$ be the set of all nodes reachable in some graph $F$ starting from some node $n$ (also called the *descendants* of $n$ in $F$). This set of reachable nodes can be obtained efficiently by the application of a depth-first search (Cormen et al., 2009, p.603-606).

From this, finally, a set of cuts for the **MSLCP** can be determined. Note that all separate sets of reachable nodes can be obtained by starting at some job $j \in J$ that is reachable from $s$ in $R$ and obtaining all jobs among its descendants. In other words, for all $j \in J : (s,j) \in R$ the set $C_j = \{j \cup (D(H,j) \cap J)\}$ comprises a set of jobs that cannot occur together. These jobs result in an infeasible **RAPP** solution and, as a consequence, in an infeasible **APP** solution; hence, they can be added as a cut for the **MSLCP**.

To demonstrate the process of the determination of these cuts, return once again to the previous example. Figure 6.6 presents the graph $H$ with two different components. In $R$, the nodes $j_1, j_3$ and $j_4$ are reachable from $s$. Hence, the cuts generated in this way are $\{j_1, j_2\}, \{j_3, j_4\} and \{j_4, j_3\}$. This shows that $j_1$ and $j_2$ cannot occur together, and similarly that $j_3$ and $j_4$ cannot occur together, which is indeed correct.

**Cut set post-processing** All cuts according to the above described method can be added to the **MSLCP**, but some of these may be superfluous. First, the same cuts may be generated more than once (as is the case in the example above: the combination of jobs containing $j_3$ and $j_4$ is generated twice). Second, some cuts may be generated while a more specific cut is also generated: for example, consider the generation of two cuts, the first with jobs $X, Y$ and $Z$ and the second with jobs $X$ and $Y$. The latter makes the former redundant. To remove redundant cuts, a small procedure is applied that iteratively adds cuts only if it is not a superset of a more efficient cut that was already added. To this end, let $C$ be the set of all cuts generated by the **RAPP** and let $\tilde(C)$ be the set of cuts with all redundant cuts from $C$ removed. Algorithm 4 gives pseudo-code for this procedure.

**Algorithm 4** Remove redundant cuts after min-cut cut genreation

**function** REMOVE REDUNDANT CUTS($C$)
    sort $C$ by the cardinality of all its elements $c \in C$
    $\tilde{C} \leftarrow \varnothing$
    **for** $c \in C$ **do**
        add $\leftarrow$ **true**
        **for** $\tilde{c} \in \tilde{C}$ **do**
            **if** $c \supseteq \tilde{c}$ **then**
                add $\leftarrow$ **false**
            **end if**
        **end for**
        **if** add = **true then**
            $\tilde{C} \leftarrow \tilde{C} \cup \{c\}$
        **end if**
    **end for**
    **return** $\tilde{C}$
**end function**

#### Overview: steps in generating cuts using the RAPP

Summarizing, the steps necessary to generate cuts are as follows.

1. Define the flow graph $G$.
2. Determine the maximum flow in $G$. From this maximum flow, it can be determined whether the **RAPP** is feasible. If the **RAPP** is feasible, it cannot be generate cuts. Generate cuts by the naive cut generation method instead (Section 6.5.1) and add these cuts to the **MSLCP**. Else, continue to the next step.
3. From $G$, determine the residual graph $R$.
4. Based on $R$, define the graph with all reachable components $H$.
5. Based on $H$, determine the set of cuts $C$.
6. Post-process the set of cuts $C$ to remove redundant cuts and store these cuts in the set $\tilde{C} \subseteq C$.
7. Return the set of cuts $\tilde{C}$, which can be added as constraints to the **MSLCP**.

#### RAPP with multiple maintenance teams

The current research only considers the **RAPP** for situations with one maintenance team, although in principle, the described framework of the **RAPP** can also be used for multiple maintenance teams ($\overline{N}$). This is achieved by generating additional instants for other teams, so that each team has its own dedicated instants. It must be noted that the inclusion of multiple maintenance teams is a further relaxation of the **APP**, since in the resulting **RAPP** solution, maintenance activities on one rolling stock unit may be performed by multiple teams. In addition to the fact that maintenance can be distributed freely over time (due to the allowance of preemption), maintenance can then also be distributed freely over maintenance teams. It must be verified that the **RAPP** then does not become a too severe relaxation of the **APP**, resulting in the fact that it becomes feasible very easily. That effect would be undesirable since it would reduce the ability of the **RAPP** to detect infeasible solutions of the **APP**.

| | notation | significance |
|---|---|---|
| **Functions** | **APP**$(J)$ | The optimal objective value of the **APP** model when applied to the set of maintenance jobs $J$ |
| | $C(J)$ | The set of cuts generated for the set of maintenance jobs $J$ |
| | $D(F,n)$ | The set of all descendants of node $n$ in graph $F$ |
| **Sets** | $C^*_k$ | The set of cuts generated up to iteration $k$ |
| | $J_{x^k}(s)$ | The set of maintenance jobs that need to be performed in maintenance shift $s$ |
| | $S$ | The set of all maintenance shifts |
| **Graphs, nodes** | $G = (N_G, E_G)$ | Flow graph describing the **RAPP** |
| | $H = (N_G, E_H)$ | Reachable Components graph after decoupling the sink and the source from $R$ and assigning unit edge weights to all positive edges in $R$ |
| | $R = (N_G, E_R)$ | Residual graph obtained after assigning maximum flow to $G$ |
| | $s$ | Source node in graphs $G$, $R$ and $H$ (or the index used to identify a shift $s \in S$) |
| | $t$ | Sink node in graphs $G$, $R$ and $H$ |
| **Variables** | $f_e \in \mathbb{R}$ | Flow assigned to edge $e$ in a graph after assigning maximum flow to $G$ |
| **Indices** | $e$ | Index used to identify an edge of some graph |
| | $k \in \mathbb{N}$ | Index to keep track of the number of iterations in the **CMSLCP** |
| | $s \in S$ | Index to identify any maintenance shift (or the source node in graph $G$, $H$ or $R$) |
| **Parameters** | $c_e \mathbb{R}$ | Capacity of edge $e$ in a graph |
| | $\ell \in \mathbb{R}$ | The restriction on the maximum running time of the CMSLCP model |
| | $\overline{N} \in \mathbb{N}$ | The number of maintenance teams available. Unless states otherwise, it is equal to 1. |

*Table 6.2: Mathematical notation in the **CMSLCP** model, for reference.*

# Chapter 7

# MSLCP results

This chapter presents the results that were obtained by running the **MSLCP** model on various instances. Its goal is to provide insight in the functioning of the model in various scenarios. It consists of two parts. Section 7.1 outlines the way results are generated by discussing the input data and the scenarios and experiments that are considered. Thereafter, Section 7.2 presents the actual results per experiment.

## 7.1 Experimental design

The current section outlines the way the results are generated. Section 7.1.1 first gives a detailed description of the data that was used for the analysis. Section 7.1.2 reports the set-up of the several batches of scenarios that were run on this data. Then, using these batches, several experiments are investigated, for which the design and purpose is described in Section 7.1.3. Section 7.1.4 gives the most important KPIs that can be used to assess the quality of any solution.

### 7.1.1 Data

This report relies on data of planned rolling stock movements on the Dutch railway network (NS, 2018b). The data set contains all rolling stock movements of trips that are operated by the Dutch Railway operator NS. The data is delivered at intervals of approximately eight weeks, which implies that each data set is valid for approximately eight weeks. A data set for such a period is referred to as *Basisdag update* (BDu). In the current research, 10 different BDus have been used. Each of these BDus are individual data sets on which the **MSLCP** model can be run.

Table 7.1 gives an overview of the different input data sets with their validity, the number of rolling stock units, the average number of activities per rolling stock unit per day, and the number of unique locations identified in the data. A location can be a station, a yard or an important point in the infrastructure (such as a crossing). An increasing trend in the number of rolling stock units and a decreasing trend in the average number of activities can be observed. These trends can be explained by the fact hat new rolling stock units have been added to the Dutch network, and a large part of this new rolling stock is not fully operational yet (since it is used for, for instance, training purposes). At the same time, when new rolling stock units replace the previous rolling stock units, the previous rolling stock units are preserved as a back-up for potential failures of the new rolling stock. Hence, the number of rolling stock increases, but the number of activities does not increase accordingly. Further, note that the number of locations is much smaller than the total number of stations in The Netherlands, since the data set only includes locations where rolling stock units end or start their service. Intermediate stops, where no train services originate or terminate, are not listed.

| BDu id | start date | end date | # RSU | # activities | # locations |
|---|---|---|---|---|---|
| 1 | 10-12-2017 | 4-2-2018 | 820 | 10.2 | 112 |
| 2 | 5-2-2018 | 8-4-2018 | 835 | 10.1 | 114 |
| 3 | 9-4-2018 | 9-6-2018 | 834 | 10.2 | 112 |
| 4 | 10-6-2018 | 2-9-2018 | 871 | 9.6 | 113 |
| 5 | 3-9-2018 | 30-9-2018 | 886 | 9.7 | 111 |
| 6 | 1-10-2018 | 8-12-2018 | 886 | 9.8 | 111 |
| 7 | 9-12-2018 | 3-2-2019 | 924 | 8.8 | 108 |
| 8 | 4-2-2019 | 31-3-2019 | 979 | 8.3 | 112 |
| 9 | 1-4-2019 | 8-6-2019 | 991 | 8.3 | 111 |
| 10 | 9-6-2019 | 1-9-2019 | 993 | 8.2 | 115 |

*Table 7.1: BDu data set statistics. For each BDu, the start date and end date (defining the validity of the data), the number of rolling stock units (RSU) considered in the BDu, the average number of activities per day per rolling stock unit, and the number of unique locations used in the data set.*

The number of rolling stock units used for intercity services and sprinter services is displayed in Figures 7.2 and 7.3, respectively. For example, for the BDu of period 10, the total number of rolling stock units in the data set was 993. Of this number, 360 were primarily intercity rolling stock units and 354 were primarily sprinter rolling stock units (although it must be noted that the type DDZ4 is used for both sprinter and intercity services). The other 279 rolling stock units are mainly reserves or old rolling stock units that are not used anymore but are still in the analysis. The experiments in this chapter focus on the rolling stock units of the intercity type, and mainly on those rolling stock units of type VIRM4 or VIRM6.

| | BDu period | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Rolling stock type | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| VIRM 4 | 85 | 91 | 91 | 89 | 92 | 92 | 94 | 94 | 94 | 92 |
| VIRM 6 | 75 | 75 | 70 | 69 | 70 | 70 | 69 | 69 | 69 | 68 |
| ICM 4 | 46 | 46 | 46 | 46 | 46 | 46 | 46 | 47 | 46 | 46 |
| ICM 3 | 86 | 85 | 85 | 85 | 85 | 85 | 86 | 85 | 86 | 86 |
| DDZ 6 | 17 | 16 | 16 | 16 | 17 | 17 | 16 | 16 | 15 | 17 |
| DDZ 4 | 25 | 24 | 24 | 24 | 25 | 23 | 25 | 24 | 24 | 24 |
| SW9-25KV 2+9 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 |
| SW7-25KV 2+7 | | | | | | | 15 | 15 | 15 | 15 |
| Total | 346 | 349 | 344 | 341 | 347 | 345 | 363 | 362 | 361 | 360 |

*Table 7.2: The number of rolling stock units used for intercity services in various BDus. It must be noted that DDZ4 is also used partially for sprinter services.*

Each BDu specifies the rolling stock circulation for all rolling stock units that the BDu includes. Appendix B.1 details how the rolling stock circulation is extracted from a BDu.

### 7.1.2 Scenario batches

Three different groups of scenarios (*scenario batches*) have been run, for which the specific characteristics are described in the current section. In a scenario batch, for each parameter

| Rolling stock type | BDu period 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| SLT 6 | 57 | 59 | 59 | 60 | 60 | 60 | 59 | 59 | 60 | 60 |
| SLT 4 | 63 | 63 | 66 | 67 | 67 | 67 | 64 | 64 | 65 | 65 |
| SNG 4 | 5 | 5 | 5 | 5 | 12 | 12 | 17 | 37 | 37 | 37 |
| SNG 3 | 5 | 5 | 5 | 5 | 10 | 10 | 12 | 32 | 32 | 32 |
| FLIRT FFF 4 | 23 | 23 | 23 | 23 | 23 | 23 | 23 | 23 | 23 | 23 |
| FLIRT FFF 3 | 29 | 29 | 29 | 29 | 29 | 29 | 31 | 31 | 31 | 31 |
| SGMM 3 | 54 | 54 | 53 | 53 | 55 | 49 | 54 | 54 | 54 | 54 |
| SGMM 2 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 |
| DDM1 4DDM | 10 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| DD-AR 3 | 16 | 16 | 16 | 16 | 16 | 16 | 17 | 17 | 16 | 16 |
| Total | 290 | 290 | 292 | 294 | 308 | 302 | 313 | 353 | 354 | 354 |

*Table 7.3: The number of rolling stock units used for sprinter services in various BDus.*

a non-empty set of parameter values is defined, after which the **MSLCP** is run on all combinations of parameter values for the various parameters. Below, first the various parameters that can be used to define scenarios are discussed, then the settings which were constant over all scenario batches are indicated, and then the settings for the various scenario batches are described.

### Parameters

The aspects that can be varied in generating results for the **MSLCP** can be derived from the assumptions listed in Table 3.3 in Section 3.2: all assumptions classified as "input" in Table 3.3 are parameters to the **MSLCP**. An overview of these parameters is given in Table 7.5.

| parameter | id in Tbl 3.3 | user | coded |
|---|---|---|---|
| Rolling stock circulation. | 1 | | |
| *- Various BDu input sets are considered* | | x | |
| *- Various numbers of rolling stock units are considered* | | x | |
| *- Various lengths of the planning horizon (in days) are considered* | | x | |
| Set of nighttime maintenance locations | 2 | | |
| *- Assumed to be equal to set of all locations in the analysis* | | | x |
| Set of potential daytime maintenance locations | 4 | | |
| *- Assumed to be equal to set of all locations in the analysis* | | | x |
| Maximum number of daytime locations that can be opened. | 5 | | |
| *- Various numbers are considered* | | x | |
| Maintenance types | 6 | | |
| *- Various types are considered* | | x | |
| Initial conditions | 9 | | |
| *- Assumed that all rolling stock units are as-good-as-new at the start of the planning horizon (i.e. $b_{ik} = 0$ for all $i \in I, k \in K$)* | | | x |
| The start of the daytime time window and the start of the nighttime time window | 16 | | |
| *- Assumed that the nighttime time window starts at 19.00 (i.e. $\delta^N = 19$)* | | x | |
| *- For the daytime time window, various values are considered.* | | x | |

*Table 7.5: Parameters in the **MSLCP** model. These parameters (with their ids) correspond to those assumptions in Table 3.3 which are classified "input", meaning they are not intrinsic to the model, but can be varied instead.*

These are classified in two categories: user and coded. This classification informs the reader about the current implementation. A parameter is classified 'user' if it can be varied by the model user, so that the model can be run for various choices of the parameter and the effects of it can be investigated. A parameter is classified 'coded' if currently no option is implemented to vary it.

All parameters considered in the current chapter are discussed below.

- *BDu.* Various BDus are considered, which comes down to feeding the algorithm with input sets for different periods.
- *Number of rolling stock units.* From these BDus, an arbitrary number of rolling stock units can be taken (constrained by the maximum number of rolling stock units in the BDu). The implementation is such that the rolling stock units are selected in order of appearance: for example, if 10 rolling stock units are to be selected, the first 10 rolling stock units are selected. Increasing the number of rolling stock units leads to higher running times. This variable is indicated in the tables of this chapter by Greek letter $\tau$.
- *Planning horizon.* Various lengths of planning horizons can be considered. Extending the planning horizon contributes to increasing running times. To retain meaningful interpretation, it should not be chosen longer than the validity of the input data (see Table 7.1). The number of days is indicated in the tables of this chapter by greek letter $\nu$.

- *Maintenance locations.* The set of nighttime maintenance locations is assumed to be equal to the set of all locations in the BDu. This implies that nighttime maintenance can take place everywhere. Similarly, the set of potential daytime maintenance locations is assumed to be equal to the set of all locations in the BDu. This implies that daytime maintenance can potentially take place everywhere.
- *Maximum number of daytime maintenance locations.* The maximum number of daytime locations can be varied. By increasing the number of daytime maintenance locations, the problem is becoming less restrictive. This results at least in an unchanged objective value, but often even in an improvement of the objective value.
- *Maintenance types.* Various definitions for maintenance types, their durations and maximum intervals can be considered.
- *The initial condition of each rolling stock unit.* For each rolling stock unit, the time passed at the start of the time horizon since the last maintenance activity of each type can be specified. In all investigated scenarios, the initial condition of the rolling stock unit at the start of the time horizon is assumed to be as-good-as-new: $b_{ik} = 0$ for all $i \in I, k \in K$.
- *Start of the daytime and nighttime maintenance windows.* The parameters for the start of the daytime time window and the start of the nighttime time window influence which MOs are classified as daytime MOs and which are considered as nighttime MOs. Throughout the entire analysis, the start of the nighttime maintenance window is fixed at 19.00: $\delta^N = 19$.

Strictly speaking, the technical parameter $\varepsilon$ should also be included in the list above, although it does not have a particular interpretation.

The default values for the parameters are listed in Table 7.6. Unless stated otherwise, these parameter settings apply to the scenarios described below.

| parameter | default value |
|---|---|
| BDu id | 10 |
| rolling stock | {all VIRM4, all VIRM6} |
| planning horizon | $\nu = 42$ days |
| maintenance locations daytime | $L^D = L$ (all locations) |
| maintenance locations nighttime | $L^N = L$ (all locations) |
| maximum number of daytime maintenance locations | $L^D_{max} = 20$ |
| maintenance types | $K$ ={Type A, Type B} |
| maintenance duration Type A | $\upsilon_A = 0.5$ hours |
| maintenance duration Type B | $\upsilon_B = 1.0$ hours |
| maintenance interval Type A | $o_A = 24$ hours |
| maintenance interval Type B | $o_B = 48$ hours |
| initial condition | $b_{ik} = 0$ hours (as-good-as-new) |
| start of daytime time window | $\delta^D = 7$ (07.00) |
| start of nighttime time window | $\delta^N = 19$ (19.00) |
| technical parameter | $\varepsilon = 0.001$ |

*Table 7.6: Default values in the scenario batches*

**Scenario batch 1**

The first scenario batch is intended to gain insight in how the number of rolling stock units, the number of days in the planning horizon, the start of the daytime time window and the number of daytime locations influence the model behavior and model solution.

- Number of days: $\{7, 21, 42\}$. This variable is indicated by $\nu$ in the output tables.
- Rolling stock units: {10 VIRM4, 20 VIRM4, all VIRM4, (all VIRM4 and all VIRM6)}. The total number of rolling stock units is indicated by $\tau$ in the output tables.
- Start of daytime time window $\delta^D$: {7, 10}
- Number of daytime locations $L^D_{max}$: {0, 1, 2, 3, 5, 20}

The other values are equal to the default settings. The total number of scenarios in this batch then becomes 144.

**Scenario batch 2**

The second batch can be used to gain insight into the sensitivity of the model to various BDus and the sensitivity of the model to a longer duration of maintenance. For the latter, an extra maintenance duration setting is added where the default maintenance duration is multiplied by a factor 1.5. This means that the parameter settings are varied in the following way:

- Maintenance types:
  - *(Default)* Maintenance type A has maximum interval $o_A = 24$ and duration $v_A = 0.5$, maintenance type B has maximum interval $o_B = 48$ and duration $v_B = 1$.
  - *(Extended maintenance durations)* Maintenance type A has maximum interval $o_A = 24$ and duration $v_A = 0.75$, maintenance type B has maximum interval $o_B = 48$ and duration $v_B = 1.5$.
- BDu id: $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

The other values are equal to the default settings. The total number of scenarios in this batch then becomes 20.

**Scenario batch 3**

In the third scenario batch, the working of the model is tested on the large instance that includes all rolling stock units intended for intercity transportation. This setting is tested for both the small time horizon of 7 days as the more realistic time horizon of 42 days. The maximum number of daytime locations is chosen larger than the default setting since it is expected that, due to the higher problem size, a higher number of daytime locations is necessary to attain a feasible solution.

The parameter settings then become as follows:

- Number of days: $\tau \in \{7, 42\}$
- Rolling stock units: (all VIRM4, all VIRM6, all ICM3, all ICM4, all DDZ4, all DDZ6, all ICD)
- Number of daytime locations $L^D_{max}$: {20, 30}

The other values are equal to the default values. The total number of scenarios in this batch then becomes 4.

### 7.1.3 Experiments

Using the previously discussed scenario batches, various relations are investigated in 8 experiments. These experiments are discussed below.

**Experiment 1: The influence of the maximum number of daytime maintenance locations**

The maximum number of daytime maintenance locations, influences the total amount of activities that can be performed during the day and is hence of key interest in the current research. It is investigated how this parameter influences the amount of work that can be carried out during the day and the costs. For this experiment, scenario batch 1 is used.

**Experiment 2: Location consistency**

The **MSLCP** returns an optimal choice for the locations that need to be opened for daytime maintenance. This optimal choice obviously depends on the input, such as the period under investigation or the length of the time horizon. In reality, the input is often uncertain or may change over time. This experiment investigates whether the choices on the locations to open are consistent over multiple inputs for the **MSLCP**. In practice, this aspect is important, since the choice regarding the locations to open is often made for a longer period of time and not changed instantly when new information becomes available. For this experiment, scenario batch 1 and 2 are used.

**Experiment 3: Hours of activity and associated costs**

The **MSLCP** returns a schedule with maintenance activities. From this schedule, the total hours of activity (both during daytime and nighttime) can be computed. It is useful to investigate the total hours of activities, since this allows to make estimates regarding costs. To this end, a simple cost calculation proposed in Section 7.1.4, is used, according to which the costs for various scenarios are assessed. Moreover, it is of interest to investigate the spread of the hours of activities over various locations. This may offer valuable information about how efficiently the capacity of each location is used. For this experiment, scenario batch 1 and scenario batch 2 are used.

**Experiment 4: Performance of comprehensive scenario**

The former results were produced with a subset of the rolling stock units operating on the Dutch railway network, only considering the rolling stock types VIRM4 and VIRM6. This choice is made to reduce computation times, but is at the same time rather arbitrary. In Experiment 7, all rolling stock units that serve intercity (long-distance) lines are included. It is investigated whether the smaller-scale results carry over to a larger, productive scenario for a realistic case study. For this experiment, scenario batch 3 is used.

Four additional experiments are given in Appendix B.3.

### 7.1.4 KPIs

The following three main KPIs are used in Section 7.2 to assess the quality of various solutions.

- *Hours of activity.* The hours of activity (averaged per day) are computed by summing the durations of all activities and dividing it by the number of days in the planning horizon. It can be computed in total or split out per location where the maintenance activities take place. Unless stated otherwise, the hours of activity are given for the daytime maintenance activities only. After all, the main interest in the current research is how much work can be performed during daytime (as to reduce the capacity pressure on the maintenance locations during nighttime).

- *Day share.* The day share is the percentage of work that is performed during daytime. It is calculated by taking the total hours of activity over the entire planning horizon during daytime and dividing it by the total hours of activity over the entire planning horizon both during daytime and during nighttime.

- *Average costs.* A very simple cost calculation is established to assess the cost associated to any solution. This cost model computes the cost based on the number of opened locations and on the hours of activity. Let $c^{A,D}$ and $c^{A,N}$ be the cost per hour associated to an activity during daytime or during nighttime, respectively. Let $c^{L,D}$ and $c^{L,N}$ be the cost per day associated to opening any location during daytime or during nighttime, respectively. Then, let the total cost $C$ be defined as follows:

$$C = \sum_{i\in I}\sum_{j\in J_i}\sum_{k\in K}\left(x_{ijk}\cdot d_{ij}\cdot v_k\cdot c^{A,D} + x_{ijk}\cdot(1-d_{ij})\cdot v_k\cdot c^{A,N}\right) + \sum_{l\in L}\left(y_l^D\cdot c^{L,D} + y_l^N\cdot c^{L,N}\right) \tag{7.1}$$

  For now, it is assumed that there are no fixed costs for opening a maintenance location, and hence $c^{L,D} = c^{L,N} = 0$. Moreover, it is assumed that $c^{A,D} = €\,30$ and $c^{A,N} = €\,50$. These numbers are fictitious, since actual values for NS could not be published due to confidentiality requirements. However, these numbers should have the order of magnitude of actual hourly personnel costs.

Three batches of scenarios have been run, indicated in Section 7.1.2. Especially for scenario batch 1, it is not always straightforward how to present the results. The reason for this is that scenario batch 1 contains many (144) scenarios for all combinations of $L^D_{max}$, $v$, $\tau$ and $\delta^D$. Subsets of scenarios from scenario batch 1 may sometimes result in comparable results, which justifies taking an average over the values of KPIs of this group to reflect the average behaviour of the KPI of this group. However, sometimes there may be a large variation within groups. For example, the number of hours of activity varies evidently strongly with the number of rolling stock units incorporated in the results. For such a group, i.e. that contains various numbers of rolling stock units, it would not be justifiable to present an average of the hour of activity since the resulting number would have no relevant interpretative worth.

Therefore, on a case-by-case basis it is determined what averages are presented for what subsets of scenarios. It is attempted to report on the choice made in each specific case as accurately as possible. However, to provide some insight in this process, take the construction of a figure concerning the costs as an example. This may help to better understand the choices made in Section 7.2.

1. First, the scenario batch is considered. Take for example scenario batch 1.

2. Then, the subset of this scenario batch used is determined (although also the entire scenario batch may be considered). For example, in some cases only the results for those scenarios for which the start of the daytime window $\delta^D = 7$ may be presented.

3. Next, it needs to be decided for which groups of scenarios results are presented. For example, a graph may contain the relation between the costs and various values of $L^D_{max}$.

4. From the above, it becomes clear that the choices for $\delta^D$ and $L^D_{max}$ are determined. This means that the values presented are averaged over scenarios with various values of planning horizon $\nu$ and number of rolling stock units $\tau$.

## 7.2 Results

In this section, the results for the experiments introduced in Section 7.1.3 are presented. In order to generate these results, the **MSLCP** is implemented using Python and solved using Gurobi.

Section 7.2.1 until Section 7.2.4 present the results for Experiment 1 to Experiment 4. As a reference, the result tables indicating unaggregated results for all scenario batches is given in Appendix B.2.

### 7.2.1 Experiment 1: The influence of the maximum number of daytime maintenance locations

The maximum number of locations for daytime maintenance closely relates to the goal of the **MSLCP**: minimizing the number of nighttime maintenance activities. When more locations can be opened during daytime, supposedly more activities can be performed during nighttime.

Note that it need not be the case that the maximum number of locations for daytime maintenance is always attained: especially when the maximum number of daytime maintenance is relatively large, it occurs often that less locations than permitted are opened for daytime maintenance. However, there is no mechanism in the **MSLCP** to keep the number of opened locations as low as possible. It may be happen that in an optimal solution of the **MSLCP**, some locations are 'opened' without any workload assigned to it (i.e. more locations are opened than necessary). The interpretation of the actual number of opened daytime maintenance locations therefore has limited interpretative value. As a result, the current section does not provide any statistics on the actual number of opened locations for daytime maintenance.

#### Day share

To investigate to what extent the goal of the **MSLCP** is reached, it is relevant to look at the day share. This statistic is presented in Table 7.7. It shows that the (average) day share is increasing with $L^D_{max}$. Averages of the day shares are computed over all scenarios in a group for a specific combination of $L^D_{max}$ and $\delta^D$ (i.e. for different $\nu$ and $\tau$). For example, to compute the value of 42.0% for $L^D_{max} = 20$ and $\delta^D = 7$, the collection day shares for all scenarios for which $L^D_{max} = 20$ and $\delta^D = 7$ are averaged. This collection contains scenarios with different values of the planning horizon $\nu$ and the planning horizon $\tau$. The groups are composed in such a way since the variance between the numbers within these groups seems to be relatively low. This can also be seen in Table B.3. (If, on the other hand, numbers would have been averaged over different values for, for instance, $L^D_{max}$, this would not hold: there is a large variance in these groups, since high values of $L^D_{max}$ relate to high values of the day share. In that case, the presented values would have no particular meaning.)

The fact that the day shares are increasing in $L^D_{max}$ is expected. When more locations are opened for daytime maintenance, more maintenance can be performed during daytime and hence the day share increases. This is particularly interesting since the goal of NS is to move hours of activity from nighttime to daytime. Apparently, when opening 20 locations, up to 42.0 % of the work can be performed during nighttime.

Also the fact that the day shares are higher for $\delta^D = 7$ compared to the day shares for $\delta^D = 10$ is expected. As the daytime time window starts earlier, more maintenance opportunities can be classified as daytime maintenance opportunities. When these maintenance opportunities are then used, they can contribute to a higher day share.

| $L^D_{max}$ | $\delta^D = 7$ | $\delta^D = 10$ |
|---|---|---|
| 0 | 0.0% | 0.0% |
| 1 | 4.2% | 2.8% |
| 2 | 8.9% | 6.4% |
| 3 | 13.3% | 10.2% |
| 5 | 22.3% | 16.4% |
| 20 | 42.0% | 25.8% |

*Table 7.7: Day share for the start of the daytime $\delta^D = 7$ and $\delta^D = 10$. These figures are computed as follows: first, the day share for every scenario from Table B.2 is calculated. These day shares are averaged to compute the figures in the current table.*

**Hours of activity and costs**

In the previous paragraph, it is indicated that the day share increases for an increasing maximum number of daytime maintenance locations $L^D_{max}$. This is a positive sign, since the objective is to reduce nighttime maintenance activities. However, it is necessary to also investigate the total hours of activity. After all, it may be possible that, although the percentage of hours of activity during daytime increases, the total hours of activity increases as well, and this would be undesirable.

The relation of $L^D_{max}$ and $\tau$ with the total hours of activity, both during daytime and during nighttime, are presented in Figure 7.1. It considers a subset from scenario batch 1 where $\delta^D = 7$. It presents for each combination of the number of rolling stock units $\tau$ and the maximum number of daytime maintenance locations $L^D_{max}$. For this subset, the hours of activity is presented. The hours of activity presented are averaged over the scenarios with various values of planning horizon $\nu$ (since the hours of activity does not show much variation for various values of $\nu$, this choice is deemed acceptable).

It shows that indeed, the total number of hours of activity per day increases when the maximum number of daytime maintenance locations increases. This means that, when more possibilities for daytime maintenance arise, this leads to more maintenance in total. From this figure it also becomes clear that this holds for various numbers of rolling stock units, and (evidently) that the number of hours of activity also increases when more rolling stock units are added to the analysis.

However, to investigate whether the final result is still desirable (i.e. whether the benefit of the increasing day share outweighs the disbenefit of the increasing number of hours of activity during daytime and nighttime), one can look at the costs. The costs for various values of $L^D max$ and $\tau$ have been presented in Figure 7.2. These costs have been computed using Equation 7.1. Recall that this cost calculation assumes that the costs for daytime maintenance are lower than the cost for nighttime maintenance, per hour of activity. It shows that total costs are decreasing. Although the total hours of activity increase slightly, the total costs still decrease. This is due to the fact that relatively more maintenance is performed during daytime.

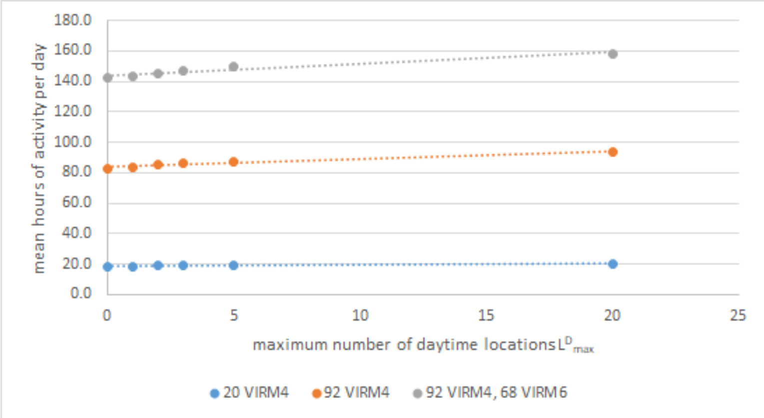



*Figure 7.1: Relation between $L^D_{max}$ and the total hours of activity (both during daytime and during night-time), for various numbers of rolling stock units, for start of the daytime time window $\delta^D = 7$ and with all VIRM4 and VIRM6 rolling stock units included. Numbers are averaged over scenarios with different values of planning horizon $\nu$*

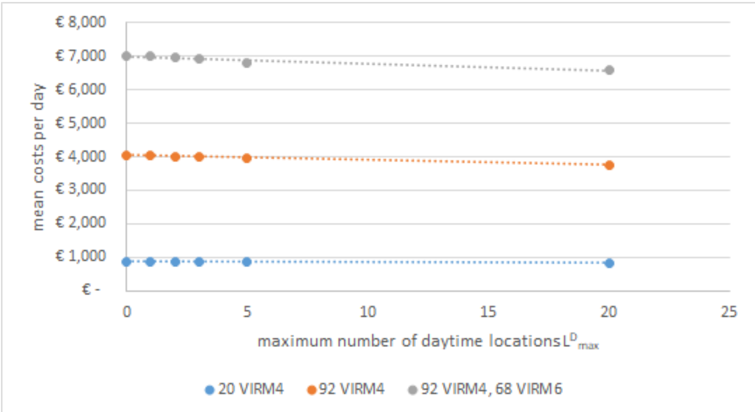



*Figure 7.2: Relation between $L^D_{max}$ and the costs, for various numbers of rolling stock units, for start of the daytime time window $\delta^D = 7$ and with all VIRM4 and VIRM6 rolling stock units included. Numbers are averaged over scenarios with different values of planning horizon $\nu$.*

### 7.2.2 Experiment 2: Location consistency

The current section investigates whether the chosen maintenance locations are similar in the various scenarios. It is desirable that the optimal locations are consistent over various scenarios to solidify any advice on the locations that need to be opened for daytime maintenance. Insight in the location consistency over a specific set of scenarios can be given for any location by reporting the number of times it was chosen throughout the set of scenarios.

#### Consistency for various planning horizons

First, the influence of the planning horizon on the location choice consistency is investigated. For some use cases, the running time of the **MSLCP** model may be too long. Then, it is of interest whether a shorter planning horizon yields similar locations, and hence that running the model with a short time horizon is sufficient to determine good maintenance locations. This may be the case since, for many railway operators, the rolling stock circulation repeats itself on a weekly basis.

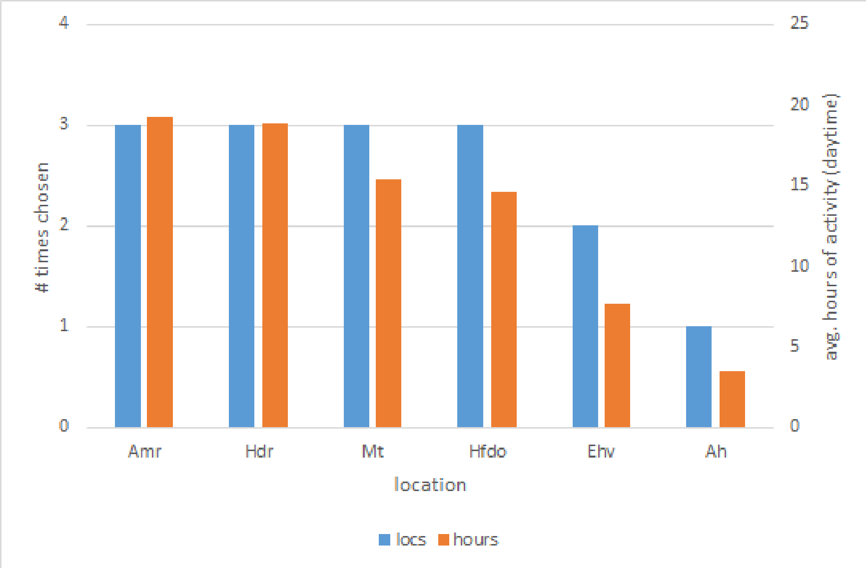



*Figure 7.3: Location consistency for various planning horizons. For each location, it is indicated in how many of the investigated scenarios the location was chosen. Here, the maximum number of locations $L^D_{max} = 5$, the start of the daytime time window $\delta^D = 7$ and all VIRM4 and VIRM6 rolling stock units were included. For these parameter settings, scenario batch 1 contains three scenarios, for $\nu \in \{7, 21, 42\}$.*

In scenario batch 1, every scenario has been run for three values of the planning horizon $\tau$: 7, 21, and 42. To investigate the influence of the planning horizon, it is important to compare scenarios with all other parameter values being equal. To this end, Figure 7.3 considers a subset from Scenario batch 1 for $L^D_{max} = 5$, $\delta^D = 7$, including all rolling stock units of type VIRM4 and VIRM6. This subset of scenarios contains three scenarios (for $\nu \in \{7, 21, 42\}$). The results for this subset are presented in Figure 7.3. The results for a similar subset, but then with $L^D_{max} = 20$, are presented in Figure 7.4. For readability, only the results for these subsets are presented here, but these results can be produced for other subsets as well. Recall that the three scenarios are chosen from scenario batch 1, which contains results for 144 scenarios in total.

Figure 7.3 shows that, when $L^D_{max} = 5$, the locations Amr, Hdr, Hfdo and Mt are chosen in all three scenarios. This serves as evidence that these locations are good choices for daytime maintenance in the given subset, but more importantly it shows that the location choice is robust under various time horizons. Moreover, it shows the average hours of activity during daytime. For completeness, also the hours of activity per location (averaged over the various values for planning horizon $\nu$) have been presented. When $L^D_{max} = 5$, $\delta^D = 7$ and all rolling stock units are included, the hours of activity during daytime is higher than 2 hours per day for these locations that are chosen in all 3 scenarios. In other words, for the locations for which the choice is most consistence, also the highest workload is found.

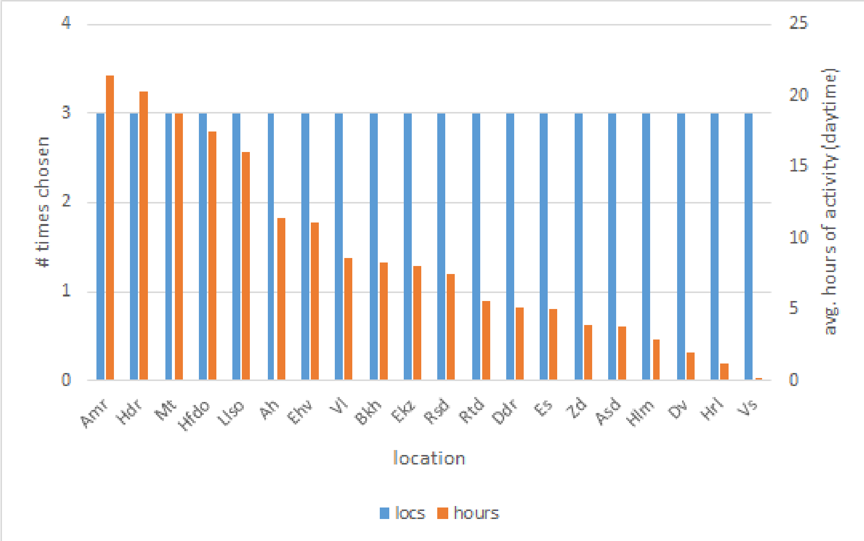



*Figure 7.4: Location consistency for various planning horizons. For each location, it is indicated in how many of the investigated scenarios the location was chosen. Here, the maximum number of locations $L^D_{max} = 20$, the start of the daytime time window $\delta^D = 7$ and all VIRM4 and VIRM6 rolling stock units were included. For these parameter settings, scenario batch 1 contains three scenarios, for $\nu \in \{7, 21, 42\}$.*

Figure 7.4 shows similar results when the maximum number of locations $L^D_{max} = 20$. Note that the location choice is for all three values of planning horizon $\tau$ exactly equal (each time the same 20 locations are chosen).

Note that Amr, Hdr, Mt and Hfdo are still the four best-scoring locations: they are chosen in all three scenarios for different planning horizons $\nu$, and the highest workload assigned to these locations is highest. Hence, also when many locations can be opened (20 in this case), the locations found for a smaller maximum number of daytime maintenance locations (5 in this case, see Figure 7.3) still remain good candidates. Observe that this is not obvious: it may have been true that when more locations can be opened, the location choice changes drastically. This would have meant that the sensitivity of the location choice with respect to the value of $L^D_{max} = 0$, which would not have been desirable in practice since it would have required a very deliberate choice of this parameter.

### Consistency over different scenario types

Figures 7.3 and 7.4 showed the location consistency over three different scenarios with varying time windows. It is, however, also interesting to see how consistent the location choice is throughout various input settings for the start of the daytime time window $\delta^D$ and the number of rolling stock units in the analysis $\tau$. To this end, a subset of scenario batch 2 is considered for which $L^D_{max} = 20$. This group entails 24 scenarios, for all combinations of $\nu$, $\tau$ and $\delta^D$. (Note that it is necessary to create this subset, since the location consistency for scenarios with different values of $L^D_{max}$ cannot be compared in the same graph.) Figure 7.5 displays the location consistency throughout all scenarios for this group.

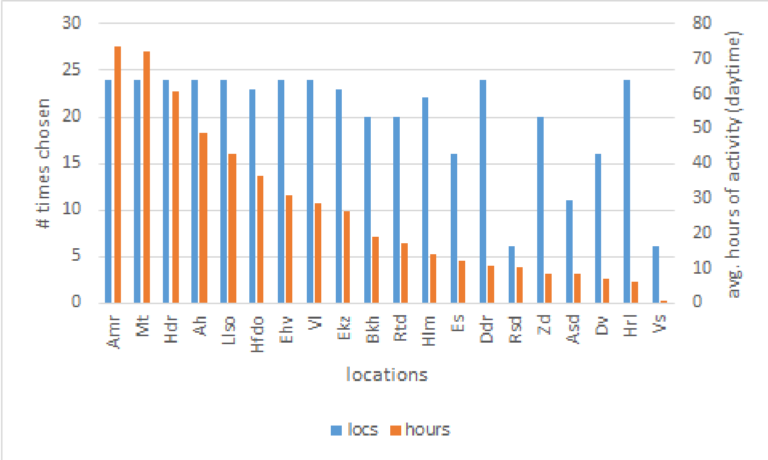



*Figure 7.5: Location consistency over different input parameters. For each location, it is indicated in how many of the investigated scenarios the location was chosen. Also the average workload per location is presented, averaged over all scenarios in which the location is opened. A subset of scenario batch 1 is used, for which the maximum number of locations $L^D_{max} = 20$. This subset contains 24 scenarios, i.e. one for each combinations of $\nu$, $\tau$ and $\delta^D$.*

It appears Amr, Mt, Hdr, Ah, Llso, Ehv, Vl, Ddr and Hrl are chosen in all 24 scenarios. Nonetheless, not on all of these locations a similar amount of work seems to be performed: for example, Hrl is open in all scenarios, but only a very limited amount of work is performed at this location. The results concerning such a location need to be considered with care. Two underlying reasons may contribute to the fact that it is consistently opened and still not much workload is assigned to it. First, it may be the result of the fact that Hrl needs to be opened to find a feasible maintenance schedule. Second, it may be the result of the fact that opening the location Hrl results in the fact that a schedule can be created in which much more daytime maintenance can be performed (potentially also at locations other than Hrl).

However, from Figure 7.5 it can be concluded that the location choice over various inputs for $\nu$, $\tau$ and $\delta^D$ is relatively consistent. Many locations are chosen in more than 20 scenarios. Also, there are some location that are chosen less than 10 times and this is also valuable information: it means that often, a location is *not* a good candidate to open during daytime. It is more difficult to interpret those locations that are chosen in approximately half of the scenarios: for these locations it is uncertain whether they are consistently good location candidates over various input sets.

### Consistency over different BDus

Scenario batch 2 has included multiple BDus (rolling stock circulations of different periods). These scenarios can be compared well: their parameters are exactly equal; the only difference is the time period.

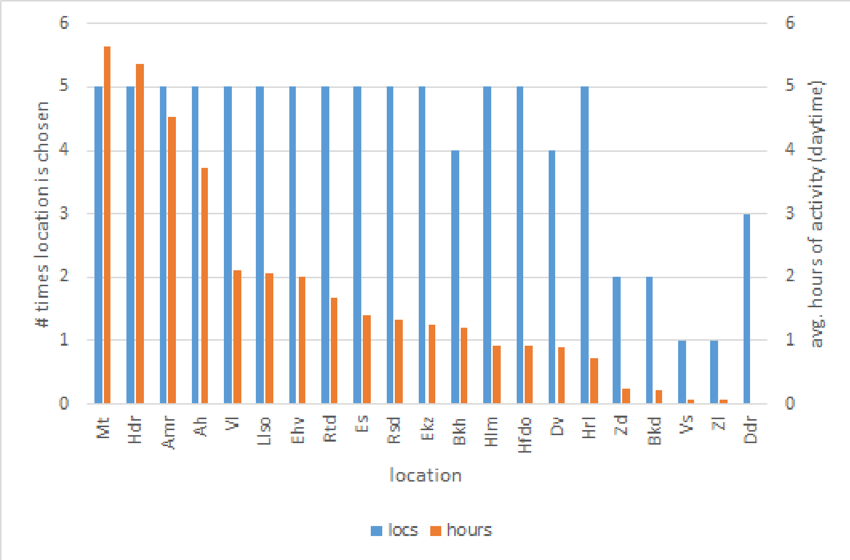



*Figure 7.6: Location consistency over different BDus. For each location, it is indicated in how many of the investigated scenarios the location was chosen. Also the average workload per location is presented, averaged over all scenarios in which the location is opened. Scenario batch 2 is used for the default maintenance types.*

From the 10 scenarios run, 5 resulted in feasible solutions. Figure 7.6 shows that most locations are chosen in all 5 scenarios for which feasible solutions were obtained. This implies that the optimal locations are consistent across different BDus. From a managerial perspective, this is important, since it means that once a location is chosen, it is still optimal to choose this location in a next period. This means that the applicability of the applicability of the results of the **MSLCP** in this case are not only valid for the planning horizon of one instance, but also carry over to a longer time horizon.

There are some occurrences of the situation where a location was chosen in some input sets, and not in other input sets. However, in general this does not concern the locations to which high workloads are assigned. It may therefore not be of high influence to the eventual goal of the **MSLCP** to not open this location. However, it must be noted that to obtain real insights in this, it is important to run the **MSLCP** again with a lower maximum number of maintenance locations, or to run the **MSLCP** with the constraint that the locations that are deemed 'not important' are closed.

### 7.2.3 Experiment 3: Hours of activity and associated costs

This paragraph investigates the influence of various parameters on the total hours of activity, on the share of it that can be performed during the day, and on the resulting associated costs.

#### Influence of start of the daytime time window $\delta^D$

The start of the daytime time window $\delta^D$ is a parameter in the **MSLCP**. Recall that a maintenance activity is classified to be during daytime if and only if its start time is at or after the start of the daytime time window (and before the start of the nighttime time window). Hence, by varying the parameter $\delta^D$, the start of the daytime time window can be varied. (In a similar way, the start of the nighttime time window $\delta^N$ can be varied as well, but this analysis is not reported.) The choice of this parameter is not evident since in practical cases it is not always clear whether a maintenance should be classified daytime or nighttime.

In scenario batch 1, each scenario has been run for $\delta^D = 7$ and $\delta^D = 10$. This means that for each combination of $\tau$, $\nu$ and $L^D_{max}$, one scenario with $\delta^D = 7$ and one with $\delta^D = 10$ is available. Results for the hours of activity and associated costs are given in Table 7.8. These results are grouped by and presented separately for the number of rolling stock units $\tau$, since the total hours of activity per day (both during daytime and nighttime) and the costs show much variation for various numbers of rolling stock units $\tau$ is expected to show much variation. Within these groups, scenarios with different settings for the maximum number of daytime maintenance locations $L^D_{max}$ and planning horizon $\nu$ are found. The total hours of activity and the costs do vary (to some extent) over these scenarios as well. Their variations are addressed in the second part of this section. The variation for hours of activity and costs for these scenarios turns out to be lower than throughout scenarios with different values for $\tau$, and therefore it is deemed acceptable to present averages in Table 7.8 over various values for $L^D_{max}$ and $\nu$. It turns out that the total hours of activity and the costs are approximately equal for the two different choices of $\delta^D$.

The fact that each scenario in scenario batch 1 has been run for $\delta^D = 7$ and for $\delta^D = 10$ allows also for a pairwise comparison between thos pairs of scenarios where only the setting of $\delta^D$ is different (and the settings for $\tau$, $\nu$ and $L^D_{max}$ are equal in both scenarios). Since scenario batch 1 contains in total 144 scenarios, in total 72 pairs can be identified (of which 18 for each choice of $\tau$). For each pair, the *absolute deviation* can be calculated. This absolute deviation can then be averaged over all absolute deviations in a group. Results for this calculation are summarized in Table 7.8. It shows that the mean absolute deviation of the hours of activity and the mean absolute deviation of the costs is very low in respect to the mean hours of activity and the mean costs. For instance, for $\tau = 160$, the mean absolute deviation of the cost over all scenario in this group is only *e* 53.23. This is a small amount compared to the mean costs of 6,454 for $\delta^D = 7$ and *e* 6,506 for $\delta^D = 10$.

Day shares have not been presented because these show much variation over $L^D_{max}$. This relation, however, has already been addressed in Section 7.2.1 and, specifically, in Table 7.7.

| | hours | | | costs | | |
|---|---|---|---|---|---|---|
| $\tau$ | $\delta^D = 7$ | $\delta^D = 10$ | MAD | $\delta^D = 7$ | $\delta^D = 10$ | MAD |
| 10 | 9.2 | 9.0 | 0.12 | € 409 | € 413 | € 4.13 |
| 20 | 18.2 | 18.0 | 0.19 | € 806 | € 815 | € 9.33 |
| 92 | 83.7 | 82.7 | 0.96 | € 3,726 | € 3,760 | € 34.77 |
| 160 | 143.3 | 141.6 | 1.72 | € 6,454 | € 6,506 | € 53.23 |

*Table 7.8: For various values of the number of rolling stock units $\tau$, results are presented for the mean number of hours of activity per day (both during daytime and nighttime) and the means costs per day. For these figures, the mean is presented for $\delta^D = 7$ and $\delta^D = 10$. Then the Mean Absolute Deviation (MAD) is calculated by comparing in a pairwise manner all observations within a group for a specific $\tau$, taking their absolute deviation, and then averaging over all absolute deviations. For instance, for all scenarios in the group for which $\tau = 10$, the difference is taken for the scenario in which $\delta^D = 7$ and $\delta^D = 10$. Each such a group contains 18 observations (all combinations of $L^D_{max}$ and $\nu$) and hence the MAD for this group is computed by calculating the mean of all 18 absolute deviations.*

### Influence of the number of rolling stock units and the time horizon

It is also of interest to investigate the influence of the number of rolling stock units and the number of days in the planning horizon on the number of hours of activity. Note that the following relations are expected: if the number of rolling stock units increases, also the hours of activity per day and the costs per day are expected to increase (since more rolling stock units need to be maintained); if, however, the planning horizon increases, the hours of activity per day and the costs per day are not expected to change (since the KPIs are averaged over the number of days in the planning horizon).

Figure 7.7 shows the relation between the number of rolling stock units and the mean hours of activity per day. Figure 7.8 shows the relation between the number of rolling stock units and the mean costs per day. In both figures, these relations are given for the three various planning horizons $\nu$ considered in scenario batch 1. The figures consider only the scenarios where the start of the daytime window $\delta^D = 7$. The data points in the figures are averages over scenarios with various $L^D_{max}$ (6 in total), since the hours of activity and mean costs per day do not seem to be heavily influenced by $L^D_{max}$ (see Tables B.3 and B.2).

Figures 7.7 and 7.8 indicate that there seems to exist a linear relationship between the number of hours and costs on the one hand, and the number of rolling stock units on the other hand. This is expected: doubling the number of rolling stock units in the analysis is expected to lead to double hours of activity and double costs in return. There also seems to exist a small positive trend (which is potentially not negligible) between the planning horizon on the mean hours of activity and mean costs per day. Whether this trend is significant and, if it is, what the underlying causes for this trend are has not been discovered yet in the current study.

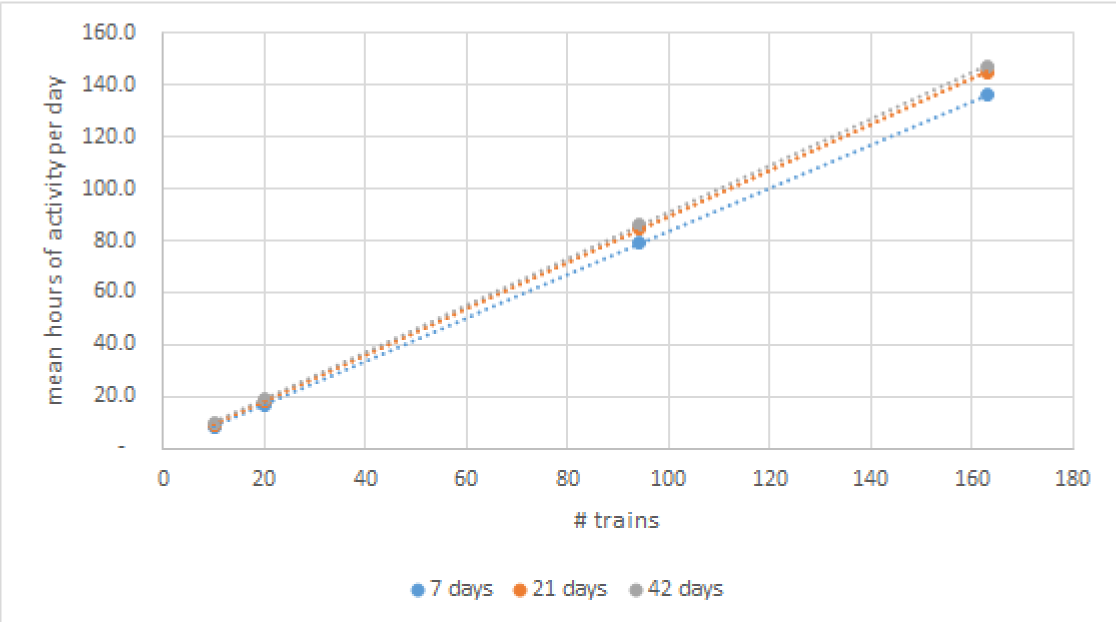



*Figure 7.7: For varying number of rolling stock units $\tau$ and varying time horizon $\nu$, the mean number of hours of activity per day. The start of the daytime time window $\delta^D$ is constant at 7 AM.*

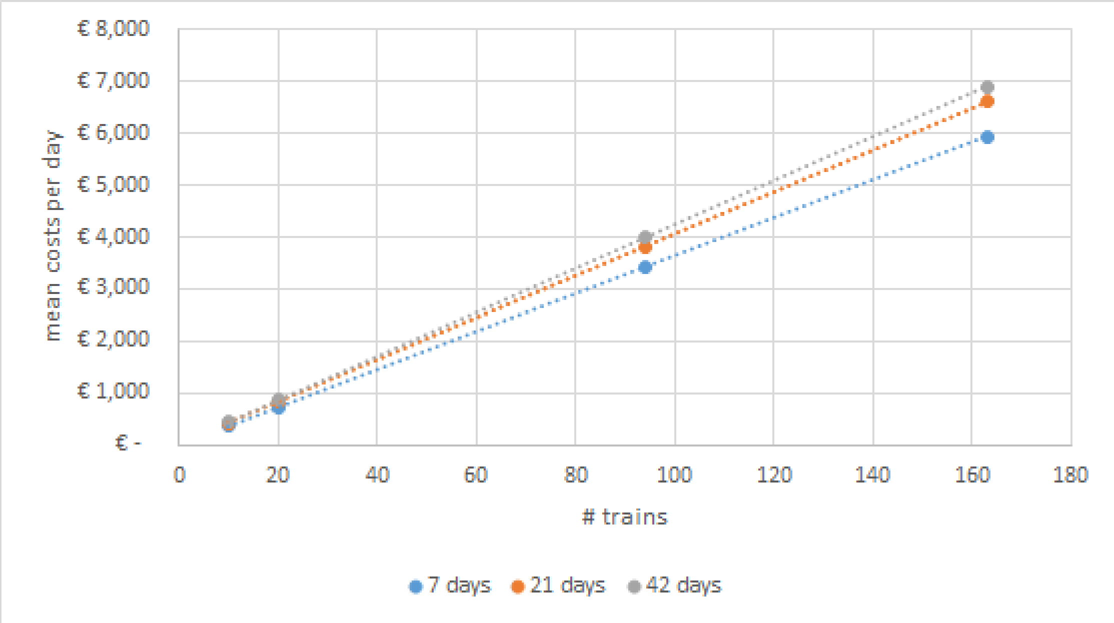



*Figure 7.8: For varying number of rolling stock units $\tau$ and varying time horizon $\nu$, the mean costs per day. The start of the daytime time window $\delta^D$ is constant at 7 AM.*

**Distribution of workload over locations**

It is insightful to investigate how the hours of activity are distributed over various locations. To this end, it proves useful to investigate the various scenarios in batch 2, where the **MSLCP** model is run for various input sets. These scenarios have, apart from the BDu used, the same characteristics and can therefore be compared well.

Consider the scenarios from batch 2 for the default maintenance type. Recall that these scenarios differ only by the input period and can therefore be compared well. Five of these scenarios resulted in a feasible solution.

Figure 7.6 presents for all locations that were opened at least once throughout scenario batch 2, in how many of the five scenarios they were opened. Moreover, it presentes average number of hours of activity on this location averaged over all scenarios in which this location was opened. See Table B.4 for an overview of the day shares and hours of activity in scenario batch 2.

It can be observed that there are a few locations with more than three hours of activity and that these locations are consistently chosen over the multiple scenarios. This is a good sign since it indicates that the choice on whether a location is opened consistently or not correlates with the workload assigned to that location. In other words, if a location is opened consistently throughout multiple scenarios, this usually also means that a relatively high workload is assigned to this location.

### 7.2.4 Experiment 4: Performance of comprehensive scenario

Where scenario batch 1 and 2 consider only rolling stock units of type VIRM4 and VIRM6, scenario batch 3 focuses on a larger instance containing all rolling stock units that are primarily intended for intercity lines on the Dutch railway network. Table 7.9 gives the most important results for the scenarios from the third scenario batch.

| $v$ | $L^D_{max}$ | hrs. activity | day share | costs | running time (s) |
|---|---|---|---|---|---|
| 7 | 10 | 305.9 | 22.6% | € 13,911 | 204 |
| | 20 | 310.6 | 30.9% | € 13,614 | 201 |
| 42 | 10 | 328.8 | 22.2% | € 14,976 | 11,522 |
| | 20 | 334.9 | 30.1% | € 14,729 | 7,885 |

*Table 7.9: The mean hours of activity per day, the day share, the mean costs per day and the running time of the four different scenarios in scenario batch 3, for different settings of the planning horizon $v$ and the maximum number of daytime maintenance locations $L^D_{max}$.*

This figure shows that, for a maximum number of 20 maintenance locations, a day share of over 30 % can be attained. This leads to total costs of approximately 13,000-15,000 euros. The day share appears to be higher for the scenarios with a maximum number of 20 locations opened for daytime maintenance, consistent with the findings in Experiment 1 (7.2.1).

By multiplying the day share by the total hours of activity, the total hours of activity performed during daytime can be computed. Figure 7.9 gives an overview of how this total hours of activity performed during daytime are distributed over the various locations.

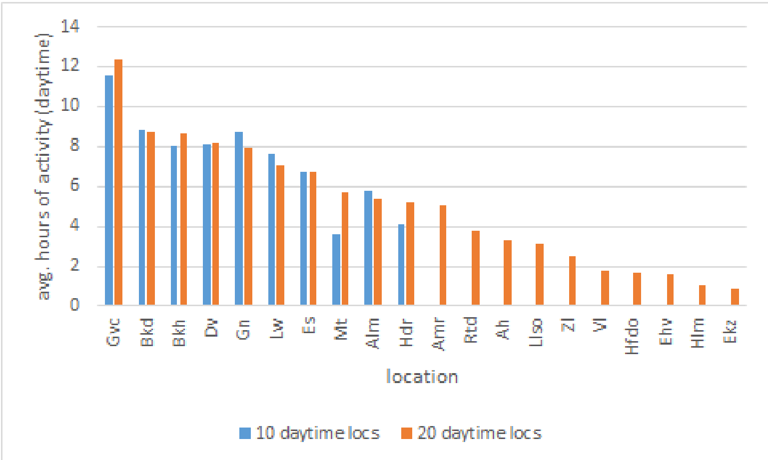



*Figure 7.9: Average number of hours of activity per location performed during daytime, for two scenarios from batch 3 for which $v$ = 42. Results are presented separately for the scenario for which $L^D_{max}$ = 10 and $L^D_{max}$ = 20.*

It shows that, in the situation with a maximum number of locations for daytime maintenance of 10, at least 5 locations have a workload of more than 8 hours per day (Gvc, Bkd, Bkh, Dv and Gn), which can be considered substantial since it is enough to provide work to one maintenance team. Moreover, it shows that the addition of maintenance locations does not seem to reduce the average workload on any of the initial 10 locations. Hence, the initial 10 locations are still good choices, even when daytime maintenance is possible at more locations. The added locations, however, are assigned a much lower workload than the initial 10 locations. It is therefore questionable whether the addition of these locations is worthwhile.

# Chapter 8

# CMSLCP results

The current chapter provides results for the **CMSLCP** by applying it to a realistic problem instance. Its goal is to provide insight in the running time and solution quality of the **CMSLCP** using the various cut generation processes designed in Chapter 6.

Section 8.1 describes the experimental design and Section 8.2 gives the corresponding results.

## 8.1 Experimental design

This section presents the experimental design used to generate results for the **CMSLCP** model. The input data is detailed in Section 8.1.1, defining the problem instance to which the **CMSLCP** is applied. Section 8.1.2 defines two set-ups that are used to generate results, one focusing on the capacity of one maintenance shift only, the other focusing on the capacity of all maintenance shifts. Section 8.1.3 lists the various cut generation processes for which results are obtained in both set-ups. Section 8.1.4 discusses the KPIs used to assess the quality of the results.

### 8.1.1 Data

The problem instance considered in the current chapter uses a rolling stock circulation originating from NS BDu data (comparable to Section 7.1.1). It uses the BDu from period 3 (valid from 9-4-2018, see Table 7.1). From this BDu, it considers the 7 days from 10-4-2018 until 16-4-2018. The first day is cut off since the data set is not guaranteed to give all rolling stock movements of this day (see also the discussion in Section 7.1.1). To reduce computation times, only 4 rolling stock types are considered: ICM4, DDZ4, DDZ6 and DD-AR3. These rolling stock types are chosen in such a way that they result in some maintenance location capacity issues, especially at maintenance location Zl. This comprises a total of 141 rolling stock units (cf. Tables 7.2 and 7.3 in Chapter 7). Of these rolling stock units, 4 were not active in the 7-day period considered (due to, for example, heavy maintenance), reducing the total number of rolling stock units included in the current analysis to 137.

### 8.1.2 Set-ups

Two set-ups are considered to generate results for the **CMSLCP**. Below, first all **CMSLCP** parameters are listed and their values given. Then, the two set-ups are defined.

#### Parameters

Two types of parameters prevail: the first type entails **MSLCP**-specific parameters and are necessary to generate any **MSLCP** solution (see also Section 7.1.2), the second type entails **CMSLCP**-specific parameters.

- **MSLCP**-specific parameters
  - *Planning horizon.* The planning horizon is set to 7 days, equal to the total number of days in the input data.
  - *Maintenance locations.* Like in Chapter 7, both the set of nighttime maintenance locations ($L^N$) and the set of potential daytime maintenance locations ($L^D$) are assumed to be equal to the set of all locations in the BDu.
  - *Maximum number of daytime maintenance locations.* It is assumed that 5 locations can be opened for daytime maintenance at maximum, i.e. $L^D_{max} = 5$.
  - *Maintenance types.* The default maintenance types from Chapter 7 are used, meaning that there are two maintenance types, maintenance type A having a duration of 30 minutes and an interval of 24 hours, maintenance type B having a duration of 60 minutes and an interval of 48 hours.
  - *Initial conditions.* Rolling stock units are assumed to be as-good-as-new at the start of the planning horizon ($b_{ik} = 0$ for all $i \in I, k \in K$).
  - *Technical parameter.* The technical parameter $\varepsilon$ has a value of $\varepsilon = 0.001$.
- **CMSLCP**-specific parameters
  - *Number of maintenance teams.* Throughout the current chapter, it is assumed that at each shift, one maintenance team is available, i.e. $\overline{N} = 1$. This choice is favoured since the current implementation of the min-cut cut generation method is only available for one maintenance team.
  - *Set of shifts.* The set of shifts $S$ is dependent on the set-up used and is discussed below. In both set-ups, only the capacity of daytime maintenance shifts is considered; the capacity of nighttime maintenance shifts is ignored. This choice is reasonable in the light of the gradual introduction of a policy of daytime maintenance in practice, where capacity during daytime at first is limited.
  - *Cut generation method.* Three cut generation methods are available, designed in Chapter 6. This leads to 10 different *cut generation variants*, listed in Section 8.1.3. Results are generated for each cut generation variant.

#### Set-ups

Two set-ups are used to generate results. These two set-ups differ by the set of shifts $S$ for which the **CMSLCP** attempts to prevent violations of the capacity constraints.

First, the *single-shift set-up* focuses on one particular maintenance shift: the daytime maintenance shift in Zl on 11-4-2018. In this case, the set of shifts $S$ contains only one maintenance shift. This maintenance shift appears to be 'hard' to solve, making this single-shift set-up particularly suitable to compare and investigate the performance of various cut generation processes in solving a capacity violation of a specific shift. A maintenance shift that is 'easy' to solve is less suited for this goal since such a shift is often solved quickly by all cut generation methods, making it harder to identify any differences between cut generation methods.

Second, the *all-shifts set-up* focuses on all daytime maintenance shifts. The set of shifts $S$ contains maintenance shifts for all possible combinations of maintenance location and date in the planning horizon. This set-up is primarily useful to provide insight in how quickly the **CMSLCP** is able to reduce the total number of maintenance shifts with a capacity violation. In the all-shifts set-up, three cut generation variants are considered: the naive, the Binary Search Heuristic for 15 cuts, and the min-cut. This set represents all types of cut generation variants and chooses the best-performing one out of all eight heuristic ones (based on the results of the single-shift set-up, see Section 8.2.1).

### 8.1.3 Cut generation variants

In Chapter 6, four cut generation methods have been proposed: the naive method, the Basic Heuristic method, the Binary Search Heuristic method, and the min-cut method. The heuristic methods allow for the generation of multiple different cuts.

The results in the present section are generated using different cut generation methods. The following ten variants are distinguished, referred to as *cut generation variants*.

- Cut generation by the naive cut generation method (one variant)
- Cut generation by the Basic Heuristic cut generation method, for 1, 2, 5 and 15 cuts (four variants)
- Cut generation by the Binary Search Heuristic cut generation method, for 1, 2, 5 and 15 cuts (four variants)
- Cut generation by the min-cut cut generation method (one variant)

### 8.1.4 KPIs

The following KPIs are used to assess the quality of the **CMSLCP** model in the single-shift set-up and in the all-shifts set-up.

- Single-shift set-up
  - *Convergence.* The **CMSLCP** iteratively adds constraints to the **MSLCP**. Therefore, in the **CMSLCP**, the objective value of the initial **MSLCP** converges to the objective value of a solution of the **MSLCP** that satisfies all capacity constraints. The convergence is an important measure of the quality of the **CMSLCP** model: the quicker it converges, the more useful it is in practical contexts. It is graphically displayed by showing the course of the current **MSLCP** objective value as a function of the number of iterations and as a function of the total time elapsed.
  - *Computation time.* The computation time per iteration can be separated in the computation time for the three main sub processes: solving the **MSLCP** subject to all previously generated cuts, solving the **APP** to identify capacity violations, and generating cuts. Note that the second of these is performed for each maintenance shift and note that the third of these is performed for each maintenance shift for which the capacity is exceeded.
  - *Cut efficiency.* A cut is a combination of jobs that result in a capacity violation. It is added to the **MSLCP** to prevent this combination from showing up in a next iteration. The lower the number of jobs in a cut, the more 'general' it is and therefore the more efficient. The average number of jobs per cut is therefore reported as a measure of cut efficiency.

- All-shifts set-up
  - *Number of shifts with capacity violation.* In the all-shifts set-up, the capacity of multiple shifts is addressed. The number of shifts with a capacity violation decreases in the course of the **CMSLCP**. It is presented graphically as a function of the number of iterations and as a function of the total time elapsed.
  - *Convergence.* The convergence is reported as in the single-shift set-up (see above).

It can be said that the convergence as a function of elapsed time is the most important KPI, since it represents how quickly the **CMSLCP** is able to find an optimal solution. Note that this is determined by two aspects. First, it is determined by the time required per iteration. Second, it is determined by the total number of iterations, which in turn is a consequence of the efficiency of the cuts in each iteration. Therefore, in the single-shift set-up, the second and the third KPIs (computation time and cut efficiency, respectively), can be considered to be explanatory for the first KPI (convergence).

## 8.2 Results

The current section generates the KPIs (discussed in Section 8.1.4) for the single-shift set-up in Section 8.2.1 and for the all-shifts set-up in Section 8.2.2. Unless stated otherwise, the running time has been restricted to two hours. If no optimal **CMSLCP** solution is attained, the algorithm terminates with a sub-optimal solution. In order to generate these results, the **MSLCP**, **CMSLCP**, **APP** and **RAPP** are implemented implemented using Python and solved using Gurobi. For the implementation of the **RAPP**, the package NetworkX (Hagberg et al., 2008) is used. The corresponding maximum flow problem is solved using the preflow-push algorithm (see e.g. Cormen et al. (2009, p. 765)), that is included in the implementation of NetworkX.

The capacities of the initial **MSLCP** solution are calculated using the **APP** and presented in Appendix C.2, showing that the 21 maintenance shifts require more than 1 maintenance team, implying that the initial **MSLCP** violates the capacity constraints.

### 8.2.1 Single-shift set-up

The single-shift set-up focuses at finding a solution that satisfies the capacity constraints for one particular maintenance shift: the daytime maintenance shift in Zl at 11-4-2018. All cut generation variants have been run for two hours. For none of these variants, the **CMSLCP** has been able to find a solution that satisfies capacity constraints within two hours of running time. However, although for none of the cut generation variants the **CMSLCP** was able to find an optimal **MSLCP** solution, the solutions that it found did improve over multiple iterations, obtaining better (though still sub-optimal) solutions.

Below, first the convergence of the **CMSLCP** is discussed. Then, the two underlying causes (computation time and cut efficiency) are discussed to explain the convergence.

#### Convergence

Figure 8.1 provides a graphical representation of the development of the objective function of the **MSLCP** solution over time and over multiple iterations. It shows that all heuristic cut generation variants achieved an objective of approximately 887. To be precise, the heuristic cut generation variants' final objective values are between 887.277 and 887.281[1], thereby coming closest to the (unknown) optimal value and providing a lower bound (887.281) for it. Of these heuristic cut generation variants, the variants with higher number of cuts reach this objective value earlier (i.e. in less time and in less iterations) than the variants with lover number of cuts. The cut generation variant 'Binary Search Heuristic' with 15 cuts provided best, i.e. it achieved the value of 887 in the least amount of time and in the least amount of iterations.

When comparing the Binary Search Heuristic with the Basic Heuristic, it is found that their convergence is similar in terms of iterations, but that the convergence of the Binary Search Heuristic is a bit quicker time-wise. This is an indication that the improvement per iteration is comparable for both, but that the time consumption per iteration is less for the Binary Search Heuristic.

In solving the maintenance shift under consideration, the heuristic cut generation variants outperform both the naive and the min-cut cut generation variants, in time as well as in iterations. For the latter two, however, much more iterations were performed. This is an indication that the computation time per iteration is better for the min-cut and naive cut generation methods, but that the achieved approach to the optimal solution is worse.

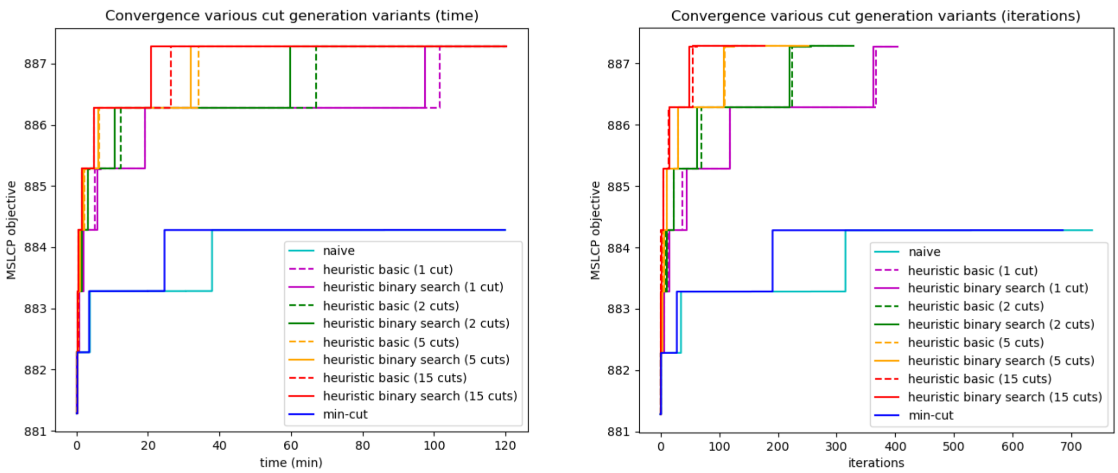



*Figure 8.1: Convergence of the **CMSLCP** in the single-shift set-up. For each cut generation variant, the course of the value of the **MSLCP** is displayed as a function of elapsed time (left) and as a function of the current iteration (right).*

In an attempt to nonetheless find an optimal value to benchmark the cut generation variants, the best-performing cut generation variant was run for 14 hours. The results are given in Figure 8.2. Unfortunately, even these 14 hours were not enough to find an optimal solution to the **CMSLCP**. The run however did provide a new lower bound to the optimal objective value of 888.279.

[1] Recall that the objective value is mainly composed of the total number of daytime activities. The reason that the value is nonetheless not integer is due to the fact that, besides a unit value for each daytime activity, also a value $\epsilon$ is added for every performed maintenance activity. See also Section 4.2.

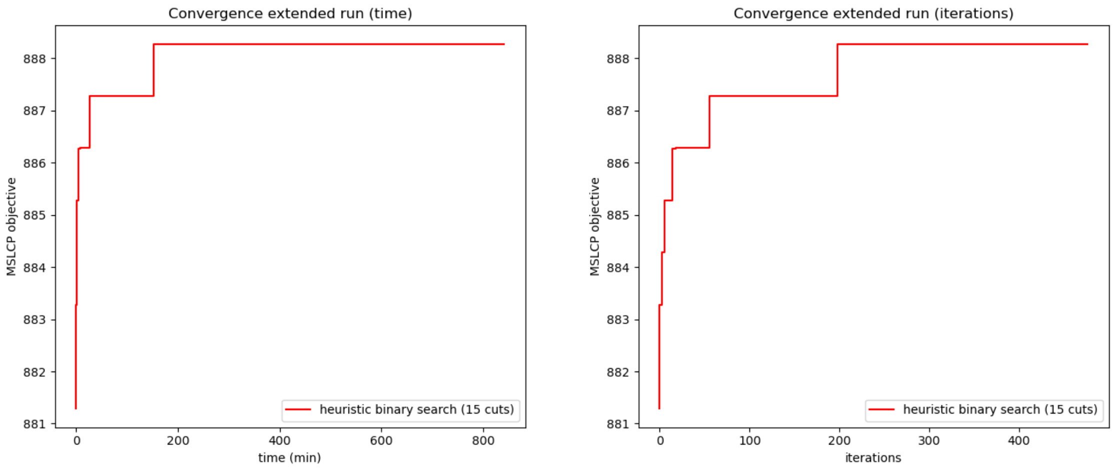



*Figure 8.2: Convergence of the **CMSLCP** in the single-shift set-up, for a long run of the heuristic cut generation variant with 15 cuts. The course of the value of the **MSLCP** is displayed as a function of elapsed time (left) and as a function of the current iteration (right).*

### Computation time

The convergence of the **CMSLCP**, measured in time (as opposed to the number of iterations) is dependent on the required computation time per iteration. Table 8.1 breaks down the computation time per iteration in various sub processes.

| | sub processes | | | | |
|---|---|---|---|---|---|
| | **MSLCP** | **APP** | cut gen. | other | total |
| naive | 9.3 | 0.3 | 0.0 | 0.1 | 9.8 |
| Basic Heuristic (1 cut) | 16.6 | 0.5 | 1.0 | 0.2 | 18.2 |
| Basic Heuristic (2 cuts) | 20.2 | 0.5 | 2.0 | 0.2 | 22.9 |
| Basic Heuristic (5 cuts) | 22.0 | 0.5 | 5.5 | 0.2 | 28.3 |
| Basic Heuristic (15 cuts) | 26.2 | 0.5 | 17.4 | 0.2 | 44.3 |
| Binary Search Heuristic (1 cut) | 16.3 | 0.5 | 0.9 | 0.2 | 17.8 |
| Binary Search Heuristic (2 cuts) | 19.4 | 0.5 | 1.9 | 0.2 | 21.9 |
| Binary Search Heuristic (5 cuts) | 23.1 | 0.6 | 5.0 | 0.2 | 28.8 |
| Binary Search Heuristic (15 cuts) | 25.0 | 0.5 | 14.9 | 0.1 | 40.5 |
| *extended run (14 hours)* | 87.8 | 0.7 | 17.1 | 0.2 | 105.7 |
| min-cut | 9.5 | 0.4 | 0.5 | 0.2 | 10.5 |

*Table 8.1: Computation time per iteration for each cut generation variant, in seconds, broken down into the main contributing processes to the computation time: the computation of an **MSLCP** solution subject to all cuts generated so far, the determination of a capacity violation using the **APP**, and the cut generation process itself, and other processes. The latter relates to all remaining computations, such as results storage. In addition to the standard cut generation variants, the results for the extended run of the Binary Search Heuristic cut generation variant with 14 hours of running time instead of 2 hours are presented.*

It can be observed that the naive and min-cut cut generation variants require the least time per iteration. This is in correspondence with the fact that in these variants many iterations could be run within 2 hours (see Figure 8.1).

Moreover, Table 8.1 shows that the generation of cuts in the Basic Heuristic version requires somewhat more time than the Binary Search Heuristic. This concurs with the expectation that can be drawn from the design of both heuristics: the Binary Search Heuristic improves upon the Basic Heuristic in the sense that it requires less iterations to generate a cut. Also, the iterations of the heuristic cut generation variants take more time for higher numbers of cuts, which is a direct result of the time it takes to generate more cuts.

The average running time of the **APP**, necessary to determine whether capacity of a maintenance shift is vioalted, is well below one second, and consistently so over all cut generation variants.

The most time is consumed by solving the **MSLCP**. Interestingly, the **MSLCP** takes more time to run in the heuristic cut generation variants than it does in the naive and min-cut cut generation variants. To understand this, it is relevant to look at the computation time of the **MSLCP** for the extended run of 14 hours. Figure 8.3 presents it as a function of the current iteration. It shows that the running time of the **MSLCP** (as well as its variance) increases for later iterations. The expected explanation for this is that due to the added cuts, the **MSLCP** becomes increasingly constrained and solving it becomes increasingly hard. This leads to higher computation times for the **MSLCP**. It is no surprise that, in Table 8.1, this effect is most clearly visible for the heuristic cut generation variant with 15 cuts. This variant performs best and constrains the **MSLCP** most rapidly of all variants; hence, the longer computation times that are the result of this increasingly constrained **MSLCP** are first encountered in this cut generation variant.

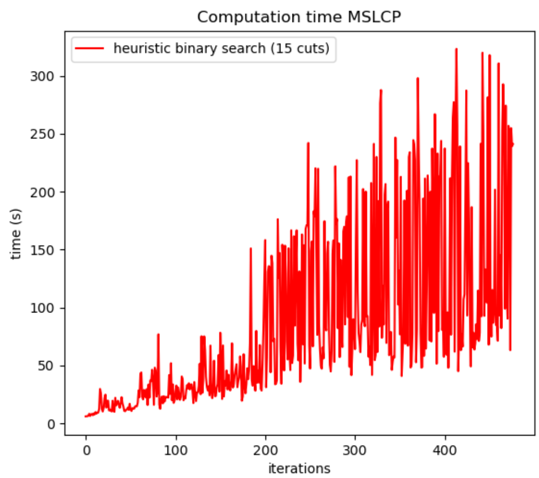



*Figure 8.3: Computation time of the **MSLCP** in seconds, per iteration of the **CMSLCP** for the Binary Search Heuristic cut generation version with 15 cuts per iteration, in an extended run of 14 hours.*

### Cut efficiency

Besides the computation time, the convergence of the **CMSLCP** is also dependent on the efficiency of the generated cuts. Table 8.2 presents for each cut generation variant, the number of jobs per cut, which is a measure of the efficiency of the generated cuts.*It must be noted that the number of jobs per cut must be interpreted with care: since the implementation did not filter jobs with 0 duration (i.e. rolling stock units that did not need to receive maintenance), these jobs may have become also be part of the cuts, whereas they are in fact not restrictive. This does not mean that the result is not valid, but it does imply that, had those jobs been filtered out first, the number of jobs per cut could have been lower and the cuts could have been more efficient.*

| | # jobs/cut |
|---|---|
| naive | 18.0 |
| Basic Heuristic (1 cut) | 10.7 |
| Basic Heuristic (2 cuts) | 10.6 |
| Basic Heuristic (5 cuts) | 10.8 |
| Basic Heuristic (15 cuts) | 11.0 |
| Binary Search Heuristic (1 cut) | 10.5 |
| Binary Search Heuristic (2 cuts) | 10.7 |
| Binary Search Heuristic (5 cuts) | 10.9 |
| Binary Search Heuristic (15 cuts) | 11.3 |
| min-cut | 9.3 |

*Table 8.2: The average number of jobs present in a cut, for each cut generation method.*

It is observed that the min-cut cut generation method generates the most restrictive cuts. However, to its disbenefit it must be noted that it does not produce many cuts per iteration. For the current maintenance shift under consideration, it produced each 1 cut at maximum (although for other shifts, it may be capable of producing more cuts per iteration; this depends on the nature of the jobs in a maintenance shift). Furthermore, the naive cut generation variant contained most jobs per cut, reflecting the fact that it does not use any intelligence to generate the cuts. Lastly, performance of each of the heuristic cut generation variants are comparable, although their efficiency is somewhat less than produced by the min-cut cut generation variant.

### 8.2.2 All-shifts set-up

Section 8.2.1 examined the results obtained by applying the **CMSLCP** to a single maintenance shift that appeared hard to solve. However, in realistic cases, not all maintenance shifts are as hard to solve. Therefore, the current section considers the all-shifts set-up, attempting to solve the capacity violations for all shifts of the problem instance.

As in the single-shift set-up, no optimal solutions were found within the running time restriction of two hours. Note that this is expected: after all, the all-shifts set-up is more restrictive than the single-shift set-up, since it prevents the capacity violation of more maintenance shifts.

#### Capacity violation

In the initial **MSLCP** solution, maintenance activities are assigned to 35 different maintenance shifts. Of these maintenance shifts, 21 require a capacity of more than 1 maintenance team. Since in the current set-up it is assumed that only 1 maintenance team is available, this implies that the capacity of those 21 maintenance teams is violated.

The **CMSLCP** has been applied to reduce the number of maintenance shifts for which the capacity is violated. Figure 8.4 presents the number of capacity violations as a function of elapsed time and as a function of the current iteration.

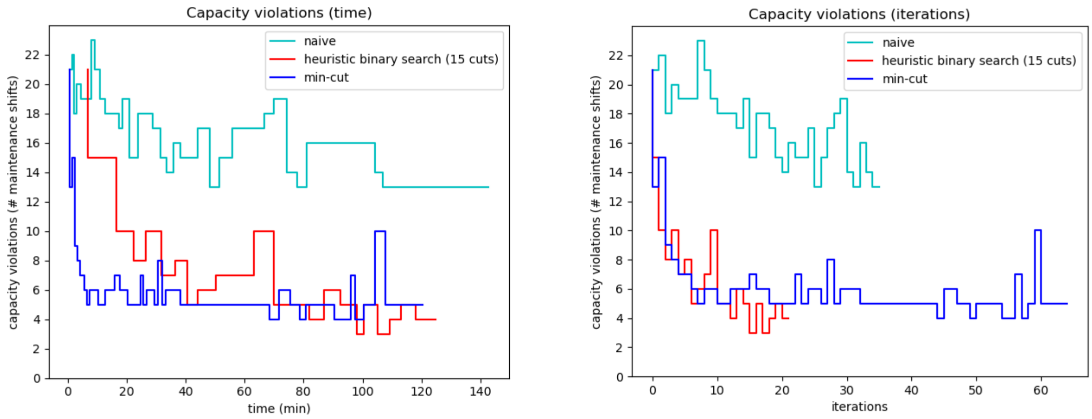



*Figure 8.4: Number of shifts for which the capacity is violated (i.e the required capacity is more than 1 maintenance team), for three cut generation variants, as a function of elapsed time (left) and as a function of the current iteration (right). The naive cut generation method took longer than 2 hours since solving the **MSLCP** in the last iteration (that started before the threshold of 2 hours of running time) took very long; the process terminated as soon as this iteration was finished.*

First, it becomes clear that the number of maintenance shifts for which the capacity is violated is decreasing. However, the decrease is not strictly monotonic. The added cuts as a result of the violation of capacity in one of the maintenance shifts, may induce a new **MSLCP** solution that assigns maintenance in such a way that the capacity of maintenance shift which was formerly sufficient, now becomes violated.

Of the three cut generation variants investigated, the naive cut generation variant is clearly the worst performing. After two hours of running time, it contains considerably more maintenance shifts for which capacity is violated than the other two cut generation variants.

More strikingly is the development of the number of violations in the min-cut cut generation variant compared to the Binary Search Heuristic cut generation variant.

When looking at the development in terms of the elapsed time, the capacity violations in the min-cut cut generation variant decrease at the beginning much more sharply than in the binary search cut generation variant, after which they in both remain constant for around 5 capacity violations. The min-cut cut generation variant found a solution with 5 maintenance shift violations or less after 7.6 minutes, whereas the Binary Search Heuristic cut generation method found such a solution only after approximately 44.2 minutes. The practical implications of this are relevant: when no feasible solution can be obtained in reasonable time, the preferred option is to get a good sub-optimal solution as quick as possible. The min-cut cut generation method seems better suited for this goal.

To gain a little more understanding on this behavior, observe also the capacity violations as a function of the current iteration. At the beginning, the Binary Search Heuristic and min-cut cut generation variants show a similar path. This implies that, in each iteration, the resulting cuts in both variants lead to similar benefits in the reduction of capacity violations. However, the running time of the min-cut cut generation method per iteration is considerably lower than in the Binary Search Heuristic cut generation method, leading to a better performance in terms of computation time.

### Convergence

Figure 8.5 displays the convergence of the **MSLCP** objective value in the all-shifts set-up.

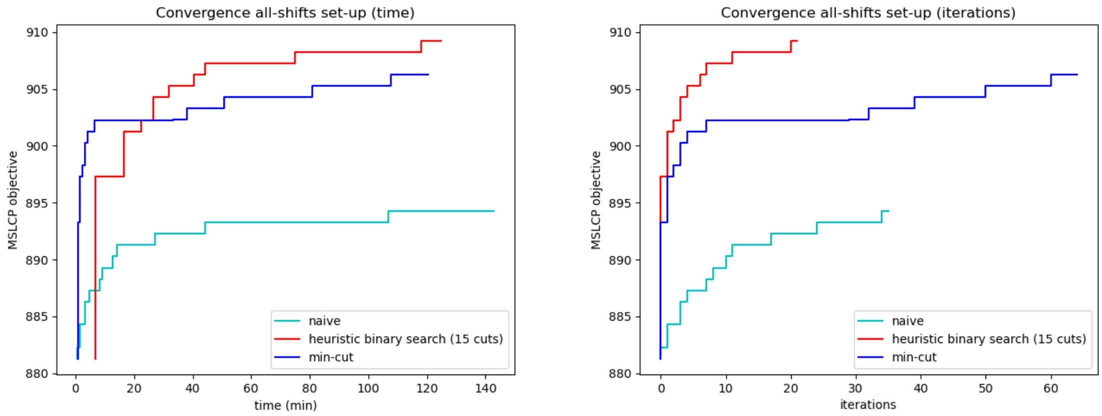



*Figure 8.5: Convergence of the* ***CMSLCP*** *in the all-shifts set-up. For three generation variants, the course of the value of the* ***MSLCP*** *is displayed as a function of elapsed time (left) and as a function of the current iteration (right).*

At first it can be noted that the course of the **MSLCP** objective value for the three investigated cut generation variants is similar as in, for instance, Figure 8.1: as a result of the added cuts, the value of the objective of the **MSLCP** gradually increases.

As in Figure 8.4, it can be seen that, in terms of iterations, the course of the **MSLCP** objective at the beginning of the run is very similar for the min-cut and Binary Search Heuristic cut generation processes.

In the first couple of iterations (right side of Figure 8.5), both methods are equally capable of detecting 'simple' infeasible combinations of jobs that, when added as a cut to the **MSLCP**, immediately cause a unit step in the **MSLCP** objective. The min-cut cut generation method has an advantage, since its running time per iteration is shorter. This is reflected in the left side of Figure 8.5, where the increase in objective value is quicker in case of the min-cut cut generation method.

In a later stage, however, the cuts added by the Binary Search Heuristic cut generation method yield a better convergence of the **MSLCP** (right side of Figure 8.5). Hence, from a time perspective, in a later stage the Binary Search Heuristic cut generation method overtakes the min-cut cut generation method (as can be seen in the left of Figure 8.5).

From a practical point of view, however, it may be true that it is not absolutely necessary to provide a solution in which all capacity violations are solved, and that the capacity violation reduction obtained in this first stage is already sufficient. The benefit of the quicker convergence time may outweigh the disbenefit of a solution with some capacity violations. In this case, the min-cut cut generation method may be the preferred option.

# Chapter 9

# Use case

While Chapter 7 aimed to run the **MSLCP** model on numerous scenarios to gain insight in the behavior of the model for many different parameter values, the current chapter takes a different perspective. It aims to return to the original problem faced by railway operator NS and demonstrate how this problem can be solved with the help of software programme *Viriato* in combination with the developed **MSLCP** model.

Section 9.1 returns to the problem NS is facing. Then, Section 9.2 lays out the approach that is used to solve this problem: it proposes a use case for which solutions can be produced to gain insight in the practical implications of the matter investigated in the earlier sections of the current research. These solutions are produced with the help of software programme *Viriato*. Then, Section 9.3 provides results for the proposed use case. Section 9.4 indicates how capacity can be addressed in the current framework. Section 9.5 presents a discussion of the results obtained for the current use case. Since Chapter 7 and the current chapter both present results based on the **MSLCP** tool, it also includes a short note (in Section 9.5.3) on how the setup in the current chapter differs from the results obtained in Chapter 7.

## 9.1 NS problem revisited

As already explicated in Section 1.7.2, the problem NS is facing is centered around the increasing pressure on capacity of maintenance locations during nighttime. As a result, the objective of NS is to achieve a decrease this pressure by performing more maintenance activities during daytime. This potential policy switch, however, comes with at least two major complicating factors. First, performing maintenance during daytime as well as during nighttime adds a considerable complication to the maintenance planning process. Second, it is not straightforward which locations need to be opened for daytime maintenance. Usually is not desirable to open all potential maintenance locations during daytime, since the capacity use of many of these locations during daytime would often be too low.

Therefore, the problem of NS is twofold. First, an efficient method to produce a feasible planning needs to be devised. Second, a method to determine optimal maintenance locations for daytime maintenance is necessary.

## 9.2 Approach

The goal of this section is to define a use case and to explain how solutions for this use case are generated. As such, it is a prelude to the actual results presented in Section 9.3.

The goal of the use case is to acquire insights in the workings of the model, to show the potential of daytime maintenance planning and to demonstrate the impact of the maximum number of locations that can be opened for daytime maintenance on the extent to which the objective of NS is reached. The use case is presented in two versions: the small-scale version and the large-scale version. The small-scale version includes a limited number of rolling stock units and is, as such, suitable to test the model on a small scale and present readable visualizations of solutions. The large-scale version is an extension of the small-scale version including all rolling stock units used for intercity services. Due to the higher number of rolling stock units it is less suited to explain the model workings, but on the other hand it represents a more realistic situation. After all, in practical situations, many rolling stock units are involved and each of them influences the optimal maintenance location choice.

Section 9.2.1 explains the characteristics of the use case at hand. Then, Section 9.2.2 describes the steps that are taken to generate solutions for this use case, Section 9.2.3 introduces the scenarios for which solutions are generated, and Section 9.2.4 indicates how the quality of these solutions are measured.

### 9.2.1 Use case specification

This section describes the use case at hand. First its small-scale version is discussed; then the large-scale version is presented by discussing the differences compared to the small-scale version.

#### Small-scale version

Similar to Chapter 7, NS BDu data is used, which is constructed according to the same procedure as discussed in Section 7.1.1. It concerns rolling stock circulation data of NS, specifying all planned trips in a specified time horizon. For the current use case, the BDu data for a period of 7 days, between June 10, 2019 and June 16, 2019, is used (originating from the BDu for period 10, see Table 7.1). The use case contains all rolling stock units of rolling stock type *DDZ4*, of which 24 exist in the data set. This rolling stock type is useful for the current purposes since there are not too many rolling stock units of this type (so that the problem size is restricted, which helps in understanding the results than can be acquired) and the rolling stock units spread out over the entire network (which is representative in most use cases in The Netherlands).

The data for the use case has been imported in the software program *Viriato*. Viriato is comprehensive planning software produced by *SMA und Partner A.G.*. Using Viriato, visualizations of the implemented rolling stock circulation can be produced. A more extensive discussion on the use of this software for the current purposes is given in Section 10.3.

To demonstrate the contents of the use case, Figure 9.1 presents the planned rolling stock movements for the rolling stock units in the use case on Monday June 10, 2019. For example, the first row corresponds to the rolling stock circulation for a specific rolling stock unit on June 10, 2019. It is planned to depart from On at 06.09 and departs in Gn at 06.17. Then, it stands still in Gn until 07.16, when it departs in the direction of Ut. It arrives in Ut at 09.11, where it stands still for some minutes. It departs at 09.18 in the direction of Rtd where it arrives at 09.55, and so forth.

At the following nodes, the rolling stock units under consideration start or terminate train services: Amsterdam Central Station (Asd), Bokkeduinen yard (Bkd), Binckhorst yard (Bkh), Groningen (Gn), The Hague Central Station (Gvc), 's Hertogenbosch (Ht), Lelystad (Lls), Lelystad yard (Llso), Leiden (Ledn), Leeuwarden (Lw), Onnen yard (On), Roosendaal (Rsd), Rotterdam

(Rtd), Utrecht (Ut), Zwolle (Zl), Zutphen yard (Zpge). Recall that it is assumed that maintenance can only be performed during maintenance opportunities, that is when rolling stock units have a planned standstill between different train services. Hence, intermediate nodes, where rolling stock units may stop but do not start or originate a train service, are not considered: these are not relevant in the current context since no maintenance is performed at these locations.

Figure 9.2 presents the rolling stock circulation of three particular rolling stock units (out of the 24 rolling stock units in total) for four days. For example, in Figure 9.2a, the first row corresponds to the rolling stock circulation for a rolling stock unit on June 10th, 2019; the second row corresponds to the rolling stock circulation of this same rolling stock unit on Tuesday, June 11th, 2019, and so further. Note that, although the planning horizon was 7 days, only the first 4 days of this period are presented here to maintain readability.

*(Text continues after figure.)*

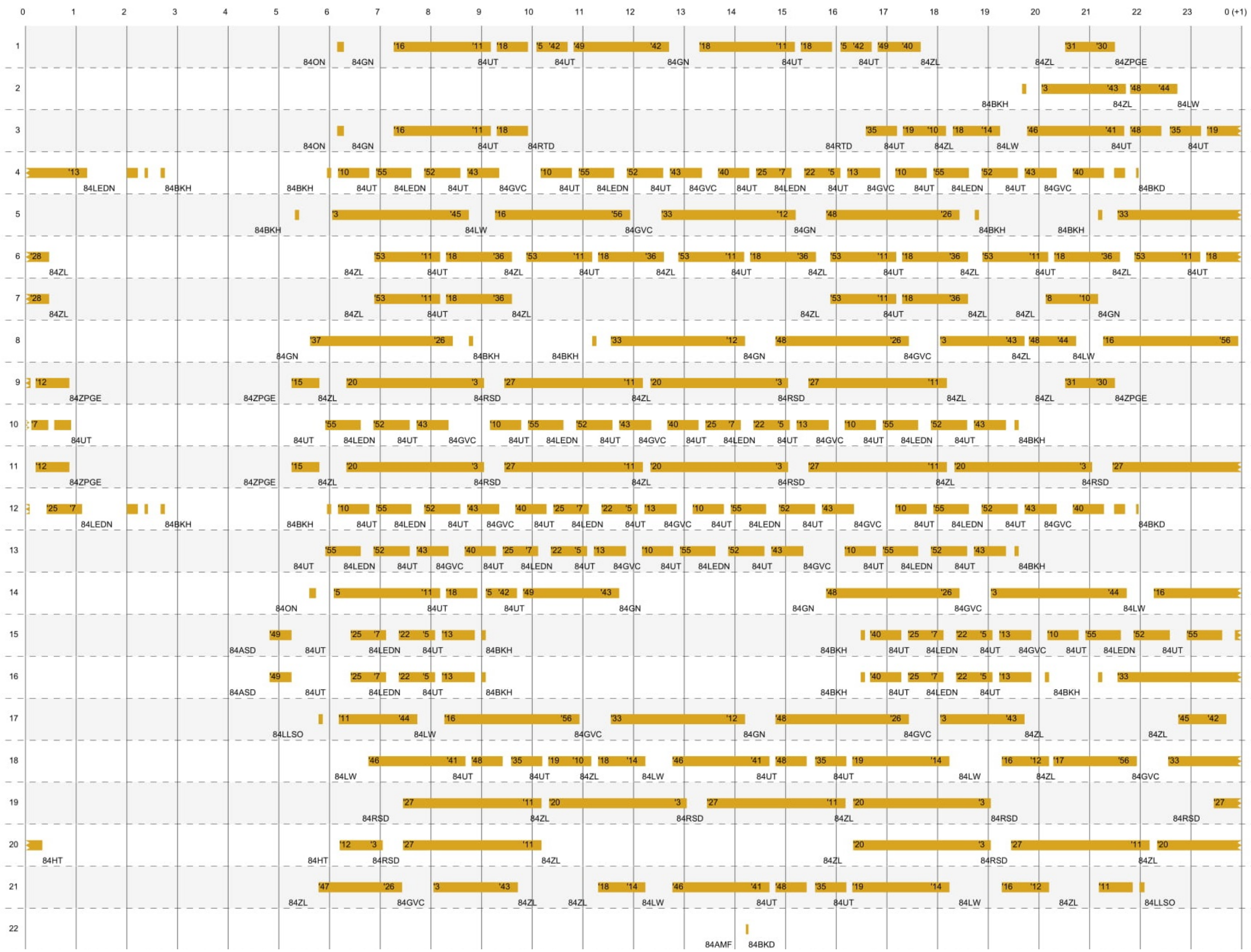
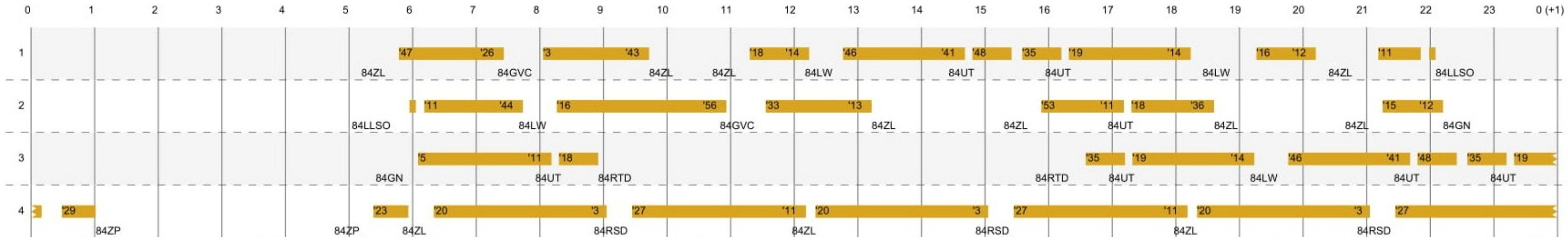
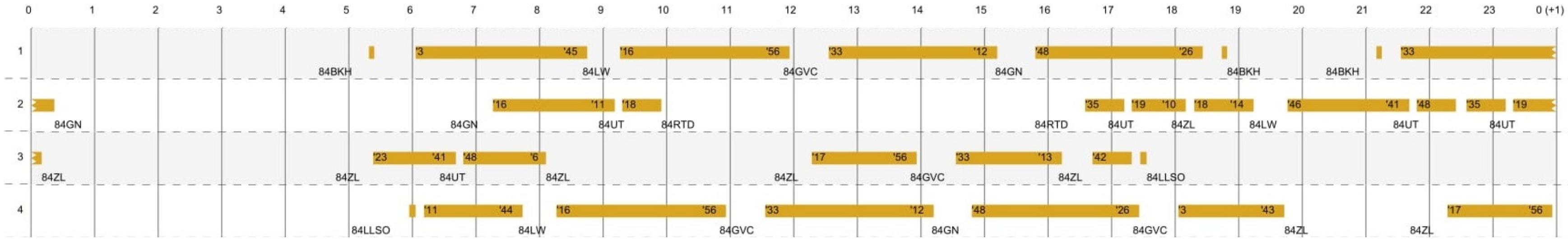


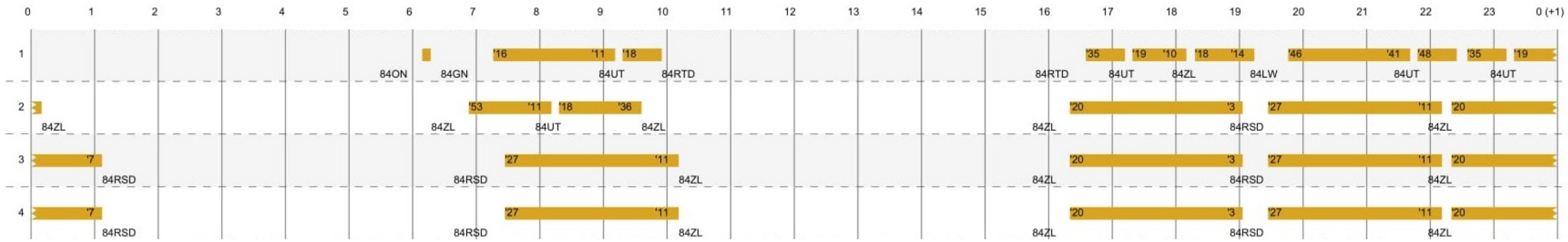
Figure 9.1: Overview of all planned rolling stock circulations in the small-scale version of the use case, described in Section 9.2, on Monday June 10, 2019. Each row corresponds to the rolling stock circulation of one rolling stock unit. Although the use case contains 24 rolling stock units in total, hence this figure presents the rolling stock circulations for 22 rolling stock units: 2 rolling stock units did not have any planned movements on the presented day and are therefore omitted.



*(a) Rolling stock unit 1*



*(b) Rolling stock unit 2*



*(c) Rolling stock unit 3*

*Figure 9.2: Rolling stock circulation of the use case under consideration, for three different rolling stock units (Figure 9.2a-9.2c) and four different days, with no maintenance assigned to it. For each rolling stock unit, the rolling stock movements with location and time for each planned departure and arrival are indicated. The four rows presented correspond to the rolling stock circulation of a rolling stock unit for four subsequent days from 10 June 2019 to 13 June 2019. The model determines MOs in which maintenance activities are executed, rather than their exact timing, and hence the visualized timing of maintenance activities should be interpreted with care. For reference, rolling stock units 1, 2 and 3 are listed in Viriato under vehicle id 3, 5 and 7, respectively.*

**Large-scale version**

The large-scale versions shows many similarities when compared to the small-scale version: the same BDu input period (period 10) is used, and the input data contains 7 days between June 10, 2019 and June 16, 2019. The difference of the large-scale version compared to the small-scale version is that the large-scale version contains (much) more rolling stock units: specifically, it contains all rolling stock units that are used for intercity services, that is all rolling stock units of types VIRM4, VIRM6, ICM3, ICM4, DDZ4, DDZ6 and ICD. Table 7.2 already indicated that this entails 360 rolling stock units in total. Of these, 348 were active in the 7-day period considered in the large-scale version of the use case.

Due to its large size, visualizations for this version are less comprehensible and therefore not presented. Moreover, this version of the use case was not implemented in *Viriato*: it was expected that some time would be required to import the data into Viriato and more since no visualizations needed to be presented, it was chosen to generate results outside of Viriato. This choice does not affect the eventual KPIs.

### 9.2.2 Solution generation method

To mitigate the capacity problem that NS is facing during nighttime, it is desirable to produce a schedule that minimizes the total number of maintenance activities during the night. To address the problems of NS properly, two aspects need to be optimized, where optimality is defined as minimizing the total number of activities during nighttime while still satisfying the required intervals between maintenance activities. On the one hand, an optimal maintenance schedule needs to be created. On the other hand, optimal locations for daytime maintenance need to be found. Observe that these two aspects influence each other: the optimal maintenance schedule is dependent on the locations used for maintenance, and similarly, the optimal locations used for daytime maintenance depend on the maintenance schedule that can be made when these locations are open. Therefore, the two aspects need to be addressed simultaneously.

In addition, several things need to be taken into account. First, a distinction between daytime maintenance and nighttime maintenance needs to be made to address properly the problem NS is currently facing during nighttime. Second, the number of maintenance locations for daytime maintenance that can be opened is usually restricted. Third, the number of maintenance activity types, their durations and their intervals are not in any situation the same and need to be variable by the user of the method.

To this end, the **MSLCP** model proposed in the current research is used together with planning software Viriato. The input of the model is a rolling stock circulation (or *roster*) and the duration and intervals of the maintenance types that need to be scheduled. The output is the rolling stock circulation with maintenance activities assigned to it (in such a way that the maintenance intervals are satisfied) and the set of locations that need to be opened for daytime maintenance. The **MSLCP** model is applied to the current use case.

Viriato has a loose connection to the **MSLCP** model so that the rolling stock maintenance activities can be scheduled using the **MSLCP** model within Viriato. This capability is demonstrated using the small-scale version of the rolling stock circulation of the use case described in Section 9.2.1.

### 9.2.3 Scenario design

Below, first the parameters that are relevant in the current use case are discussed, and then the specific settings for the small-scale and large-scale scenario are given.

**Parameters**

The following parameters need to be determined to generate results for the current use case.

- *Use case version.* In Section 9.2.1 two versions of the use case have been introduced: the small-scale version and the large-scale version. The large-scale version contains much more rolling stock units compared to the small-scale version. The choice of the use case therefore influences the size of the considered problem.

- *Maximum number of daytime maintenance locations.* The maximum number of daytime maintenance locations influences the number of maintenance activities that can be performed during daytime and hence influences to what extent the goal of NS (i.e. reducing the capacity pressure during nighttime) can be achieved. In the current analysis, results are produced for 0, 1, 2, 3, 4 and 5 maximum maintenance locations that can be opened.

- *Maintenance types.* In the default situation, two maintenance types apply: maintenance type A having a duration of 45 minutes and a maximum interval of 24 hours, and maintenance type B having a duration of 90 minutes and a maximum interval of 48 hours. This is referred to as the *default maintenance types.* Besides, in a separate scenario, short durations are considered where maintenance type A has a duration of 30 minutes (and a maximum interval of 24 hours) and maintenance type B has aduration of 45 minutes (and a maximum interval of 48 hours). Results for this separate scenario are presented in Appendix D.2.

- *Start of the daytime and nighttime time windows.* In all scenarios of the current analysis current analysis, the start of the daytime time window is assumed to be at 07.00 and the start of the nighttime time window is assumed to be at 19.00. These together define the time windows for daytime as well as nighttime maintenance and are referred to as the *default time windows.* Recall that any maintenance activity assigned to a maintenance opportunity is assumed to be performed during the day if and only if the corresponding maintenance opportunity starts and ends during daytime of the same day; else it is assumed that maintenance is performed during the night. Hence, each maintenance opportunity is classified to be either a daytime or a nighttime maintenance opportunity. By choosing the start of the daytime and nighttime time windows, one has control over this classification. In an earlier stage of the research, a scenario was considered with the start of the daytime time window at 10.00, but this has shown to yield undesirable results (see Appendix D.1).

**Small-scale scenario**

This scenario is based on the small-scale version of the use case, meaning it contains a limited number of 24 rolling stock units. It considers 0, 1, 2, 3, 4 and 5 maximum maintenance locations that can be opened, default maintenance durations and default time windows. It considers two maintenance types: maintenance type A has a duration of 45 minutes and a maximum interval of 24 hours, and maintenance type B has a duration of 90 minutes and a maximum interval of 48 hours.

**Large-scale scenario**

In the large-scale scenario, the same settings apply as in the small-scale scenario, with this difference that the large-scale version of the use case (including 348 rolling stock units) is considered.

### 9.2.4 Use case KPIs

When a solution to the problem indicated in Section 9.1 is found, it is important to measure to what extent the goals of NS are reached in this solution (in other words, to define the 'quality of a solution'). To this end, the following aspects are of interest. These aspects are computed in the Section 9.3 to assess the quality of the presented solutions.

- *Opened locations.* It is important for railway operators to know which locations need to be opened according to the model.
- *The average number of activities and hours of activity per day.* This gives an indication of the total workload in a scenario.
- *The percentage of activities performed during daytime.* Since the goal of NS is to move work from the night to the day, it is of interest to know how well this goal is achieved. The 'day share', i.e. the percentage of hours of activity that is carried out during daytime, is a useful measure to express this. (Although the objective of NS of reducing the amount of nighttime activity, focusing on the day share is just another appropriate way of looking at the problem of NS: after all, an increasing day share reflects the total amount of work that can be moved *from* the night to the day. As a result, an increasing day share corresponds to a decrease of nighttime activity and hence to better achievement of the goals of NS.)
- *The average costs per day.* One of the most important drivers of railway operators are costs. In order to give insight into the costs of a scenario, costs are calculated using both an optimistic and a conservative method. This calculation is detailed below.
- *The workload distribution over various locations.* From a managerial perspective, it is important to gain insight in the expected hours of work that need to be performed at each location.

#### Cost calculation

The costs are calculated using an optimistic method and a conservative method. Both methods are based on the hours of activity at a maintenance location and both methods assume that one hour of activity costs € 50 during nighttime and € 30 during daytime. These values are fictitious.

In the optimistic method, the number of hours of activity is multiplied by the costs per hour. As a result, if to a location on average 11 hours of workload are assigned during nighttime and 3 hours of activity during daytime, this costs $11 \times €50 + 3 \times €30 = €640$ on average per day according to the optimistic method.

The conservative cost computation method takes a different approach. Note that the optimistic method assumes that the costs are proportional to the amount of workload. However, in many situations this is not reasonable. For example, in order to perform one activity, usually one employee to be paid for an entire day. To map this characteristic, in the conservative cost computation method costs are incurred per eight hours: for 2 hours of workload, 8 hours need to be paid (one employee); similarly, for 10 hours of workload, 16 hours need to be paid (two employees). Hence, the aforementioned example with 11 hours of nighttime activity and 3 hours of daytime activity, this costs $2 \times 8 \times €50 + 1 \times 8 \times €30 = €1,040$ on average per day according to the conservative method: it would require to pay two employees for an entire shift during nighttime and one employee for an entire shift during daytime.

The above assumes that the maintenance durations are specified in such a way that maintenance activities can be performed by one employee: for example, if the maintenance duration is 60 minutes, it is assumed that it can be finished in 60 minutes by one employee. This is not always the case: often, more employees work together on a task. If applicable in the situation under consideration, the cost calculation method must be adapted accordingly.

In both methods, the costs are first calculated per location. The resulting values are summed to obtain a cost indication for the entire network.

## 9.3 Results

The current section presents solutions for the problem of NS. Results are obtained for two scenarios, introduced in Section 9.2.3. The results for the small-scale and large-scale scenario are presented below.

Both scenarios have been run for maintenance durations of 45 minutes for type A and 90 minutes for type B, and with the start of the daytime time window at 10.00.

As mentioned before, two additional iterations of this small-scale scenario have been run to discover appropriate parameter values for the daytime time window and for the maintenance durations. Results for these scenarios are presented in Appendix D. Sorter maintenance durations resulted in maintenace in maintenance schedules that are too tight in reality. A later start of the daytime time window resulted in invalid maintenance schedules. These scenario settings have therefore not been further pursued.

### 9.3.1 Small-scale scenario

#### Visualizations

Section 9.2 already introduced Figure 9.2, which visualizes the rolling stock circulation for three rolling stock units for four days (June 10th, 2019 until and including June 13th, 2019). This section presents similar figures, but then including maintenance corresponding to the set-up of the small-scale scenario.

Figure 9.3 presents a visualization of the solution with no possibilities for daytime maintenance and Figure 9.4 presents a visualization of the solution with at most three locations opened for daytime maintenance. Although the planning horizon was 7 days, only the first 4 days of this period are presented here. Type A maintenance activities are indicated by blue rectangles and Type B maintenance activities are indicated by black rectangles. Recall from Chapter 4 that maintenance activities are scheduled by the model in a maintenance opportunity, and the model does not determine the exact time when the maintenance activity is to be performed. For now, however, the maintenance activities are planned in the middle of the maintenance opportunity in which they need performed. If multiple maintenance activities of different types are scheduled in the same maintenance opportunity, they are planned in such a way that the maintenance activities are equally distributed over the maintenance opportunity. This way of planning the maintenance activities within the maintenance opportunities in which they are scheduled may be naive and can possibly be optimized.

A complete rolling stock circulation for all 24 rolling stock units, for 7 days, for the situation with three locations opened for daytime maintenance is presented in Appendix D.4.

Several observations can be made from these visualizations that correspond to the expectations from the model formulation.

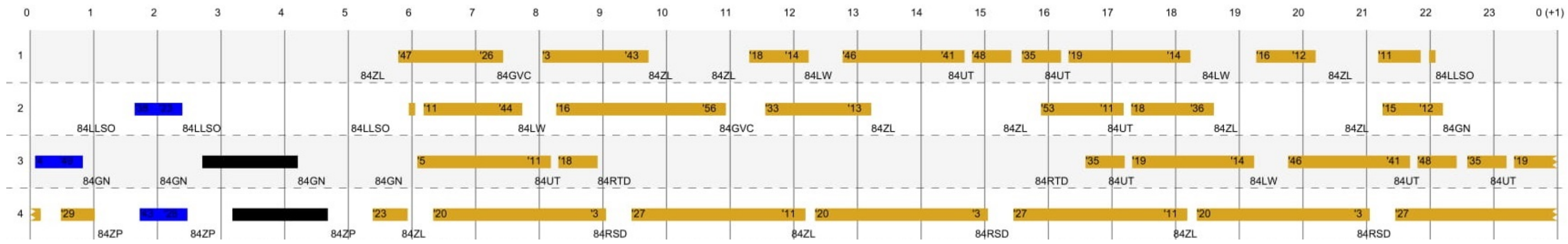
- The rolling stock units are as-good-as-new at the beginning of the time horizon. This becomes for example visible in Figure 9.3a, where no maintenance activities need to be performed on the first day.

- The intervals between maintenance activities are measured from the end of the first maintenance opportunity until the start of the second maintenance opportunity, which can, for example, be observed in Figure 9.3b. On day 2, maintenance activities of Type A and Type B are scheduled in the maintenance opportunity from 09.55 to 16.35 in Bkh on day 2. The next maintenance of Type A is scheduled on day 3 in the maintenance opportunity from 08.06 to 12.17 in Zl. The start of this second maintenance opportunity (08.06 on day 3) is less than 24 hours after the end of the first maintenance opportunity (16.35 on day 2) and hence this interval between maintenance activities satisfies the criterion. The same holds for the next maintenance activity of Type B: it is scheduled in the maintenance opportunity starting on day 3 at 17.33 and ending on day 4 at 05.57 in Llso. The start of this second maintenance opportunity (17.33 on day 3) is less than 48 hours after the end of the first maintenance opportunity (16.35 on day 2) and hence this interval between maintenance activities satisfies the criterion.

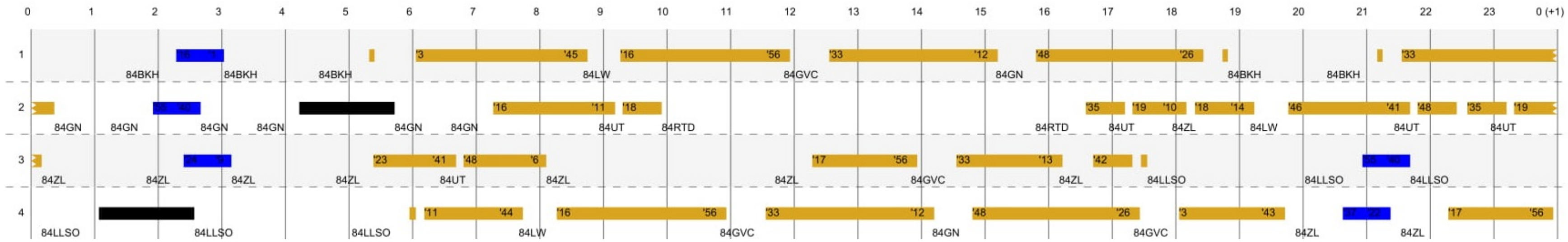
- Figure 9.3 and 9.4 present results for situations where 0 and 3 maintenance locations can be opened during daytime at maximum, respectively. It is visible that across these scenarios, the number of daytime maintenance activities increases. In the situation with 3 maintenance location, for example, it turns out that Rtd, Zl and Bkh are opened during daytime. This results in maintenance opportunities during daytime (see for example the daytime maintenance activities in Zl and Rtd in Figure 9.4c).

- Recall that a maintenance opportunity is classified to be during daytime if and only if its start time is after 07.00 and its end time is after 19.00 of the same day. Hence, the maintenance opportunity for rolling stock unit 2 on day 2 in Rtd (between 09.55 and 16.35) is during daytime, and hence it can be used for daytime maintenance if Rtd is opened for daytime maintenance (which is the case in Figure D.5b but is not the case in Figure D.3b). Similarly, the maintenance opportunity for rolling stock unit 2 on day 3 in Llso, starting at 17.33, is during nighttime: although it starts during daytime (before 19.00), it does not end during daytime of the same day and hence it is classified a daytime maintenance opportunity. As a result, it becomes possible to do maintenance here, since Llso is only opened during nighttime. This can be seen for instance in Figure 9.4b.

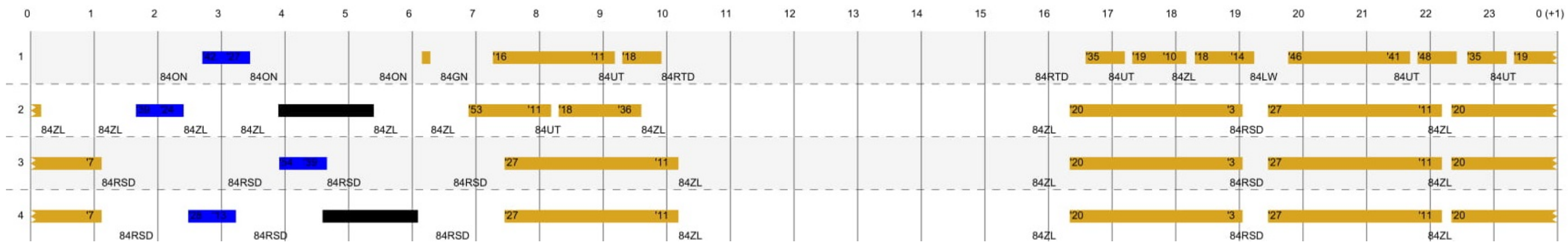
*(Text continues after the figure.)*




*(a) Rolling stock unit 1*

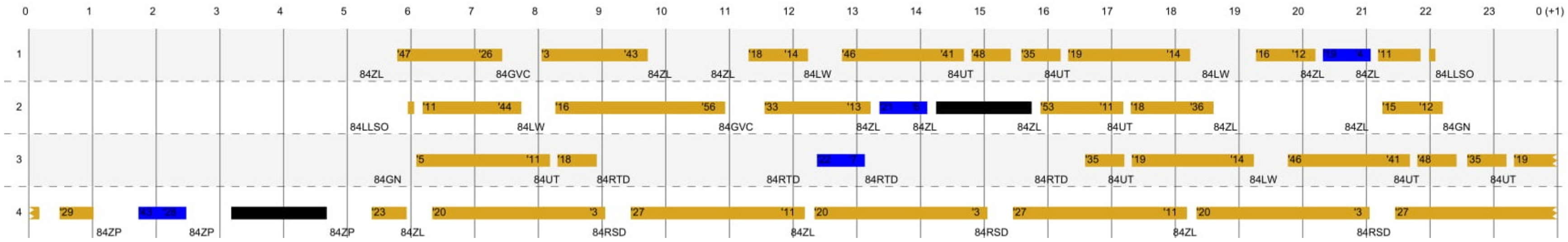



*(b) Rolling stock unit 2*

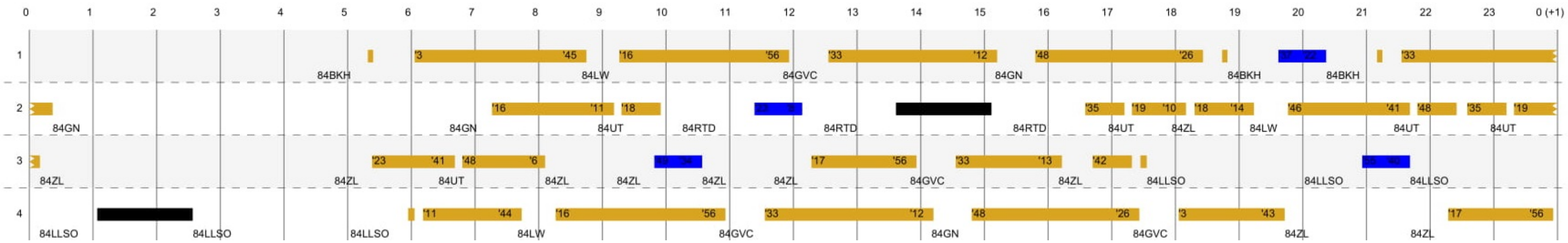



*(c) Rolling stock unit 3*

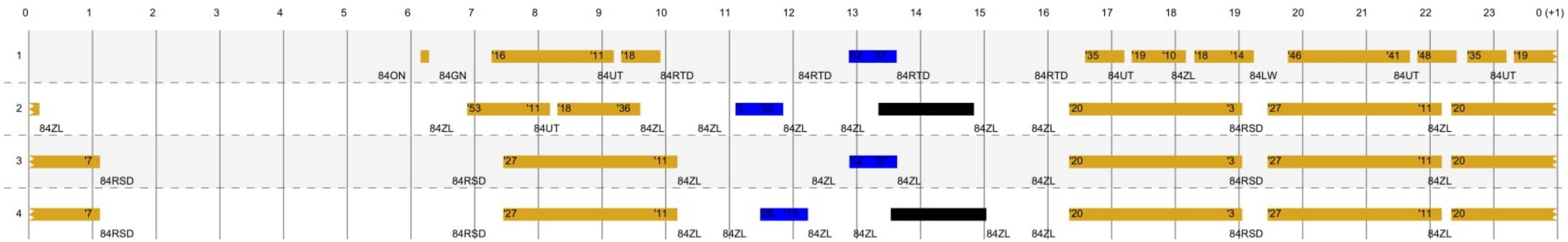
*Figure 9.3: Visualizations for the small-scale scenario, of a maintenance schedule obtained by the **MSLCP** if no daytime maintenance is allowed, for three rolling stock units. For each rolling stock unit, the rolling stock movements with location and time for each planned departure and arrival are indicated. The four rows presented correspond to the rolling stock circulation of a rolling stock unit for four subsequent days from 10 June 2019 to 13 June 2019. The model determines MOs in which maintenance activities are executed, rather than their exact timing, and hence the visualized timing of maintenance activities should be interpreted with care. For reference, rolling stock units 1, 2 and 3 are listed in Viriato under vehicle id 3, 5 and 7, respectively.*



(a) Rolling stock unit 1



(b) Rolling stock unit 2



(c) Rolling stock unit 3

*Figure 9.4: Visualizations for the small-scale scenario, of a maintenance schedule obtained by the* ***MSLCP*** *if three locations can be opened for daytime maintenance at maximum, for three rolling stock units. For each rolling stock unit, the rolling stock movements with location and time for each planned departure and arrival are indicated. The four rows presented correspond to the rolling stock circulation of a rolling stock unit for four subsequent days from 10 June 2019 to 13 June 2019. The model determines MOs in which maintenance activities are executed, rather than their exact timing, and hence the visualized timing of maintenance activities should be interpreted with care. For reference, rolling stock units 1, 2 and 3 are listed in Viriato under vehicle id 3, 5 and 7, respectively.*

**KPIs**

Firstly, the number of chosen locations is given in Table 9.1.

| # locations | Bkh | Zl | Rtd | Gvc | Bkd |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | x | 0 | 0 | 0 | 0 |
| 2 | x | x | 0 | 0 | 0 |
| 3 | x | x | x | 0 | 0 |
| 4 | x | x | x | x | 0 |
| 5 | x | x | x | x | x |

*Table 9.1: Locations opened in the small-scale scenario for a varying number of maximum number of daytime maintenance locations that can be opened.*

Observe that, iteratively, Bkh, Zl, Rtd, Gvc and Bkd are opened.
Several KPIs for this scenario are presented in Table 9.2.

| | | | | costs (€) | |
|---|---|---|---|---|---|
| # locations | # activities | hrs. of activity | day share | optim. | cons. |
| 0 | 29.9 | 29.8 | 0.0% | 1,489 | 6,800 |
| 1 | 29.9 | 29.8 | 5.4% | 1,457 | 7,040 |
| 2 | 30.1 | 30.2 | 12.1% | 1,438 | 7,280 |
| 3 | 30.4 | 30.4 | 17.6% | 1,414 | 7,520 |
| 4 | 31.0 | 30.9 | 22.9% | 1,401 | 7,760 |
| 5 | 31.3 | 31.2 | 26.5% | 1,394 | 8,000 |
| mean | 30.4 | 30.4 | | | |

*Table 9.2: KPIs for the six variants in the small-scale scenario, showing for various choices for the maximum number of opened daytime maintenance locations the average number of activities per day, the average number of hours of activity per day, the percentage of hours of activity that are performed during the day, and the associated costs according to the optimistic method and the conservative method, defined in Section 9.2.4, in €.*

The average number of activities and the average hours of activity are both equal to 30.4. These numbers are expected, as can be seen in Table 9.3: it calculates the expected numbers of numbers of activities and hours of activity, based on the number of rolling stock units and number of days in the analysis. The expected numbers would have been 30.9 and 20.6, respectively. The expected numbers are very close to the actual numbers, supporting the correctness of the **MSLCP** model.

| | # activities | | | # hours of activity | | |
|---|---|---|---|---|---|---|
| type | per rsu | total | avg. per day | per rsu | total | avg. per day |
| A | 6 | 144 | 20.6 | 4.5 | 108 | 15.4 |
| B | 4.5 | 72 | 10.3 | 3 | 108 | 15.4 |
| | | | 30.9 | | | 30.8 |

*Table 9.3: For both maintenance types, the expected number of activities and number of hours of activity, per rolling stock unit, in total over the time horizon, and averaged per day. The number of hours of activity are expected based on the time horizon of 7 days and the intervals of 1 and 2 days for type A and B, respectively. The total number of activities can be achieved by multiplying this number by 24 (the total number of rolling stock units). Dividing this number by 7 yields the average amount of activities per day. The number of hours of activity can be obtained by multiplying the number of activities of type A by 0.5 and of type B by 1 (the respective maintenance durations). The average number of activities and the average number of hours of activity for both types (30.9 and and 30.8, respectively) are approximately equal to the numbers presented in Table 9.2.*

The method to compute the costs has been described in Section 9.2.4: recall that the optimistic costs are proportional to the hours of activity, and that the conservative method calculates costs per eight hours of activity. Observe that, due to a more daytime maintenance location and the subsequent increase in the number of daytime maintenance activities, the optimistic costs decrease. At the same time, the conservative costs increase. Apparently, the opening of daytime maintenance locations has induced the deployment of new maintenance teams during daytime, incurring costs, but this has not led to an equally large decrease in the number of maintenance teams necessary during nighttime.

In addition, note that the average day share is equal to 26.5 % in the situation with 5 daytime maintenance locations, meaning that over a quarter of all maintenance activities in the considered scenario can be performed during daytime.

The distribution of the workload during daytime over the various opened locations is given in Figure 9.5.

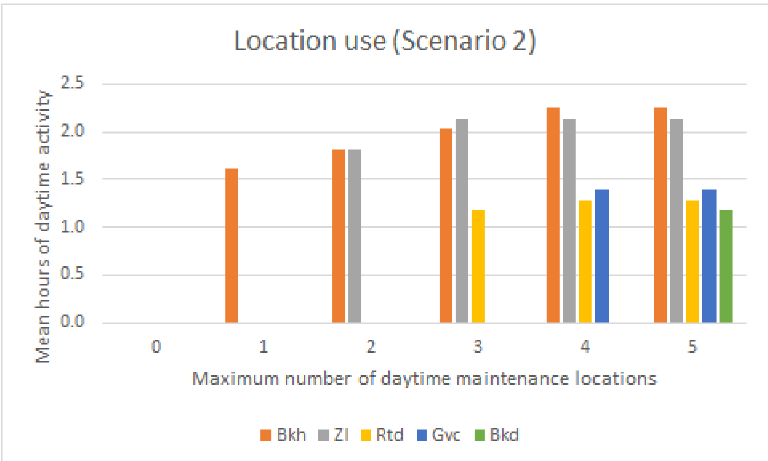



*Figure 9.5: Hours of activity per location, averaged per day, for various values of the maximum number of locations for daytime maintenance, in the small-scale scenario.*

At maintenance locations Bkh and Zl a workload of over 1.5 hours average per day can be experienced. When opening a third location, in this case Rtd, this not only results in workload assigned to Rtd, but also to extra workload assigned to Bkh and Zl. The new possibility of daytime maintenance in Rtd also enables more daytime maintenance in locations that were already open. This demonstrates the network-effect of the railway logistics: enhancements at one location can induce enhancements at other locations.

### 9.3.2 Large-scale scenario

Above, the small-scale version of the use case is used. This version includes only a limited number of rolling stock units. However, in practical situations, a larger number of rolling stock units prevails. Therefore, the current section investigates the results for the large-scale version of the use case.

A first observation was is made is that, a priori, the large scale version of the use case does not yield a feasible solution, since for one rolling stock unit no feasible maintenance schedule could be made. This rolling stock unit has been excluded from the analysis to be able to still provide results. This matter is elaborated upon in Appendix D.3.

The optimal location choice in the current scenario is given in Table 9.4.

| # locations | Bkd | Bkh | Dv | Rtd | Llso |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | x | 0 | 0 | 0 | 0 |
| 2 | x | x | 0 | 0 | 0 |
| 3 | x | x | x | 0 | 0 |
| 4 | x | x | x | x | 0 |
| 5 | x | x | x | x | x |

*Table 9.4: Locations opened in the large-scale scenario for a varying number of maximum number of daytime maintenance locations that can be opened.*

It appears that, although Bkh and Rtd are still part of the solution, the location choice is different from those obtained in the small-scale scenario. Now, Bkd seems to be a very good location to choose for daytime maintenance. This location did not excel in the former scenario. Note that in the large-scale version of the use case, many more rolling stock units are added to the analysis: it may entail many rolling stock units that are having maintenance opportunities in Bkd (whereas these rolling stock units were not yet part of the earlier analysis). As such it can be explained that the location choice is different.

The most important KPIs are presented in Table 9.5.

| | | | | costs (€) | |
|---|---|---|---|---|---|
| # locations | # activities | hrs. of activity | day share | optim. | cons. |
| 0 | 427.4 | 428.7 | 0.0% | 21,434 | 31,600 |
| 1 | 431.0 | 432.4 | 4.8% | 21,204 | 31,520 |
| 2 | 433.0 | 434.5 | 7.8% | 21,046 | 31,600 |
| 3 | 436.6 | 438.2 | 10.8% | 20,966 | 31,680 |
| 4 | 438.3 | 439.8 | 12.7% | 20,873 | 30,720 |
| 5 | 440.6 | 442.0 | 14.7% | 20,798 | 30,800 |
| mean | 434.5 | 435.9 | | | |

*Table 9.5: KPIs for the six variants in the large-scale use case scenario. It shows for various choices for the maximum number of opened daytime maintenance locations the average number of activities per day, the average number of hours of activity per day, the percentage of hours of activity that are performed during the day, and the associated costs according to the optimistic method and the conservative method, defined in Section 9.2.4, in €.*

The day share appears to be lower than in the small-scale scenario. The large-scale version of the use case not only adds new opportunities for daytime maintenance, it also adds many rolling stock units for which daytime maintenance is not regularly possible, driving down the share of activities that can be performed during the night. This may depend on the characteristics of the railway line on which the rolling stock units are deployed. For example on high-frequent railway lines, less maintenance opportunities of sufficient length may be present since rolling stock units arriving at a station can be deployed for a returning line quickly. Similarly, on railway lines with a less significant difference between peak demand and off-peak demand, less rolling stock units may need to be taken out of service after the morning peak, leading to less maintenance opportunities.

Furthermore, it can be noted that the costs according to the conservative method do not increase when more maintenance locations are opened. See Section 9.2.4 for a detailed description on how these costs are computed. Interestingly, the current result is opposite to the result found in the small-scale scenario, where the conservative costs increase when more work is performed during daytime. The fact that in this scenario the costs according to the conservative method do not drop, shows that due to the higher number of rolling stock units in the analysis, substantial workloads are assigned to maintenance locations so that employees can be provided with sufficient amounts of work. The employees hired during daytime can therefore be more efficiently deployed, which becomes visible in the cost estimates.

Lastly, the workload over various locations is presented in Figure 9.6.

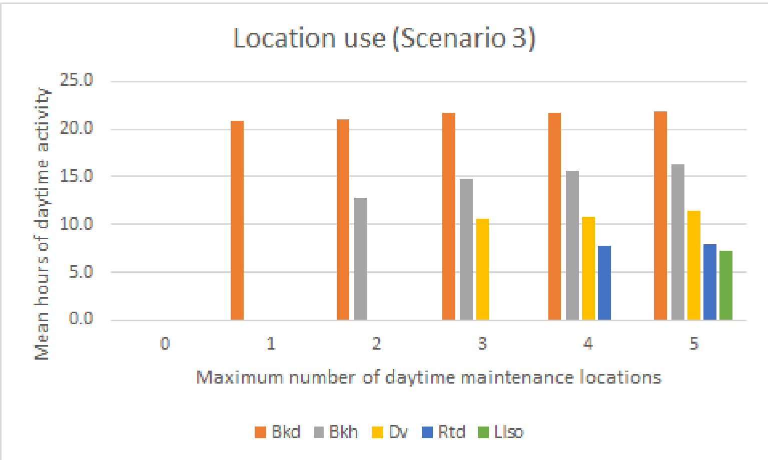



*Figure 9.6: Hours of activity per location, averaged per day, for various values of the maximum number of locations for daytime maintenance, in the large-scale scenario.*

This figure shows that, unlike in the small-scale version of the use case, a substantial amount of work seems to be assigned to maintenance locations, especially to Bkd. At the location of Bkd, more than 20 hours of work can be performed on average. When also Bkh and Dv are opened, the average workload assigned to these locations exceeds 10 hours. These numbers indicate that it may become worthwhile to station a staff team at these locations.

## 9.4 Maintenance location capacity

So far, capacity of maintenance locations had not been incorporated in the framework. However, it is often relevant to assess the capacity requirements of a given maintenance schedule. To this end, the current research introduces the **APP** tool, which can be used to assess the capacity of any solution that is produced by the **MSLCP** tool. It uses a given schedule of the maintenance activities (i.e. the assignment of maintenance activities to maintenance opportunities). Its benefit is twofold: on the one hand, it is able to create a feasible maintenance shift planning for maintenance teams. On the other hand, it is able to determine the minimum number of teams necessary.

The current section does not compute the capacity for the previous scenarios, but presents another example to demonstrate the workings of this tool.

Figure 9.7 presents a possible maintenance schedule for one day, for multiple rolling stock units. It is created based on the small-scale version of the use case and considers short maintenance durations. Clearly, all maintenance activities (indicated by blue rectangles) can be carried out sequentially by one team. The **APP** can be run to create a feasible schedule for those activities that need to be performed during daytime. The resulting schedule is given in Table 9.6.

| | release | deadline | team | start | end |
|---|---|---|---|---|---|
| 1 | 9:21 | 10:10 | 1 | 9:21 | 9:51 |
| 2 | 10:56 | 11:33 | 1 | 10:56 | 11:26 |
| 3 | 11:56 | 12:33 | 1 | 11:56 | 12:26 |
| 4 | 15:21 | 16:10 | 1 | 15:21 | 15:51 |
| 5 | 16:21 | 17:10 | 1 | 16:21 | 16:51 |

*Table 9.6: Activity planning in Gvc on 13-6-2019.*

It shows that all maintenance activities can be carried out sequentially, all by the same team.

However, now consider the schedule from Figure 9.8. Clearly, some maintenance activities during daytime are overlapping: see for example the maintenance activities for the first and second rolling stock unit. These two maintenance activities cannot be performed by the same team. Table 9.7 indicates the maintenance team planning that can be made for the Schedule in Figure 9.8.

| | release | deadline | team | start | end |
|---|---|---|---|---|---|
| 2 | 8:26 | 9:03 | 1 | 8:33 | 9:03 |
| 3 | 9:21 | 10:10 | 1 | 9:40 | 10:10 |
| 4 | 10:56 | 11:33 | 1 | 11:03 | 11:33 |
| 5 | 11:56 | 12:33 | 1 | 11:56 | 12:26 |
| 6 | 13:56 | 14:33 | 1 | 13:56 | 14:26 |
| 8 | 14:56 | 15:33 | 1 | 15:03 | 15:33 |
| 9 | 15:21 | 16:10 | 1 | 15:40 | 16:10 |
| 10 | 17:26 | 18:03 | 1 | 17:33 | 18:03 |
| 1 | 8:21 | 9:10 | 2 | 8:40 | 9:10 |
| 7 | 13:56 | 14:33 | 2 | 13:56 | 14:26 |

*Table 9.7: Activity planning in Gvc on 12-6-2019.*

Clearly, the maintenance activities for rolling stock unit 1 and 7 overlap with an other maintenance activity. Hence, these maintenance activities need to be performed by different teams, which is also indicated.

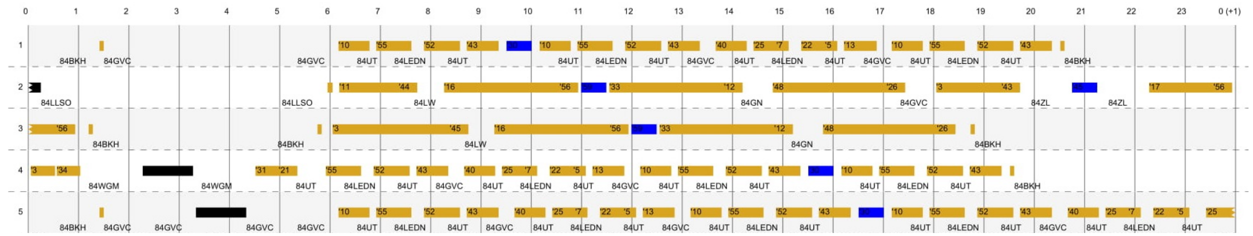


*Figure 9.7: Schedule in Gvc on 13-6-2020. Each row corresponds to the rolling stock circulation of one rolling stock unit.*

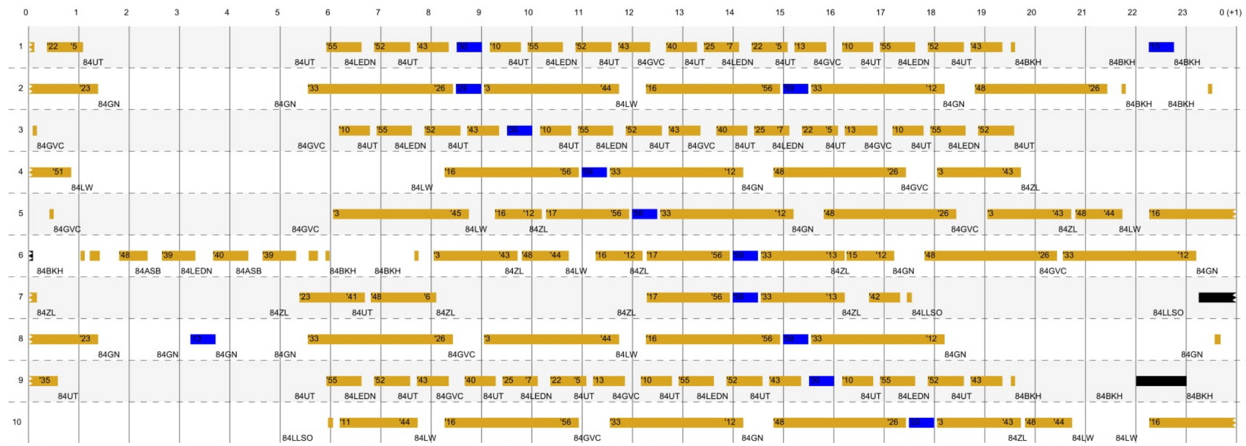


*Figure 9.8: Schedule in Gvc on 12-6-2020. Each row corresponds to the rolling stock circulation of one rolling stock unit.*

## 9.5 Use case discussion

This section addresses three aspects. First, it elaborates on the benefits of and future developments on the current use case in Section 9.5.1. Then, Section 9.5.2 discusses how Viriato was used in the current research. Section 9.5.3 indicates how the setup of the current chapter differs from the setup used in Chapter 7.

### 9.5.1 Benefits and future developments

Below, the benefits of the approach taken in the current use case are discussed and directions for future developments are identified.

#### Benefits

The benefit of a small use case is that it helps to understand the capabilities of the **MSLCP** tool and to communicate a general methodology that can be applied to solve these kind of problems. This use case is beneficial in at least hree ways. First, several visualizations of solutions can be presented, demonstrating that the solutions generated are viable. Second, for several input parameters, such as the maximum number of daytime maintenance locations, it can be shown that they can be varied and their impacts can be determined. Third, the type of results that are important to assess the solution can be demonstrated (i.e. the opened locations, the number of activities and hours of activity, the day share, the costs and the distribution of workload over various locations).

The following key observations can be made based on the current use case.

- A feasible schedule can be created for various choices of the maximum number of daytime maintenance locations. Moreover, it is possible to create a feasible schedule with no daytime maintenance locations at all. It appears that all maintenance activities are performed during the night (as expected).
- A method is provided to approach the problem NS is facing: the **MSLCP** model can be used to determine optimal maintenance locations for practical use cases, and it is shown that this allows the user to compute several important KPIs: the hours of activity and number of activities, the percentage of work performed during daytime, costs, and the spread of work over the various maintenance locations.
- Results can be produced for a small use case of the 24 rolling stock units of type *DDZ4*. In some cases, already this small subset is relevant. For example, when making the shift to daytime maintenance, it is desirable to start on a smaller scale first, to mitigate the risks that inherently come with any policy shift. The current analysis provides good insight in the expected effects and benefits of this approach if NS would decide to start a potential shift to daytime maintenance with rolling stock units of type *DDZ4* only.
- The day share decrease in the large-scale version of the use case compared to the small-scale version of the use case. This means that the results are sensitive to the rolling stock types included in the analysis: in the current case, the results for the small case are too optimistic compared to the large-scale case. The choice of rolling stock units for the small-scale version and the large-scale version resulted in relatively more daytime maintenance activities in the small-scale version compared to the large-scale version. This uncovers the risk of considering only a small subset of rolling stock units: the results may be very different in a larger context.

- In the large-scale situation, up to 14.7 % of all hours of activity can be performed during daytime, and to some maintenance locations a workload of up to 20 hours of activity can be assigned, making daytime maintenance an attractive option.

**Future developments**

There is also a number of aspects for which further investigation is relevant or necessary.

- The opened locations vary between the small-scale and the large-scale scenario. This raises questions regarding the robustness of the location choice. It is therefore advised that more results are obtained regarding the location choice for various input settings.

- The cost estimations are currently very rough. In the first place, the cost values were not disclosed by NS and therefore estimates were made. Moreover, the cost dynamics are more complex in reality than displayed here. The current approach has addressed these dynamics by calculating the costs in a conservative and an optimistic way, but these complex cost determinants need to be addressed further to solidify any statements regarding costs.

- Also, currently the costs were only taken into account after generation of the result. This however means that the costs are not necessarily minimal. If NS would like to take a more cost-oriented approach (instead of an approach oriented on minimizing the number of nighttime maintenance activities), adaptations to the current approach need to be made. Especially, the objective of the **MSLCP** model needs to be adjusted in that case.

- Some maintenance type durations are dependent on the number of staff available. For example, a cleaning task requires less minutes if multiple cleaning staff is available. If the maintenance duration is shorter, potentially more work can be performed during daytime since the necessary maintenance opportunity length is shorter. In other words, more maintenance opportunities can potentially be used for daytime maintenance. However, this also comes at the cost of needing to hire extra personnel. The relation between the benefits of shorter maintenance on the one hand and the costs of the extra personnel associated to it on the other hand is not addressed in the current research, but nevertheless very relevant. It is advised to look more into this matter to be able to draw more stable conclusions on the costs and benefits of any solution.

- Currently, the workload spread over the various locations has only been investigated on an aggregate level. In other words, only average workloads were considered. The workload assigned to any maintenance location, however, varies from day to day. It is interesting to also investigate further these day effects. For instance, it can then be disclosed whether the workload is constant over the days or whether specific peaks are visible over specific days. In practice, this knowledge is important. For instance, if the workload at a specific location is high only at a specific day of the week, then it may be better to only station personnel at this location on this specific day, and not on the other days. Moreover, the workload that may be experienced on such a day may be much higher than the currently reported average (since within this average, also the potentially low workloads on other days are included). If, on the other hand, the workload is fairly constant over the days, then this may have the practical implication that it is beneficial to station personnel at this location on all days of the week.

- The current **MSLCP** tool does not offer the opportunity to restrict capacity of maintenance locations. Nonetheless, capacity of maintenance locations may be a restricting factor. The current research provides first indications on how capacity of maintenance locations can be incorporated in the framework of the **MSLCP** in its design of the **CMSLCP**. More research is required to improve the running times of the **CMSLCP** in order to be able to also incorporate the capacity of maintenance locations in finding an optimal maintenance schedule and an optimal maintenance location choice quickly in large-scale instances.

### 9.5.2 Application of Viriato

In order to provide many of the results in the current chapter, Viriato has been used. Viriato is a software program developed by *SMA und Partner A.G.*, headquartered in Zürich. It aims to provide a comprehensive planning tool for railway industry that can be used by railway operators, authorities and infrastructure managers to optimise railway planning (SMA, 2016).

Viriato is aimed at conceptual planning, service planning and capacity planning, that is, it focuses mainly on the macroscopic and mesoscopic level. It does not intend to solve problems on the microscopic level (with, for example, microscopic simulations) (SMA, 2020c).

The software is structured in three main components (SMA, 2020c).

1. Base data, including infrastructure information, rolling stock specifications and a calendar specifying for example holidays

2. Trains, specifying individual trips and collections of trips, which can be organized and grouped in various ways

3. Timetables and vehicle rosters with various opportunities for visualization

Once the trains are given, these can be combined into a roster (rostering interface). Then this roster can be optimized using an external model, which has been done in the current case. The way in which algorithms can be coupled to Viriato is displayed graphically in Figure 9.9. First, the user specifies the required input in the Viriato Graphical User Interface (GUI) and starts the algorithm from this same GUI. Then, Viriato writes the problem to a file (called *problem file*) and invokes the external optimizer. This optimizer may be any script in any programming language. The optimizer reads the problem file, computes a solution, and writes a *solution file*, after which it terminates. Once the script terminates, Viriato looks for the solution file on the path specified in the Viriato GUI and visualizes the results in the GUI. The user can then benefit from the solution provided by the external algorithm.

#### Utilization of Viriato in the use case

The problem that NS is facing was solved using the **MSLCP** model in combination with Viriato. To this end, the input for the **MSCLP** model needs to be imported into Viriato. The following steps were taken. First, the nodes from the NS network were imported. Then, trains were imported according to NS BDu data for period 10 (see Table 7.1). These trains were connected into a valid roster according to the links of the NS BDu data. As a result, the trains appear in the so-called *rostering interface* as linked activities. These linked activities specify the planned train path for one rolling stock unit over multiple days. In addition, the maintenance type definitions were set in Viriato.

This specifies most of the required information to run the **MSLCP** model. Two parameters could not be set within Viriato and were set in the external script with the implementation of the **MSLCP** model:

- The number of daytime locations

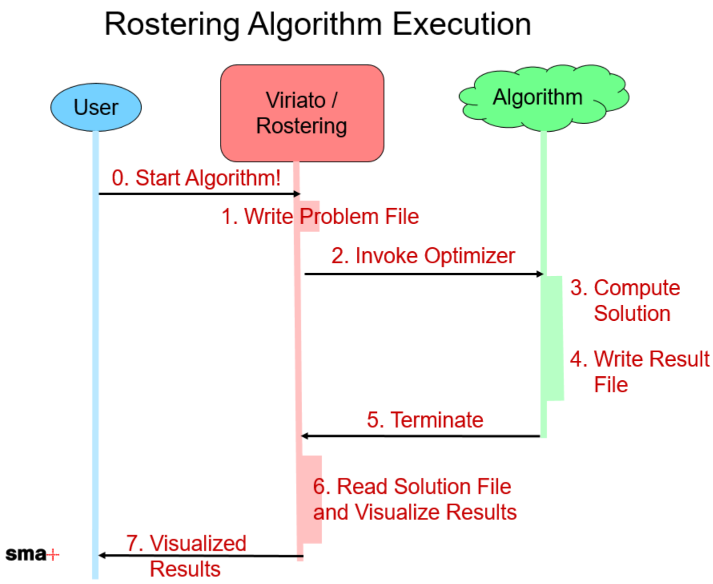



*Figure 9.9: Graphical representation of the way an external algorithm (such as the **MSLCP**) communicates with Viriato (SMA, 2020a).*

- The start of the daytime time window and the start of the nighttime time window

Section 3.2 mentioned the main assumptions of the **MSLCP**. Some of these assumptions were classified as *input*, meaning they can be specified by the user. In Table 9.9 these assumptions are listed and classified into one of three categories. Any of these assumptions can be either (1) influenced in Viriato by the user, (2) influenced in the script by the user, although this requires some specific skills, or (3) hard-coded, meaning influencing these parameters requires more extensive knowledge of the model. In case one would want to go beyond the scope of the **MSLCP**, the implementation can be altered so that the hard-coded parameters become variable as well, but in the current implementation this is not yet possible.

| | parameter | Viriato param. | code param. | hard-coded |
|---|---|---|---|---|
| 1. | Rolling stock circulation. | | | |
| | *- One BDu is considered (period 10)* | x | | |
| | *- The set of rolling stock units of type DDZ4 are considered* | x | | |
| | *- The planning horizon is 7 days* | x | | |
| 2. | Set of nighttime maintenance locations | | | |
| | *- Assumed to be equal to set of all locations in the analysis* | | | x |
| 4. | Set of potential daytime maintenance locations | | | |
| | *- Assumed to be equal to set of all locations in the analysis* | | | x |
| 5. | Maximum number of daytime locations that can be opened. | | | |
| | *- Varied by a parameter in the script* | | x | |
| 6. | Maintenance types | x | | |
| | *- Two types are considered: Type A with a duration of 30 minutes and an interval of 24 hours, and Type B with a duration of 60 minutes and an interval of 48 hours* | x | | |
| 9. | Initial conditions | | | |
| | *- Assumed that all trains are as-good-as-new at the start of the planning horizon (i.e. $b_{ik} = 0$ for all $i \in I, k \in K$)* | | | x |
| 16. | The start of the daytime time window and the start of the nighttime time window | | x | |
| | *The start of the daytime window is set at 10.00 and the start of the nighttime window is set at 19.00.* | | x | |

*Table 9.9: Parameters in the Viriato implementation of the* ***MSLCP*** *model. The ids of these parameters, given in the left column, correspond to those assumptions in Table 3.3 which are classified "input", meaning they can be chosen by the user of the* ***MSLCP****. For these parameters, the current table indicates the ease with which each parametre can be influenced in the current implementation. The column "Viriato param." indicates whether the parameter can be influenced through the Viriato interface, the column "code param." indicates whether the parameter is to be set by changing a value in the underlying script, and the column "hard-coded" indicates that the parameter setting is hard-coded.*

### 9.5.3 Relation to other MSLCP results

Chapter 7 and the current chapter both provide results based on the **MSLCP** model. The difference is that the approach Chapter 7 is aimed at providing insights in the workings of the model for many different scenarios, while the current chapter starts with a particular use case that may show up in practice and demonstrates how the **MSLCP** can be used to address this case.

To this end, some assumptions in the two sections differ, which may also explain differences in results.

- The small scale version in the current chapter uses all rolling stock units of type DDZ4, since this enabled to easily provide insightful visualizations. By contrast, the smaller cases in Chapter 7 (in scenario batch 1 and 2) used all rolling stock types of types VIRM4 and VIRM6. On the other hand, the large-scale versions in Chapter 7 and the current chapter are better comparable: both consider all rolling stock units used for intercity services.

- In general, Chapter 7 uses maintenance durations of 30 minutes for Type A and 60 minutes for Type B. In the current chapter, however, it was identified that longer maintenance durations of 45 minutes and 90 minutes, respectively, may be better suitable and were therefore used. This leads to different results.

It must be noted that the initial maintenance durations resemble the maintenance durations that are actually used by NS, and are therefore not incorrect. The fact that the current chapter uses longer maintenance durations can be considered a more conservative approach, which can be used to deliver results that are more robust during operations since more slack time is incorporated.

- The current chapter provides results for no more than 5 maintenance locations for daytime maintenance, whereas Chapter 7 investigates scenarios up to 20 maintenance locations. The current chapter aims to stay as close to practical situations as possible. Since, on the short term, it is not likely that more than 5 locations are opened for daytime maintenance, only these scenarios have been investigated that consider 5 or less daytime maintenance locations.

# Chapter 10

# Discussion

The present research addresses the problem faced by NS of increasing pressure on available capacity of rolling stock maintenance locations. To this end, it defines three problems and corresponding models to solve these problems. First, at the core of the research lies a model for the *Maintenance Scheduling and Location Choice Problem* (**MSLCP**), which takes as its main input a rolling stock circulation and provides an optimal maintenance location choice and an optimal maintenance schedule. Second, to assess the capacity of any **MSLCP** solution, the *Activity Planning Problem* (**APP**) is presented. The corresponding **APP** model takes as input a maintenance schedule delivered by the **MSLCP** and provides an optimal maintenance shift planning and the corresponding required minimum capacity. Third, the *Capacitated Maintenance Scheduling and Location Choice Problem* (**CMSLCP**) aims to provide an **MSLCP** solution that includes capacity of maintenance locations. The provided **CMSLCP** model integrates the **MSLCP** and the **APP**, finding a solution to the **MSLCP** that satisfies some predetermined capacity constraints.

The current chapter critically discusses the present research and is divided into three parts. Section 10.1 positions the current research in a broader perspective and Section 10.2 discusses its limitations. Section 10.3 discusses the usefulness of Viriato in addressing research problems like the present one.

## 10.1 A broader perspective

The current section aims to discuss the relevance of the current research by assuming a broader perspective. It starts by discussing the relevance of the current research by discussing the relevance of each of the three models considered in the current research. Then it discusses for which types of decisions it should be used by arguing it is mainly intended for decisions on the tactical level. It continues by discussing the application of the current research in other contexts. It concludes by giving two other measures that may help to improve the goal of NS to reduce capacity pressure during nighttime and how the current research relates to those measures.

### Relevance

As indicated, the current research produces three cooperating models that contribute to solving the problem faced by NS: the **MSLCP** model, the **APP** model and the **CMSLCP** model. Their development is relevant both scientifically and in practice.

The **MSLCP** contributes to literature as it simultaneously optimizes the (rolling stock) maintenance location choice and the (rolling stock) maintenance schedule. In addition it considers two time windows for maintenance (daytime and nighttime) which are not equally desirable. Moreover, it is relevant in practice as it offers railway operators the opportunity to create an optimal maintenance schedule automatically where this is currently often done manually and hence requires a lot of resources. Such a maintenance schedule can not only obtained automatically, but also relatively quickly: the current research has shown that solutions to the **MSLCP** can be achieved quickly (computation times are less than several hours for practical real-life instance sizes). Furthermore, **MSLCP** offers railway operators a method to determine the optimal maintenance locations for a given a rolling stock circulation. This would then correspond to their choice of stationing maintenance personnel at locations that rolling stock units visit or not.

The developed **APP** model provides an optimal maintenance shift planning and also provide the required capacity for any solution of the **MSLCP**. The **APP** is a variant of the well-known *Parallel Machine Scheduling Problem* (see, for example, Kravchenko and Werner (2009)). Its application in the rolling stock maintenance scheduling context is a contribution to the literature. In addition, it is very relevant in practice. Usually, based on the set of rolling stock units that visit a maintenance location during a maintenance shift and the maintenance activities that need to be performed, an optimal maintenance team planning needs to be created. The **APP** delivers such a planning. For realistic numbers of rolling stock units on maintenance locations, the generation of such a schedule is quick (a few seconds) and it can therefore be used in operations. Moreover, the **APP** gives valuable insight in the number of resources (maintenance teams) necessary for a given **MSLCP** solution.

The **CMSLCP** integrates the **MSLCP** and **APP** using a technique called Logic-Based Benders Decomposition (LBBD). Its goal is to find a solution to the **MSLCP** that satisfies some predetermined capacity constraints. The potential relevance of this method is high, since the addition of maintenance location capacity constraints to the **MSLCP** adds considerably to the realism and usefulness of the provided solution. However, the running times of the current implementation of this framework in order to find may still be too high at present for commercial purposes. Yet, from a scientific point of view, the **CMSCLP** model is utterly relevant. First, it gives a feasible approach of incorporating complex capacity constraints to the **MSLCP**. Second, it proposes an application of the LBBD in the context of rolling stock maintenance scheduling. Third, it opens up many interesting research areas on the improvement of the algorithm. In particular, the design of new cut generation processes to produce more efficient (i.e. more general) cuts is interesting.

**Current application**

Planning problems can be categorized into strategic, tactical and operational problems. The strategic level refers to decisions several years before operation with large amount of freedom of choice (such as the choice on which maintenance locations to build); the tactical level refers to decisions up to months in advance when the main conditions (such as the set of maintenance locations) are fixed, but when the assignment of resources is still variable (such as the allocation of personnel); the operational level refers to decisions close to operations (such as the dispatching of personnel to maintenance jobs). The problem at hand would be characterized mainly as a tactical planning problem. It requires inputs about, for instance, the set of maintenance locations available and the planned rolling stock circulation (therefore not being truly strategic), and on the other hand provides decision support for the maintenance processes to be carried out over the period of some weeks, rather than for the maintenance processes today and tomorrow (therefore not being truly operational).

The tactical aspect of the **MSLCP** and **CMSLCP** is mainly visible in the location choice aspect. This aspect should be considered as a tactical decision in the current research, since it is based on a rolling stock circulation (BDu) for a period of approximately eight weeks. The location choice provided by the **MSLCP** and **CMSLCP** models is only guaranteed to be optimal for this period. The maintenance locations are assumed to be available (i.e. they do not need to be built in order to be used), and the location choice should be interpreted as the railway operator's decision to allocate personnel at this location or not. As the **APP** is used to assess the capacity requirements for any **MSLCP** solution, it is used on a tactical level as well.

Yet, the current research can also serve strategic goals. The **MSLCP** and **CMSLCP** can for example be used to choose where a new maintenance location should be built. For this, however, it would be necessary to analyse and compare the optimal maintenance locations of the rolling stock circulation of various periods and compare the results. If the location choice from the **MSLCP** is robust over various input periods, this offers evidence that the location choice is also valid for strategic decisions.

Moreover, parts of the current research can also be used for operational purpose. First and foremost, the capability of the **APP** to provide a maintenance shift planning makes it very applicable for operational goals. For example, it could be used by planners to determine for a specific maintenance shift which maintenance team performs which maintenance job. Its quick running times are a large benefit. Second, also the **MSLCP** can prove useful during operations: for example, it can be run again after disruptions in the maintenance schedule, to determine an updated maintenance schedule. In that case it would be important to fix the set of daytime maintenance locations, since it can usually not be adjusted in the operational phase.

#### Other applications

In the current research, the **MSLCP** (and also the **APP** and **CMSLCP**) were designed in the context of railway operations, and more specifically, they were tailored to the problem of maintenance location capacity problems during nighttime that NS is currently facing. These models may however also be applicable in other railway operations contexts, or even in other (research) fields.

First, it should be noted that the current research has focused on the Dutch situation, and more specifically on the problem NS is facing. However, the current research could also be very applicable to other railway operators. This holds certainly for the **APP**, since the maintenance shift planning based on the arrival time and departure time of rolling stock units at a maintenance location is deemed relevant in all railway contexts. However, it holds also to a large extent the **MSLCP** and **CMSLCP**. In general, the issue of maintenance scheduling is universal. Moreover, the maintenance location choice aspect is usually relevant as well, since most railway operators have the choice to allocate personnel at maintenance locations or not. An aspect of the currently considered problem that need not be universal, however, is the formulation of the problem in the sense that capacity issues are present during nighttime and more daytime maintenance needs to be considered.

To address different goals that may apply to other railway operators and to increase the versatility of the models designed in the current research, alternative formulations for the **MSLCP** may be considered. A more general approach to the problem can, for example, be achieved by considering an objective function that includes costs. This may shift the current perspective, which without exception gives preference to daytime maintenance activities, to a more neutral perspective towards daytime maintenance and nighttime maintenance. It enables to include more complex cost structures, such as location-specific costs, or piece-wise linear cost functions to reflect economies of scale.

However, the current research may also be applicable in different fields. To this end, it is useful to compare a formulation of the framework of the problem considered in the current research to a more general, but equivalent, formulation. The framework of the current research can be formulated as follows: *rolling stock units, moving over a railway network, have maintenance opportunities, defined by their arrival an departure time at maintenance locations; a minimum number of maintenance activities needs to be assigned to these maintenance opportunities such that the interval sizes between maintenance activities do not exceed some maximum value; maintenance opportunities can be either during daytime or during nighttime, where maintenance opportunities during daytime are more desirable than maintenance opportunities during nighttime.* A more general, but equivalent formulation, would be: *moving units, moving over a given network, have activity opportunities defined by some start and end time at given locations; a minimum number of activities needs to be assigned to these activity opportunities such that the interval sizes between activities do not exceed some maximum value; activity opportunities can be either of class 1 or of class 2, activity opportunities of class 1 being more desirable than activity opportunities of class 2.*

This shows that the current research could in fact be applied in any context where moving units move over a network and activities need to be performed on these moving units. It may for example apply to the aviation industry, where airlines operate according to a given schedule and where the activity opportunities would relate to the intervals when aircraft are not flying. It may also be applied to the delivery industry, where each vehicle operates according to a schedule and the intervals during which it is not used (i.e. it is idle) can be used to perform activities to the vehicle. These activities are in many cases maintenance activities (although they need not necessarily be), since it relates to activities that need to be performed for every moving unit, within a maximum interval, minimizing the number of activities - which is typical for maintenance activities. Also, it shows that the activity opportunities need not be separated in daytime and nighttime maintenance activities, but in fact any separation into two activity classes is viable.

Even more applications outside the railway sector can be identified by observing that the **MSLCP** may be viewed as a generalization of other, more specific problems. For example, assume that its location aspect is ignored. Then the **MSLCP** reduces to the more standard scheduling problem of assigning a minimum number of activities to activity opportunities in such a way that interval constraints between maintenance activities are satisfied. A potential application of this may be the scheduling of the cleaning or maintenance of operating rooms in a hospital, given the planning of operations, such that the intervals between these activities do not exceed some maximum.

The design of the **CMSLCP** also introduces some interesting applications of the LBBD framework in scheduling contexts. In the current research it is used to incorporate the set of maintenance location capacity constraints in the **MSLCP**. These constraints can be referred to as complicating constraints, as it is not straightforward to determine whether the constraint is satisfied or violated: this requires a separate sub process, which is in the current case the **APP**. The outcome of this sub process restricts certain combinations of activities, that result in a violation of the capacity constraints, to be scheduled. This uncovers a significant potential application in other fields as well: the LBBD framework of **CMSLCP** model can be used for any scheduling problem with complicating constraints that may result in a restriction of certain combinations of activities to be scheduled. An example of such an application may be the scheduling of maintenance activities to a fleet of delivery vans, subject to some complicating constraints that govern the routing of the maintenance team between delivery vans.

An important limiting factor to take into account when regarding the applicability of the current research in other fields is the fact that it assumes a given rolling stock circulation. This assumption is at the root of the current research, and it has offered the opportunity to design a computationally tractable model to optimally schedule maintenance activities and determine an optimal maintenance location choice. However, this also means that the timing of the maintenance opportunities is fixed. There exist some applications in which exactly this assumption is problematic. For example, in the scheduling of heavy maintenance in railway operations, concerning maintenance activities with longer durations (up to several days), the rolling stock circulation is often adjusted in order to create a maintenance opportunity for heavy maintenance. The **MSLCP** does not offer this flexibility. As a result, the **MSLCP** is not applicable to heavy maintenance activities. Also in other fields, this problem applies. For example in the area of aviation, where aircraft are taken out of the circulation to undergo larger maintenance, the aircraft rotations are adjusted in order to create the opportunities for maintenance. This is not incorporated in the current research.

**Other measures to solve the NS problem**

The goal of NS is to reduce the capacity pressure of maintenance locations during nighttime by performing more activities during daytime. The current research ahs contribute to that by providing an optimal schedule and a location choice that reduces the pressure during nighttime to the farthest extent possible. It takes as input a given rolling stock circulation and a given set maintenance types that need to be performed. However, reductions in nighttime capacity pressure may be achieved by adapting the rolling stock circulation or the maintenance type definitions.

The rolling stock circulation contains, in the case of NS, many long maintenance opportunities that start just after the morning peak and end just before the evening peak, reflecting the fact that not all rolling stock units are necessary during off-peak hours. The resulting time available for maintenance (up to 8 hours) is usually much longer than the time required for maintenance (often no more than 2 hours). Therefore, Zomer (2019) and Van Hövell (2019) investigated opportunities to exchange rolling stock units after maintenance for rolling stock units that have not yet been maintained. More specifically, they consider to start maintenance immediately at the moment when a rolling stock unit arrives at the service location, and after maintenance exchange this maintained rolling stock unit for a not-yet-maintained rolling stock unit that is currently in service. In this way, the number of rolling stock units that can be maintained during daytime increases. This potentially leads to the possibility to maintain more rolling stock units during daytime and hence achieve the objective of NS (and hence the objective of the **MSLCP**) even better.

The incorporation of exchange opportunities cannot be implemented in the current model: the model should then take into account that this exchange propagates through the entire future cycle of the rolling stock planning. This would add heavily to the complexity of the model and is therefore beyond the scope of the current research. To still, somehow, account for this promising technique to increase the objectives of NS, one may consider to incorporate exchange opportunities in the inputs. The BDu should then be adjusted in the pre-processing phase so that it includes additional exhange opportunities. In this way, exchange opportunities still do not become decision variables and hence not part of the optimization, but they are considered alternative scenarios and can be analysed as such.

Apart from adapting the rolling stock circulation, also the maintenance type definitions may be adjusted to achieve more capacity reduction during nighttime. This can be the result of technical innovations. An example of such innovations are the ideas related to maintenance operations using video cameras, which are expected to lead to shorter maintenance durations. Another example is the trend where the condition of the rolling stock units is monitored remotely (cf. for example Mooren Ceng and Van Dongen (2013)), which may lead to longer maintenance intervals.

## 10.2 Limitations

The following limitations of the current research should be taken into account when interpreting its results.

**Influence of disruptions**

The most important limitation is that the current study has used data on planned rolling stock circulations (called BDu data). This data is made available multiple weeks before the start of the corresponding period and it covers a period of approximately eight weeks. It enables to specify the planned path for each rolling stock unit. The current research assumes that this path is given for the entire period of eight weeks. Based on this data, the **MSLCP** determines an optimal maintenance schedule (such that for each rolling stock unit, the intervals between maintenance activities satisfy some constraint) and an optimal location choice.

However, it is known that during operations, due to disruptions of all kinds, many changes are made to the planned rolling stock circulation. For example, consider one rolling stock unit going from A to B and another going from B to A. If a blockage occurs between A and B, the first rolling stock unit usually returns to A and the second usually returns to B. The first rolling stock unit takes over the role of the second and vice versa, which is good from a service point of view since it enables the railway operator to satisfy all train trips planned for the first rolling stock unit by the second rolling stock unit and vice versa. However this affects the maintenance schedule immediately: maybe the first rolling stock unit was scheduled for maintenance in B immediately after its planned trip from A to B, but now another rolling stock unit arrives at B which may need no maintenance at all. Moreover, besides the immediate impact of the maintenance schedule, the effects may propagate for the next days as well, especially since there is not always a mechanism, nor an incentive, to bring back the rolling stock units in their original circulations.

It is unknown how this affects the results of the current research exactly, but the above demonstrates that, due to disruptions, the provided optimal maintenance schedule and optimal maintenance location choice is often not valid anymore during operations. This issue needs to be thoroughly addressed before implementing the results of the current research in practice.

**Reachability of maintenance locations**

In the current research it is assumed that if a rolling stock unit stands still at a certain location, it can also be immediately maintained at this maintenance location. For example, if a rolling stock unit stands still at a station for 30 minutes, then maintenance activities can be assigned to this maintenance opportunity as long as the total duration of the maintenance activties does not exceed 30 minutes.

However, in many cases, in order to perform maintenance on a rolling stock unit, shunting movements are required to transport a rolling stock unit from the station to a track where maintenance can be performed and to transport it back to the station from where it can continue its service. These shunting movements are not considered in the current research, but they do impact the feasibility of the provided solutions. In the first place, these shunting movements require time. In the second place, it is not always guaranteed that these shunting movements are possible from a logistic point of view: in busy railway networks, it may not always be possible to plan these shunting movements. Therefore, the provided solution by the **MSLCP** is only valid under the condition that the required shunting movements for planned maintenance activities can be performed.

This limitation can be partially overcome by adding slack time to the maintenance duration, i.e. requiring more time in order to do maintenance than is actually necessary. This increases the flexibility during operations and hence the chances that the provided solution by the **MSLCP** can be implemented in practice. This method is not expected to result in very different solutions, since the current work has shown that the solutions of the **MSLCP** are relatively insensitive to increasing maintenance duration. Another way to address this limitation is to adapt the rolling stock circulation to already include optional shunting movements to the yard, such that if a maintenance opportunity would be used, no additional shunting movements would be necessary.

#### Inaccurate information

The current research uses at least two sources of inaccurate information. In the first place, the definitions of maintenance types are not precise. Although the type A maintenance activity may correspond to a internal cleaning activity and the type B maintenance activity may correspond to a technical inspection, their durations are only indicative. Moreover, in the current research these are assumed to be equal for each rolling stock unit, whereas in reality these values often differ per rolling stock type. This impacts the extent to which the provided maintenance schedule can be implemented in reality. For commercial purposes, it is therefore relevant to verify these values thoroughly.

A second source of inaccuracy relates to the determination of costs. Since exact cost values could not be disclosed for reasons of confidentiality, estimates had to be made. Although these estimates are believed to have a realistic order of magnitude, they are synthetic. Moreover, to compute costs, a simplified cost structure has been assumed. In reality, more complex cost structure prevail. Hence, the results presented in the current research relating to costs must be considered suggestive and need to be interpreted with care.

## 10.3 Discussion of Viriato

The **MSLCP** is coupled to Viriato and can be invoked from it, i.e. a user can define a roster in Viriato, set parameters for the model in a GUI and start the algorithm from Viriato. This section aims to provide some insight in the factors that determine whether Viriato is a good tool to support the development of academic models intended for practical use. To this end, it is useful to distinguish between two target groups: on the one hand, the user of a model (e.g. a railway operator), and on the other hand, the developer of a model (e.g. a university). Below, the benefits and recommendations are given for both users and model developers.

#### Benefits

Firstly, a large benefit of Viriato is that the railway industry can directly benefit from academic research on railway timetabling. When a model has been connected to Viriato, it can be easily accessed and applied to the data bases that are often already set-up in Viriato. There is no need to use other software programmes for the application of academic research.

Secondly, the outcomes of railway optimization tools can be immediately visualized using the visualization opportunities Viriato offers. In the current case, for the **MSLCP**, the visualization of rolling stock circulations turned out to be particularly useful. This is in the first place a benefit to the user of the model. The user can immediately benefit from the planning assistance results in his or her daily planning business. In the second place, it is also a benefit for the developer of the model, since he or she is offered the opportunity to make quick visual checks whether the results of the model make sense, and also to report the model outcomes to others less involved in the research.

Thirdly, Viriato offers the opportunity to bring academic research and practice closer together. By working in the same environment, researchers are offered the opportunity to work with exactly the same data that railway operators use, and railway operators do not need any specific extra knowledge to be able to implement the model that is developed in academics.

**Recommendations**

To improve the usefulness of Viriato in supporting the co-operation between railway operators and academia, some recommendations can be made.

The NS rolling stock data format is not in a industry standard format, such as the format *RailML*, but a proprietary format. As such, the import of NS data into Viriato is not straightforward. Viriato in general does support the industry standard RailML for importing individual trains and train trips, but currently does not support the importing of rolling stock circulations. Nonetheless, rolling stock circulations can be imported using Viriato's rostering interface. However, in order to do so, one has to import train trips (i.e. activities with departure- and arrival time and node). Then, the departure and arrival nodes of these train trips need to be matched to the nodes that are in the Viriato infrastructure database. In a next step, one has to match the rolling stock units in the circulations from the NS data with the trains imported to Viriato. This can be done via the rostering interface or via some workaround in the Viriato database. In principle this method is viable, but it costs time and requires specific knowledge of RailML and Viriato.

Also, researchers may often want to test their model on many different scenarios. However, although Viriato is designed particularly to analyse specific scenarios in-depth, it is not designed to run several hundreds of scenarios to investigate the effect of various parameter settings or to test other model-specific characteristics or behavior. To test the effect of parameters that are not specified in Viriato, a workaround is possible, since the code can be changed in such a way that the problem is run for various of these parameters. For parameters, however, that need to be specified within Viriato (for the **MSLCP**, see Table 9.9, column 'Viriato param') there is a challenge. These parameter values need to be changed manually. For example, consider the most important input: the rolling stock circulation data. There seems to be no automated opportunity to run multiple scenarios with different rolling stock circulation data.

Moreover, the interaction between the GUI and the external optimizer is currently limited. First, for users of the model, the rostering interface does not support feedback during an optimization run, for instance to communicate the current status of the algorithm. Second, no extra model-specific parameters can be specified in Viriato. An important example for this in the current use case is the value of the maximum number of daytime maintenance locations. This is an important parameter of the **MSLCP**, but cannot be specified in Viriato and needs to be specified in the code instead, forming a potential barrier for users to utilize the model. However, the aforementioned points are currently under development in the more sophisticated product, called the *Algorithm Platform*. The Algorithm Platform will offer more ways of user interaction between Viriato and external optimizers.

# Chapter 11

# Conclusion

The current chapter concludes the present research. Section 11.1 returns to the research questions posed in Chapter 1. Section 11.2 outlines the most important directions for future research. Section 11.3 gives the principal recommendations for the practical implementation of the current research by railway operator NS and railway software developer and consultant SMA.

## 11.1 Answers to research questions

The current research has found its origin in the increasing use of railway networks and as a result the increasing complexity of rolling stock maintenance scheduling and maintenance location choice. Particular focus is on the case of the Dutch railway operator NS, that faces increasing pressure on the capacity of maintenance locations during nighttime and is investigating possibilities to perform more maintenance during daytime to mitigate this pressure. This strategy of daytime maintenance, however, comes with two complications. First, the making of an optimal maintenance schedule is a complex task. Second, it is not straightforward which locations that need to be opened for daytime maintenance, that is, at which locations personnel needs to be stationed so that maintenance can be performed. These two aspects are interrelated, as the optimal maintenance schedule depends on the optimal maintenance location choice and vice versa.

Chapter 1 indicated that the current research approaches this problem by addressing a main research question with four sub questions. The remainder of the current section answers first those four sub questions and then formulates an answer to the main research question.

#### Sub question 1: literature

The first sub question was formulated as follows: *What literature regarding rolling stock maintenance and rolling stock maintenance location choice is available?* Chapter 2 has given an overview of available research in rolling stock maintenance scheduling and rolling stock location choice. Due to the similarities in the characteristics of problems in the field of railway operations and aviation, also available literature in the field of aviation has been considered. Many works include maintenance scheduling, approaching this matter usually by finding a feasible allocation of moving units (for instance rolling stock units or aircraft) to a given set of trips that need to be fulfilled. The maintenance location choice has been addressed less frequently, though some authors focused on this matter. These works, however, do not consider explicitly the scheduling of maintenance activity.

It has been shown that the simultaneous optimization of maintenance scheduling and maintenance location choice did not seem present in the available literature, and as such the current work is a clear contribution. Moreover, the current work is a contribution as it distinguishes between daytime and nighttime maintenance. In addition, the location choice for daytime maintenance locations had been addressed before, but not yet from an optimization perspective, resulting in a third contribution of the current work.

**Sub question 2: model design**

The current work addresses the second sub question - *How can rolling stock maintenance scheduling and rolling stock maintenance location choice be efficiently modeled simultaneously?* - by providing three cooperating models in Chapters 3- 6. The **MSLCP** model is at the core of the research. It takes as its main input a rolling stock circulation and provides for this rolling stock circulation an optimal maintenance schedule and an optimal maintenance location choice that minimize the total number of maintenance activities during nighttime, thereby reducing the capacity pressure during the night. The **MSLCP** does not account for the (complex) issue of maintenance location capacity. To this end the **APP** model is designed. For a given **MSLCP** solution, it determines an optimal maintenance shift planning and it determines the minimum number of teams necessary for this planning. As such, it provides a method to measure the capacity required for any **MSLCP** solution. To use the information regarding capacity requirements that the **APP** obtains to obtain an **MSLCP** solution that satisfies pre-determined constraints regarding available capacity, the **CMSLCP** is designed. The **CMSLCP** uses a technique called *Logic-Based Benders' Decomposition*, iteratively computing a solution to the **MSLCP**, obtaining the required capacity using the **APP** and using this information to constrain the **MSLCP** further so that over multiple iterations the solution of the **MSLCP** converges to one that satisfies all capacity constraints.

**Sub question 3: results**

The third research question is formulated as follows: *What results can be obtained by applying the model to various rolling stock circulations?* Chapters 7 and 9 report the results that follow from the application of the **MSLCP**. For the NS case, in case all rolling stock units of type VIRM4 and VIRM6 are considered, up to 22.3% of work can be performed during daytime if five locations are opened, to even 42.0% if 20 locations are opened during daytime. The location choices is consistent for for different lengths of planning horizons, for different input data sets, and for different maintenance durations. The four locations with the highest assigned workload (in hours of activities) are Amr, Hdr, Mt and Hfdo. For the largest scenario (including all rolling stock units used for intercity services), a day share of at least 30.1 % can be achieved if 20 maintenance locations are opend. In this case, Gvc, Bkd, Bkh en Dv show to be the locations with the highest workloads. Moreover, the model is efficient for planning purposes: for the largest scenario run (with all rolling stock units used for intercity services) for a planning horizon of 42 days, the computation time was 3.2 hours. For operational purposes, however, this running time is not sufficiently efficient.

The **MSLCP** is also applied on a practical use case. This application demonstrates how the **MSLCP** can be used in actual situations and provides visualizations of the results, proving that the schedules determined by the **MSLCP** are according to expectation and valid in practice. Moreover, it provides rough cost estimates and shows for a situation with 5 maintenance locations opened that the day share increases (meaning the capacity problems during nighttime are mitigated) while the costs do not increase.

Results for the **APP** and **CMSLCP** are presented separately from the results for the **MSLCP**. The **APP** provides a solution quickly, i.e. within seconds for realistic problem sizes. This is beneficial since in the **CMSLCP** context it needs to be run for every iteration and therefore contributes to the efficiency of the **CMSLCP** model. For the **CMSLCP**, it has been demonstrated that using the **CMSLCP** model, an **MSLCP** solution is found that satisfies predetermined constraints regarding the capacity of maintenance locations. For the considered problem instance, in a set-up focusing only on one hard maintenance shift no optimal solution was found, but in a set-up considering multiple maintenance shifts, the number of maintenance shift for which the required capacity exceeded the available capacity could be reduced from 21 to 5 in less than 8 minutes. Various versions of the important sub process which generates the cuts have been proposed, showing that the design of the cut generation method is an important, distinguishing factor influencing the convergence of the algorithm, indicating that the generation of new, more efficient cut generation processes may significantly contribute to the further improvement of the **CMSLCP**. Moreover, it has been shown that the min-cut cut generation method is able to quickly decrease the number of maintenance shifts necessary to a reasonable amount, but that concerning solving a hard maintenance shift to optimality, the min-cut cut generation method is outperformed by the binary search heuristic cut generation method with 15 cuts.

**Sub question 4: Viriato**

The fourth sub question relates to the usability of the software programme Viriato in relation to the current research: *Is the Rostering Interface to the planning software Viriato an effective and efficient tool to model this kind of problems?* It has been addressed in Section 10.3. The Rostering Interface offers some clear benefits. First, it enables railway operators to easily use new, state-of-the-art models defined in academia. Second, it empowers researchers with an easy tool to immediately visualize solutions. Third, it contributes to bridging the gap between academics and practice. Also, some recommendations for its use in research have been made. First, although Viriato is able to handle the commercial input format RailML, in many cases the input is not available in this format, requiring the researcher to make additional conversion steps. Second, researchers may often want to test their models on many different scenarios, a functionality currently not supported by Viriato. Third, the interactivity between the GUI and the Rostering Interface is still limited, although this is currently under development and it is expected that more user interaction will be possible in future versions.

**Main research question**

Based on the answers of the four sub questions, an answer can be formulated to the main research question: *Given the rolling stock circulation, how to find efficiently an optimal rolling stock maintenance schedule and simultaneously optimize the choice regarding which locations are opened during daytime and during nighttime?* The current research proposes the **MSLCP** model which shows to be an efficient method to find an optimal maintenance schedule and an optimal location choice on the basis of a given rolling stock circulation. Based on the model it is concluded that, for the case with only rolling stock units of type VIRM4 and VIRM6, up to 42.0 % of all maintenance activities can be performed during daytime. For the large case with all rolling stock units used for intercity services, the running time is at most 3.2 hours, meaning that the model is efficient based on the defined criteria for efficiency. The **MSLCP** is a contribution to the literature since it simultaneously addresses maintenance scheduling and maintenance location choice. Moreover, its explicit distinction between daytime and nighttime maintenance is an addition to the literature.

To address also the capacity of maintenance locations, the **APP** model is proposed. This model not only computes the required capacity for any **MSLCP** solution, but has as an additional benefit that it provides an optimal maintenance shift planning. It is shown to solve within 5 seconds for problem instances of realistic sizes, meaning it is efficient even for operational use. As such, it is not only a valuable addition to the **MSLCP**, but it can also be useful in operational contexts where a shift planning is required.

The **CMSLCP** is designed to find a solution to the **MSLCP** that includes maintenance location capacity. It is relevant since it addresses the practical issue of including maintenance location capacity constraints in the maintenance scheduling problem, and proposes a method to solve this problem. Moreover, from a scientific point of view it is interesting as it proposes a framework to incorporate complex constraints in scheduling problems, although more research is required into this model to further improve the efficiency of the method as to make it more suited for commercial use.

## 11.2 Future research

Several directions for future research can be recommended.

First, the further development of the **CMSLCP** is considered to be an interesting research area. Although its practical use is evident, since it provides a solution to the **MSLCP** including capacity constraints, which are relevant in many contexts, especially its scientific relevance should be underlined. Its cut generation process offers many opportunities for improvement, and the lessons learned from its development can potentially be used in many other research areas related to scheduling of activities on locations and the capacities of these locations. Especially the generation of efficient cuts cuts for scheduling problems is relevant. Currently, the sub problem of the **CMSLCP**, which is the **APP**, provides information on specific sets of activities that cannot occur simultaneously. More efficient cuts can potentially be generated by using this information more efficiently, resulting in more general cuts that constrain a larger part of the search space of the **MSLCP**, hence leading to quicker convergence of the **CMSLCP** to a feasible solution. The current research proposes the min-cut cut generation method that intends to exploit the problem structure to generate efficient cuts, and this method has indeed proved useful in order to quickly reduce the number of maintenance shifts for which the required capacity exceeds the available capacity. However, in order to solve hard instances to optimality, it is still outperformed by the heuristic cut generation method. The suggested reason for this is that the efficiency of the latter benefits from the fact that many different cuts per iteration can be generated, a functionality that, if available, would also benefit the min-cut cut generation method. More research is necessary to gain more insights in the dynamics that determine the quality of each cut generation method. This knowledge offers valuable insights on the characteristics of well-performing cut generation methods and hence gives useful inspiration for the development of new cut generation methods, which in turn may add to the quality and usability of the **CMSLCP**.

Second, an interesting next research topic is how to improve the computational performance of the **MSLCP**. This improvement is especially relevant in the light of the **CMSLCP** model, since this model requires to run the **MSLCP** in each iteration again. There are several opportunities to improve the computational performance of the **MSLCP**. Using the structure of the problem, the problem may potentially be decoupled into multiple smaller subproblems that are much easier to solve in at least three ways. A first opportunity for decoupling may lie in the fact that currently a schedule for all rolling stock units is created simultaneously, while their interaction may be limited. A second opportunity can flow from the fact that the schedule for the several maintenance types is currently created simultaneously for all maintenance types, while the maintenance types possibly do not interact much. A third opportunity may be offered by considering a *rolling horizon* framework, thereby first optimizing a few days in ahead and iteratively adding more days to the optimization. This method would consider only a subset of the decision variables initially and gradually add more decision variables, hence in essence being a column generation approach. Moreover, in the context of the **CMSLCP**, defining a so-called *warm start* of the solving procedure of the **MSLCP** model based on its previous outcome may be beneficial.

Third, the current research can be extended by validating its solutions, which are based on planned data, against realised data. It is interesting to investigate how the solutions provided by the **MSLCP** perform in practice. This can potentially be done by computing an **MSLCP** solution based on planned data from a past period, and comparing it to realised data from the same period. This may, for example, be done by constructing a simulation of the historical situation in case a maintenance schedule provided by the **MSLCP** would have been adopted.

Fourth, the models from the current research can be applied to other contexts as well. It is interesting to extend the models proposed in the current research in order for them to become more versatile. Examples of such extensions are given in Chapter 10 and include the applicability of the research to serve other railway operators, the introduction of other objectives, the investigation of other cost structures and the inclusion of exchange opportunities of rolling stock units so that more rolling stock units can be maintained during daytime.

## 11.3 Practical recommendations

NS and SMA are two partners have been closely involved during the process of the current research. This section gives recommendations for both, having regard for their unique business objectives.

#### NS

The current research has been motivated by the increasing pressure on the capacity of maintenance locations during the night. The current research has investigated the potential benefits of daytime maintenance in order to solve this problem and it has demonstrated that, indeed, a considerable amount of activities can be moved to the day. As such, it is recommended that NS start the implementation of daytime maintenance. In doing so for intercity services only, up to 30.1 % of the associated hours of activities can be performed during daytime. This would require stationing personnel during daytime at 20 maintenance locations. If daytime maintenance is implemented, however, the scheduling of maintenance activities becomes complex. The **MSLCP** tool can assist in this planning process. Moreover, this potentially saves a considerable amount of resources since the scheduling of maintenance activities requires much planning capacity.

The optimal locations at which personnel needs to be stationed to perform maintenance during daytime flows immediately from the **MSLCP**. It must be taken into account that the **MSLCP** provides a location choice based on the rolling stock circulation, which is usually valid for only eight weeks. Therefore, if NS accepts the fact that the location choice can change for a next version of the rolling stock circulation, then the solution of the **MSLCP** can just be used for the maintenance location choice. On the other hand, if NS prefers a location choice which is valid for a longer period of time, the **MSLCP** needs to be run for multiple BDu and the location choices need to be determined. Only locations that robustly show up over all (or most) BDus should then be opened.

Moreover, the **APP** model proposed in the current research is also expected to be of added value for NS and its use is recommended in operations. It runs quickly (within a few seconds of computation times) and provides an optimal minute-to-minute assignment of maintenance staff teams to maintenance jobs. Therefore, rolling stock dispatchers can use it to determine an optimal maintenance shift planning given the latest information available on the arrival and departure time of rolling stock and the activities that need to be performed to it.

**SMA**

In its consulting activities, SMA often needs to determine (optimal) rolling stock circulations for railway operators. The **MSLCP** is able to assign maintenance activities to any rolling stock circulation. In that capacity, it can enrich the rolling stock circulations that SMA creates for its clients by adding maintenance activities, making the rolling stock circulations more realistic and hence producing more reliable results. The use of **MSLCP** is therefore recommended in the creation of rolling stock circulations. In addition, the **MSLCP** can also be provided as a tool in Viriato, so that users of this software can add maintenance activities to their own rolling stock circulations themselves. Especially its scheduling capability is suitable for this - the location choice, that the **MSLCP** also provides, needs to be considered with more care and is therefore less suited as part of Viriato.

Similarly, the **APP** is relevant in the consulting activities as well as being a part of the software since it can be used as a stand-alone application (outside of the context of the **MSLCP**). In consulting, it is able to demonstrate clearly to clients that a feasible maintenance shift planning can be made for a given rolling stock circulation. Also, it can be used to determine a track occupation plan: often, a limited number of tracks is available and rolling stock needs to be assigned to these tracks to receive maintenance. Furthermore, it is recommended that the **APP** is implemented in Viriato, enabling users to immediately investigate the effects of any maintenance schedule on the planning at any maintenance location.

# Appendix A

# Interview reports

This appendix presents reports of three interviews held with representatives from railway operators outside The Netherlands: from NMBS (Belgium) in Appendix A.1, from DB Regio Bavaria (Germany) in Appendix A.2, and for SNCF Voyages (France) in Appendix A.3.

## A.1 NMBS (Belgium)

This interview was held with Hendrik Bonne on January 31, 2020. Bonne is head of the Continuous Improvement department at Belgium's main railway operator NMBS. The Continuous Improvement department is responsible for a variety of large-scale projects to improve the efficiency of all kinds of processes, among which those processes related to rolling stock maintenance. The department of Bonne operates in close cooperation with the department for short-term maintenance (also called low-level maintenance) and long-term maintenance (als called high-level maintenance); for SAP ERP systems (related to administration of maintenance plans); and for Industrial Engineering (involved in the development of for example maintenance instructions). Moreover, spread over the organization, some projects are running that involve the minimization of rolling stock out-of-service times.

**Maintenance activities**
According to Bonne, NMBS distinguishes various maintenance activities, varying in periodicity from 6 hours to multiple months.

The following technical checks are distinguished.

- Daily investigation (Dagelijks Onderzoek, DO). This is a very small technical inspection of approximately 10 minutes, that can be done by driver or by shunting personnel and that includes, for example, a breaking test. It can take place at any location, such as stations.
- Limited investigation (Beperkt Onderzoek, BO). This is a somewhat longer technical inspection of approximately 1 hour that takes place approximately every 7 days. Like DO, it can take place at any location, such as stations.
- Thorough Check (Grondige Schouwing, GS). This is a technical inspection of approximately 8 hours with a periodicity of 1 month. It should be executed at a maintenance location that is equipped for this type of maintenance (TWs).

For cleaning, there are inspections on many different levels. On the lowest level, there is a check that is executed every 6 hours and takes approximately 30 minutes. It includes going through the rolling stock unit to remove the most important garbage. Other checks need to be performed every month or every few months, and include more specific cleaning activities such as the cleaning of the ceiling.

**Maintenance locations**
Bonne indicates that, at NMBS, maintenance locations at three levels are distinguished: there are 3 Centrale Werkplaatsen (CWs), 9 Tractiewerkplaatsen (TWs) and 19 Onderhoudsposten (OPs).

- CWs are intended for heavy maintenance work. Rolling stock units visit these locations infrequently. However, if a rolling stock unit visits one of these locations, the visits take up to several days.
- TWs are responsible for many different kinds of small-scale technical maintenance activities (including repairs). The Thorough Check can be performed at this location.
- OPs are intended for light maintenance such as cleaning activities and small technical inspections. OPs are also the preferred location to park rolling stock units. DO and BO can be performed at this location, as well as cleaning activities.

Locations can be close to the station, but are sometimes also farther away (up to 15 minutes to drive). For new locations to be built, it is desired that these locations are close to large stations as a result of a new maintenance planning philosophy in which rolling stock units visit maintenance locations more often.

**Maintenance planning** Bonne explains that, at NMBS, there is no fixed rolling stock circulation that determines for each rolling stock unit which trip it will cover. The assignment of rolling stock to trips is arranged at an operational level: there is a number of rolling stock units at a location, and any of these rolling stock units may be picked to cover a given trip. After a trip, a maintenance activity may be scheduled in the timetable. Then, the rolling stock unit that covered the corresponding trip will be routed towards a maintenance location.

Currently, at NMBS, a new maintenance scheduling principle is being adopted for small-scale maintenance. Therefore, when considering maintenance planning, Bonne distinguishes the 'old' system and the 'new' system.

In the old system, rolling stock maintenance activities are scheduled after specific train trips. The scheduled time windows that can be used for maintenance are sufficient to perform all fragments that are contained in a maintenance activity of a specific type. It is, however, not always known beforehand which rolling stock unit will fulfil which train trip. Therefore, the required periodicities are "approximately" attained: sometimes the actual periodicity may be somewhat shorter than required and sometimes somewhat longer.

The new system, referred to as Timetable Integrated Maintenance (TIM), takes the timetable as a basis and sends rolling stock units to maintenance locations for short pit stops. These pit stops are not long enough to do all fragments of the maintenance activity of a specific type. Therefore, when a rolling stock unit comes in for a pit stop, the operators check which fragments of the maintenance activity need to be performed most urgently, and these are then carried out. In TIM, the maintenance time windows are much shorter compared to the old situation, but on the other hand rolling stock units visit maintenance locations much more often.

Light maintenance activities like DO, BO and interior cleaning are not planned but executed only when a rolling stock unit is at an OP. At such a moment it is checked whether such a check is necessary. Large-scale maintenance, as opposed to small-scale maintenance, is planned multiple months in advance. The daily operators make sure the rolling stock unit arrives in time, i.e. that it is in time for its appointment at a CW. This is not incorporated in the regular rolling stock schedule.

**Capacity** Traditionally, much maintenance was performed during the day. This was possible since there was a high peak load, meaning that there are a lot less rolling stock units during the day (giving the opportunity for maintenance). In the new system (TTIM), rolling stock units are inspected much more often. For this also the night is used. This means that in the new situation, an approximated 40% of all maintenance activities will be carried out during the night. According to Bonne, the current capacity is still sufficient, but there is not a large margin.

## A.2 DB Regio Bavaria (Germany)

This interview was held with Hendrik Kuhlmann on February 18, 2020. Kuhlmann is head of the department of Operational Planning (OP) at DB Regio Bavaria. His department is responsible for the planning of rolling stock circulations (i.e. connecting passenger trips to one another producing a sequence of trips for one rolling stock unit) and the shift planning for train drivers. The department focuses on the region of Bavaria, although there is sometimes some overlap with neighbouring regions.

**Maintenance activities** At the OP department, maintenance activities are scheduled for rolling stock. This concerns regular maintenance activities. Heavy maintenance activities, such as the large check that has to be performed every 4 to 6 years, are excluded from the responsibilities of the OP department.

According to Kuhlman, two types of technical maintenance activities are distinguished: small inspections and larger checks. In the first place, the small inspection has to be performed every 24 hours. This inspection is usually performed by train drivers after the last trip of the day and takes approximately 10 minutes. In addition, there is a larger check. The moment when this larger check has to be performed is officially determined by a rolling stock type-specific mileage criterion, but is in practice planned in such a way that there is an opportunity for the larger checks every 10 days (approximately). This technical check takes multiple hours; approximately 8 to 12 hours. For this larger check the rolling stock unit needs to be routed to a maintenance location.

Usually the entire rolling stock unit is checked at the same time and at the same location. However, in rolling stock unit combinations with a locomotive, the check of the locomotive is sometimes separated from the check of the rest of the train. This is undesirable, hence for newer rolling stock unit types, this separation is avoided.

In addition to technical checks, there are cleaning checks, which include emptying the toilets and mopping floors. Different rules apply to the intervals between cleaning checks. For example, if the rolling stock unit drives less than 500 km per day, a cleaning check may be scheduled every 3 days, whereas otherwise an interval of 2 days is applied, and also for local trains cleaning activities may be scheduled more often. On average, each rolling stock unit is cleaned with an interval of approximately two days. Cleaning activities have a duration of approximately 1-2 hours. Cleaning activities are usually performed at yards; especially at those yards where they ended their last trip.

**Maintenance locations** In the region of Bavaria, 6-7 maintenance locations for larger maintenance checks exist. Moreover, there are approximately 30-40 yards where cleaning activities and small technical inspections can be performed. These are usually close to train stations.

**Maintenance scheduling** Inspections are performed at the location where a rolling stock unit ends its last trips, but not explicitly planned. The larger technical checks, however, are planned. The Operational Planning department takes into account time windows for larger checks and for cleaning activities in the rolling stock circulation. When disruptions occur, a dispatcher is free to override a planned time horizon for a technical check or cleaning activity.

The rolling stock circulation is created on a yearly basis, but throughout the year many adaptations are made due to circumstances such as infrastructure construction work. These adaptations are usually made approximately 3 months in advance.

**Capacity** At DB Regio Bavaria, it depends on the specific maintenance location whether maintenance is carried out during the day and during the night. This may have several causes. Firstly, some lines have peak directions; that is, some cities, such as Munich, experience many incoming rolling stock units at the start of the day and many outgoing rolling stock units at the end of the day due to commuting traffic. As a result, during the day many rolling stock units are standing still. Therefore, at the maintenance locations in Munich, more daytime maintenance can be performed. Secondly, some contracts for railway lines include the obligation for rolling stock reserves. These reserves are often not used and can be maintained during the day when they are standing still.

Kuhlmann states that the capacity of the maintenance locations of DB Regio Bavaria is under pressure. Two causes for this can be identified. Firstly, a lot of new rolling stock is arriving as a result of increasing passenger demand. Secondly, much rolling stock (especially the new rolling stock) encounters problems and needs to visit maintenance locations more often. As a result the long term rolling stock plan cannot be used and replacement concepts need to be designed. According to Kuhlmann, this increases the demand for maintenance and leads to capacity issues at maintenance locations.

## A.3 SNCF Voyages (France)

This interview was held with Philippe Blanc on February 18, 2020. Blanc is head of the Planning department at SNCF Voyages, Axe Atlantique. SNCF Voyages is the provider of long-distance trains in France. Its Axe Atlantique takes care of all trains in the western and south-western direction. The planning department is responsible for railway planning studies, railway planning design and adaptations to the railway planning.

The long-distance railway network in France is star-shaped with four stations in Paris at the center. Most rolling stock units commute between Paris and cities in the country. There is limited interaction between rolling stock units between one line and the other: rolling stock units usually 'stay on their line'. An exception is the traffic called intersecteur, which travels around Paris (e.g. from west to east), but this comprises only a small share of the long-distance trips. SNCF Voyages' Axe Atlantique serves the lines starting or ending at station Paris-Montparnasse.

**Maintenance activities** Blanc indicates that Maintenance activities are categorized into three categories.

- Level 1-maintenance concerns the regular technical inspections. The duration of these inspections is approximately 2-3 hours. The interval of these inspections is based on mileage and takes place approximately every 3-4 days (in some cases, the interval may measure up to a week).
- Level 2-maintenance is a thorough maintenance check. To receive this type of maintenance, a rolling stock unit is out of service for an entire day. The duration of this type of maintenance is approximately 12-24 hours. Level 2-maintenance takes place with an interval of approximately 1 month.
- Level 3-maintenance is heavy maintenance. This type of maintenance is performed with an interval between 6 to 24 months and takes multiple days.

Toilet cleaning is taken into account separately, although it is often combined with Level 1-maintenance. The duration of this activity is approximately 1 hour. It takes place approximately every 2-3 days.

Interior cleaning is not considered part of the maintenance process. Interior cleaning takes place after each long-distance trip. Its duration is flexible (varies between 5 to 30 minutes) and depends on the available time: if little time is available, the quality of the cleaning inspection is scaled down.

In addition to these activities, there are also some minor inspections that the train drivers have to perform before each trip, but these inspections do not have a periodic nature.

**Maintenance locations** Blanc explains that there are seven maintenance locations, of which four are in Paris (near each large station in Paris). At the maintenance locations in Paris, all levels of maintenance can be performed. The other three maintenance locations are smaller and capable of maintenance activities of Level 1 (and sometimes parts of Level 2), but never of Level 3. Cleaning activities can be performed at all maintenance locations.

**Maintenance scheduling** According to Blanc, maintenance activities of Level 1 and Level 2 are planned in the rolling stock circulation, including toilet cleaning. Level 3-maintenance and cleaning activities are not taken into account in the roster but handled separately. The first steps in the creation of this roster are two years before the moment of execution, but it is continually updated. Four months before execution the ticket sale starts and the roster should then be final (although also after this date many adjustments are made).

**Capacity** At SNCF Voyages, peak hour demand around Paris is observed. This means that many trains enter Paris in the morning and many trains leave Paris in the evening. As a result, many rolling stock units can be maintained (especially Level 1-maintenance) during the day in Paris. Also, part of the rolling stock units is out-of-service to receive Level 2- or Level 3-maintenance. As a result, maintenance locations are in operation during the day (with emphasis on Level 2- or 3-maintenance) as well as during the night (with emphasis on Level 1-maintenance).

According to Blanc, capacity problems are being encountered currently at maintenance locations, both because not enough track is available and because maintenance staff is scarce.

# Appendix B

# MSLCP results

The current appendix presents additional **MSLCP** results. Appendix B.1 describes how a rolling stock circulation is obtained from NS input data. Appendix B.2 presents unaggregated results from all scenario batches from Chapter 7, for reference. Appendix B.3 defines additional **MSLCP** experiments and Appendix B.4 presents the corresponding results.

## B.1 Rolling stock circulations

The current section outlines how, from the BDu data that was provided as input by NS, the rolling stock circulation is obtained, which is used throughout the current research.

A BDu contains all planned rolling stock unit movements to be operated by NS for the period concerned. For each rolling stock unit movement, the BDu contains the time and location of departure as well as the time and location of arrival. In the BDu, various subsequent rolling stock unit movements on one specific day are linked to each other, defining a planned 'day path' for a rolling stock unit on a specific day. Each day path is linked to a day path on the following day. This link is called a *night transition*. Any night transition in the BDu satisfies the criterion that the end location of a day path of one day is equal to the start location of the day path for the next day to which the first day path is linked.

Day paths are given for a standard week. This means that, for instance, the set of day paths on a Monday of some particular week is identical to the set of day paths on Monday of any other week. Note that this does not necessarily imply that the rolling stock circulation for any specific rolling stock unit is identical every week. In general, the transitions cause the rolling stock units to have different day paths each week.

The day paths and corresponding transitions define planned paths for all rolling stock units. By connecting day paths for multiple days using the given transitions, a rolling stock circulation can be obtained.

To ease the interpretation of the results and to match the mathematical model formulation, the planning horizon starts at midnight of the first day. In the input data, however, a new day starts at approximately 4am in the morning (although this moment in time is not guaranteed). This implies that, for the first day, the activities between midnight and (approximately) 4am are not considered as they are still part of a previous day which is not in the data set. To prevent this problem, the first day has been cut off, resulting in the time horizon starting at midnight the following day. For any activities that start before midnight of the first day and end after midnight of the first day, the start time has been set to midnight.

A similar problem exists at the end of the planning horizon: it is desired that the planning horizon ends at midnight, but in the input data set there are still activities between midnight and (approximately) 4am which are falsely included in the planning horizon. To prevent this problem, the activities after the end of the planning horizon have been cut off. For any activities that start before midnight of the last day and end after midnight of the last day, the end time has been set to midnight.

Another issue is that, in most BDus there are a few occurrences of non-consecutive trips, meaning that the end location of the one trip is not equal to the departure location of the next location. In most cases, the two (distinct) locations are nonetheless very close to each other. For example, it may occur that a rolling stock unit arrives at the northern part of some location and departs at the southern part of this same location, that in some cases have a distinct location name. This issue is not problematic in practice, but for the implementation it is: for the intermediate MO it is not straightforward at which location it takes place. Therefore, in the current research, for such cases it is assumed that these MOs take place at the location where the second trip departs. This choice is arbitrary, but it is expected to have negligible influence on the results.

For instance, in period 10 (9-6-2019 until and including 1-9-2019), there are 606,662 records (rolling stock movements). There are 1,012 occurrences of unequal start- and end locations. There are 4 different combinations of locations for which this happens. This works in both directions. Table B.1 indicates for which combinations of locations this occurred.

| location combination | | occurrences |
|---|---|---|
| Amsterdam work shop north (Aswpln) | Amsterdam work shop south (Aswplz) | 324 |
| Amsterdam Dijksgracht east (Dgro) | Amsterdam Dijksgracht west (Dgrw) | 164 |
| Deventer (Dv) | Deventer goods yard (Dvge) | 12 |
| Zutphen (Zp) | Zutphen goods yard (Zpge) | 512 |
| | | 1012 |

*Table B.1: The input data for BDu id 10 did not always match the criterion that the first activity ends at the same location where the second activity starts. This occurred for the combinations of locations indicated in this table. It is also indicated how often it occurred.*

## B.2 Result tables

This section presents tables with KPIs for scenario batch 1, 2 and 3 from Chapter 7. Many of the results in Chapter 7 can be reproduced using these tables. Also, they serve for the reader to understand the contents of the scenario batches introduced in Section 7.1.2.

Two KPIs are presented: the average number of hours of activities per day (both during daytime and during nighttime) and the *day share*, i.e. the percentage of the number of hours of activity that is performed during daytime. Most results presented in the sequel of this section are based on these figures.

Table B.2 gives the mean hours of activity per day and Table B.3 gives the day shares (i.e. hours of activity performed during daytime as a fraction of the total hours of activity) for scenario batch 1. Table B.4 gives the mean hours of activity per day and the day shares for scenario batch 2. Note that some infeasible solutions have been found here, for which no particular reason has yet been found. Table B.5 gives the mean hours of activity per day and the day shares for scenario batch 3.

*(Text continues after tables.)*

| $\delta^D$ | 7 | | | | | | | | | | | | 10 | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\nu$ | 7 | | | | 21 | | | | 42 | | | | 7 | | | | 21 | | | | 42 | | | |
| $L^D_{max}$ $\tau$ | 10 | 20 | 92 | 160 | 10 | 20 | 92 | 160 | 10 | 20 | 92 | 160 | 10 | 20 | 92 | 160 | 10 | 20 | 92 | 160 | 10 | 20 | 92 | 160 |
| 0 | 8.2 | 16.3 | 76.6 | 132.0 | 9.0 | 17.8 | 81.4 | 139.9 | 9.3 | 18.2 | 82.8 | 142.2 | 8.2 | 16.3 | 76.6 | 132.0 | 9.0 | 17.8 | 81.4 | 139.9 | 9.3 | 18.2 | 82.8 | 142.2 |
| 1 | 8.3 | 16.6 | 77.3 | 133.1 | 9.1 | 17.9 | 82.1 | 141.2 | 9.3 | 18.3 | 83.6 | 143.5 | 8.2 | 16.5 | 77.1 | 132.8 | 9.1 | 18.0 | 82.1 | 140.9 | 9.3 | 18.3 | 83.6 | 143.2 |
| 2 | 8.4 | 16.6 | 78.2 | 134.6 | 9.2 | 18.4 | 83.4 | 143.0 | 9.5 | 18.7 | 84.9 | 145.3 | 8.2 | 16.7 | 77.9 | 134.1 | 9.2 | 18.2 | 83.1 | 142.3 | 9.5 | 18.6 | 84.7 | 144.6 |
| 3 | 8.4 | 17.0 | 79.4 | 136.4 | 9.4 | 18.6 | 84.8 | 145.0 | 9.7 | 19.0 | 86.4 | 147.4 | 8.4 | 17.0 | 78.7 | 135.3 | 9.3 | 18.5 | 84.1 | 143.8 | 9.6 | 18.8 | 85.7 | 146.2 |
| 5 | 8.7 | 17.6 | 80.6 | 137.9 | 9.6 | 19.0 | 86.1 | 147.2 | 9.8 | 19.3 | 87.7 | 149.8 | 8.4 | 17.1 | 79.6 | 136.4 | 9.5 | 18.9 | 85.3 | 145.2 | 9.8 | 19.3 | 86.9 | 147.6 |
| 20 | 8.9 | 18.3 | 85.8 | 146.0 | 10.1 | 20.0 | 91.9 | 155.9 | 10.4 | 20.2 | 93.4 | 158.5 | 8.4 | 17.4 | 81.7 | 140.0 | 9.7 | 19.3 | 88.1 | 149.5 | 10.0 | 19.7 | 89.8 | 152.0 |

*Table B.2: Mean hours of activity per day (of 24 hours), calculated as the total number of hours of activity divided by the number of days in the planning horizon, for all scenarios in scenario batch 1, for different values of the start of the daytime hour $\delta^D$, the number of days in the planning horizon $\nu$, the number of rolling stock units $\tau$ and the maximum number of locations opened during daytime $L^D_{max}$.*

| $\delta^D$ | 7 | | | | | | | | | | | | 10 | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\nu$ | 7 | | | | 21 | | | | 42 | | | | 7 | | | | 21 | | | | 42 | | | |
| $L^D_{max}$ $\tau$ | 10 | 20 | 92 | 160 | 10 | 20 | 92 | 160 | 10 | 20 | 92 | 160 | 10 | 20 | 92 | 160 | 10 | 20 | 92 | 160 | 10 | 20 | 92 | 160 |
| 0 | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% |
| 1 | 8% | 5% | 5% | 3% | 4% | 5% | 3% | 3% | 4% | 4% | 3% | 3% | 3% | 4% | 3% | 2% | 2% | 3% | 3% | 2% | 3% | 2% | 3% | 2% |
| 2 | 13% | 12% | 8% | 7% | 8% | 11% | 8% | 7% | 8% | 9% | 8% | 7% | 4% | 8% | 7% | 5% | 8% | 8% | 7% | 5% | 7% | 7% | 7% | 5% |
| 3 | 16% | 16% | 13% | 11% | 14% | 15% | 13% | 11% | 14% | 14% | 13% | 11% | 9% | 12% | 11% | 9% | 11% | 12% | 10% | 9% | 11% | 10% | 10% | 9% |
| 5 | 28% | 29% | 21% | 18% | 23% | 26% | 21% | 18% | 23% | 22% | 21% | 18% | 11% | 19% | 17% | 13% | 19% | 21% | 16% | 13% | 20% | 18% | 16% | 13% |
| 20 | 35% | 45% | 45% | 39% | 46% | 45% | 44% | 39% | 43% | 41% | 44% | 38% | 14% | 27% | 28% | 24% | 28% | 31% | 28% | 23% | 29% | 28% | 28% | 23% |

*Table B.3: Share of the hours of activity performed during daytime (of 24 hours), calculated as the total number of hours of activity during daytime divided by the total number of hours of activity during both daytime and nighttime, for all scenarios in scenario batch 1, for different values of the start of the daytime hour $\delta^D$, the number of days in the planning horizon $\nu$, the number of rolling stock units $\tau$ and the maximum number of locations opened during daytime $L^D_{max}$.*

| $L^D_{max}$ | Standard duration: hrs. of activity | Standard duration: day share | Increased duration: hrs. of activity | Increased duration: day share |
|---|---|---|---|---|
| 1 | 151.2 | 21.9% | n/a | n/a |
| 2 | n/a | n/a | n/a | n/a |
| 3 | n/a | n/a | n/a | n/a |
| 4 | n/a | n/a | n/a | n/a |
| 5 | n/a | n/a | n/a | n/a |
| 6 | n/a | n/a | n/a | n/a |
| 7 | 153.7 | 23.8% | 223.1 | 13.0% |
| 8 | 153.3 | 24.1% | 223.1 | 13.7% |
| 9 | 153.3 | 24.1% | 223.3 | 13.8% |
| 10 | 152 | 22.7% | 221.7 | 12.7% |
| mean | 152.7 | 23.3% | 222.8 | 13.3% |

*Table B.4: Hours of activity and day shares for scenarios in scenario batch 2, for different values of the maximum number of daytime maintenance locations $L^D_{max}$ and for the two different maintenance definitions included in this scenario batch. For some scenarios, no solutions could be obtained since the model turned out to be infeasible.*

| $\nu$ / $L^D_{max}$ | 7: hrs. of activity | 7: day share | 42: hrs. of activity | 42: day share |
|---|---|---|---|---|
| 10 | 305.9 | 22.6% | 328.8 | 22.2% |
| 20 | 310.6 | 30.9% | 334.9 | 30.1% |

*Table B.5: Hours of activity and day shares for scenarios in scenario batch 3, for different values of the maximum number of daytime maintenance locations $L^D_{max}$ and the planning horizon $\nu$ in days.*

## B.3 Definition of additional MSLCP experiments

The following experiments are run in addition to those mentioned in the main text. Their results are presented in Appendix B.4.

**Experiment 5: Interval distribution** One of the constraint sets of the **MSLCP** governs that the intervals between subsequent maintenance activities do not exceed certain bounds. The way intervals should be measured, however, is ambiguous. Therefore various ways to calculate the interval are computed an presented to provide more insight in the model workings. For this experiment, scenario batch 1 is used.

**Experiment 6: Feasibility determinants** The definition of when an MO is classified as daytime is not straightforward. Another method to define this has been proposed as well. This experiment investigates the effect of that definition on the number of feasible solutions and on the quality of the feasible solutions. For this experiment, scenario batch 1 is used, and in addition some information from another batch run with another daytime definition (not reported under the scenario batches in Section 7.1 due to its minor role in the Results section).

**Experiment 7: Maintenance type sensitivity** The maintenance duration is an input to the **MSLCP**. This maintenance duration is an estimate. For a specific railway operator, it may vary from day to day. Similarly, the requirements for the maintenance duration may vary from railway operator to railway operator. As such, it is of interest to investigate the sensitivity of the model to variations in the duration of maintenance, which is the focus of this experiment. For this experiment, scenario batch 2 is used.

**Experiment 8: Running times** For a model which can be used in practice, it is important that the running times of the model are acceptable. To this end, for various scenarios the computation times are presented and an insight is provided into the main determinants of this running time.

## B.4 Results for additional MSLCP experiments

### B.4.1 Experiment 5: Interval distribution

It is also relevant to investigate whether the intervals between maintenance activities are as expected. In the model formulation, the intervals are measured from end of the first maintenance opportunity to the start of the second maintenance opportunity. However, the maintenance activities take place, in reality, before the end of the first maintenance opportunity and after the start of the second maintenance opportunity. This means that the actual time between two consecutive maintenance activities of the same type may in reality be longer than the maximum interval. In other words, the way the intervals are measured in the model may be too optimistic.

To this end, Table B.6 gives statistics on the actually attained intervals. It distinguishes between three methods to measure the interval between maintenance activities. The e-s method measures the interval from the end of the first MO until the start of the second MO. This measure is also used in the **MSLCP** to determine the time between maintenance activities. However, two other methods may also be advocated. The e-e method measures the interval between maintenance activities as the time from the end of the first MO until the end of the second MO. The s-s method measures the interval between maintenance activities as the time from the start of the first MO until the start of the second MO.

| | type A | | | type B | | |
|---|---|---|---|---|---|---|
| $L^D_{max}$ | e-s | s-s | e-e | e-s | s-s | e-e |
| 0 | 16.1 | 26.5 | 26.9 | 37.3 | 50.6 | 51.2 |
| 1 | 16.0 | 26.0 | 26.4 | 37.2 | 50.5 | 51.1 |
| 2 | 15.9 | 25.6 | 25.9 | 37.1 | 50.1 | 50.7 |
| 3 | 15.7 | 24.9 | 25.3 | 37.1 | 50.1 | 50.6 |
| 5 | 15.8 | 24.6 | 25.0 | 36.2 | 48.9 | 49.5 |
| 20 | 15.4 | 23.5 | 23.8 | 34.5 | 46.0 | 46.6 |

*Table B.6: The maintenance intervals measured from end to start (e-s), start to start (s-s) or end to end (e-e) of the MO in which the first maintenance activity takes place and the MO in which the second, consecutive maintenance activity takes place. Figures given for various $L^D_{max}$. Further, $\delta^D = 10$, $\nu = 42$ and the analysis contains all VIRM4 and all VIRM6.*

It appears that the interval from end to start is always below the requirement (for type A 24 hours, for type B 48 hours). This is expected since the interval requirement is included as a constraint in the **MSLCP** model. The other intervals (measured from the start of the first MO to the start of the second MO, or measured from the end of the first MO to the end of the second MO, respectively, are around or a little above the requirement. This is presumably not problematic since, contrary to the e-s measure, these measures are mostly too conservative in practice. The e-e and s-s measures implicitly incorporate rolling stock degradation during either the first or the last MO. For example, in the e-e measure, it is assumed the rolling stock unit's degradation starts at the end of the first MO (i.e. when it starts driving again) - this is considered reasonable. However, it assumes that the degradation ends a the end of the last MO - this is not reasonable, since at that moment, the rolling stock unit has already been maintained. A similar (symmetric) argument applies to the s-s measure.

### B.4.2 Experiment 6: Feasibility determinants

In the current situation, an MO is classified during daytime if and only if its start time is after $\delta^D$ and its end time is after $\delta^N$, and in addition, the start and end times are on the same date. This is in some cases too conservative, since MOs that are partially during the night and partially during the day are classified as nighttime MOs, whereas in reality maintenance could have been performed during the day.

Therefore, another maintenance definition is proposed. This maintenance definition classifies an MO to be during the day if and only if its *end time is during the day*. Table B.7 gives for each value of $L^D_{max}$ how many feasible solutions were obtained.

| | | standard definition | | alternative definition | |
|---|---|---|---|---|---|
| | # scenarios | 07.00 | 10.00 | 07.00 | 10.00 |
| 0 | 24 | 12 | 12 | 0 | 0 |
| 1 | 24 | 12 | 12 | 0 | 0 |
| 2 | 24 | 12 | 12 | 0 | 0 |
| 3 | 24 | 12 | 12 | 0 | 1 |
| 5 | 24 | 12 | 12 | 0 | 3 |
| 20 | 24 | 12 | 12 | 9 | 12 |

*Table B.7: Number of feasible solutions in each group for a specific $L^D_{max}$. Each group contains 24 scenarios, of which 12 for $\delta^D = 7$ and 12 for $\delta^D = 10$. The number of feasible solutions for both definition is reported.*

In the standard definition, a feasible solution is obtained for both $\delta^D = 7$ and $\delta^D = 10$. However, the alternative definition appears to yield many infeasible solutions. In the alternative definition with $\delta^D = 7$ only 9 feasible solutions were found (out of 24 scenarios) and in the alternative definition with $\delta^D = 10$ only 16 feasible solutions were found (out of 24 scenarios).

The reason that the alternative definition performs so badly is because it classifies all maintenance activities with an arrival hour between 07.00 and 19.00 (or 10.00 and 19.00, depending on the choice of $\delta^D$) as a daytime activity. What often occurs is that a nighttime maintenance opportunity ends just after 07.00 (or 10.00). These maintenance opportunities, however, are often critical for the creation of a feasible schedule. Hence, such an MO needs to be used, otherwise no feasible maintenance schedule satisfying the interval constraints can be generated. However, since in the alternative definition, such an MO is classified to be during daytime, the MO can only be used if the associated location is opened during daytime. Hence, as a result, many locations need to be opened for daytime maintenance to be able to create a feasible schedule. Many of these maintenance locations are not very attractive locations to open for daytime maintenance, since they may only be used for this single MO (to create a feasible schedule). In other words, the opening of these locations is necessary because otherwise no feasible solution is obtained, as opposed to because it would reduce the capacity issues during nighttime.

The above explains why the choice of $\delta^D = 7$ yields even more infeasible solutions as the choice for $\delta^D = 10$: if $\delta^D = 7$, there is a higher chance that a critical 'nighttime' MO ends just after the start of the daytime time window and is hence classified as a daytime maintenance. Moreover provides an explanation for why the number of infeasible solutions decreases if more maintenance locations can be opened for daytime maintenance: if more maintenance locations are available for daytime maintenance, it is less problematic that a critical MO requires to open a location during daytime, since an abundant number of locations may be opened during daytime. If, however, the number of maximum daytime locations is lower, a lower number of critical MOs may lead to infeasibility already.

### B.4.3 Experiment 7: Maintenance type sensitivity

The **MSLCP** assumes that the durations of maintenance types are known. However, these durations are estimates and may contain uncertainty. This experiment addresses the sensitivity of the model to a higher maintenance duration.

To this end, the scenarios from batch 2 have been taken as a starting point. The 5 feasible solutions for the default maintenance types are compared to the 4 feasible solutions for the maintenance type with extended durations.

First, the consistency of the location choice for two different maintenance duration choices is relevant. Since the estimate for the maintenance duration is uncertain, the actual maintenance durations in practice may deviate from the value used to find an optimal maintenance schedule and an optimal location choice. Therefore, it is desirable that the maintenance location choice is consistent for varying maintenance durations, since this would imply that the provided solutions are robust under various maintenance durations.

Figure B.1 displays the location consistency for both variants of the maintenance durations. To describe the consistency of any location, the percentage of scenarios in which it was opened is reported, relative to the total number of scenarios in which a feasible solution could be obtained. As an example, consider the location Bkd. For the standard duration, it is opened in 40% of the scenarios (i.e. in 2 of the 5 scenarios that resulted in a feasible solution for this maintenance type duration), whereas for the increased maintenance durations, it is opened in 50% of the scenarios (i.e. in 2 of the 4 scenarios that resulted in a feasible solution for this maintenance type duration).

As a result, it can be inferred that the maintenance location choice is consistent over varying maintenance durations for most locations: most locations chosen in all of the scenarios for the standard durations are also chosen in all of the scenarios for the increased durations. There are, however, some maintenance locations for which this does not hold. Most strinking in this respect is the location Zl, which is chosen in all of the scenarios for the standard duration but in none of the scenarios for the increased duration. This means that the decision whether or not to open Zl as a maintenance location is highly sensitive to the prevailing maintenance durations and should therefore be assessed with care.

Also, the day shares and costs over various choices for the maintenance durations are relevant. These figures are displayed in Table B.8 for those scenarios in which a feasible solution was found for both variants of maintenance durations.

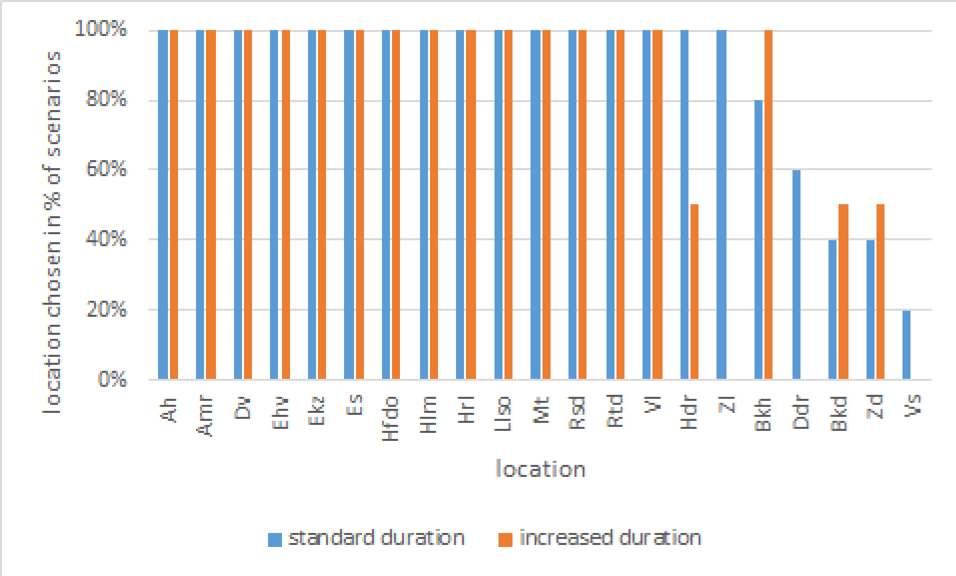



*Figure B.1: Comparison between the opened locations in the default scenario and the alternative scenario.*

.

| BDu id | standard duration hrs. activity | day share | cost | increased duration hrs. activity | day share | cost |
|---|---|---|---|---|---|---|
| 7 | 153.7 | 23.8% | € 6,953 | 223.1 | 13.0% | € 10,575 |
| 8 | 153.3 | 24.1% | € 6,923 | 223.1 | 13.7% | € 10,542 |
| 9 | 153.3 | 24.1% | € 6,927 | 223.3 | 13.8% | € 10,546 |
| 10 | 152.0 | 22.7% | € 6,912 | 221.7 | 12.7% | € 10,520 |
| mean | 153.1 | 23.7% | € 6,929 | 222.8 | 13.3% | € 10,546 |

*Table B.8: The mean hours of activity, day share and cost for both the standard maintenance durations and the increased maintenance durations. These are presented for those BDu ids (namely 7, 8, 9 and 10) for which in both situations (i.e. with both maintenance durations) a feasible solution could be obtained.*

It appears that the day shares are fairly consistent across the various BDus, but not across both choices of maintenance duration. It appears that the day share drops considerably when longer maintenance durations are considered. The expected explanation for this is that, due to increased maintenance durations, less suitable maintenance opportunities are available during daytime. As a result, less work can be performed during daytime and the day share drops.

Moreover, the toatl hours of activity increases. This is a result of the fact that the increased durations result in the fact that maintenance takes more time.

The costs are also much higher for increased durations compared to standard durations. Two reasons apply. First, the increased duration leads to more hours of activity in total, leading to higher costs. Second, the increased udration leads to less work during daytime, leading to higher costs.

### B.4.4 Experiment 8: Running times

The current experiment addresses the computation time of the **MSLCP**.

Based on the results for the scenarios in batch 1, the influence of the number of days in the analysis on the running time as well as the influence of the number of rolling stock units in the analysis on the running time can be investigated.

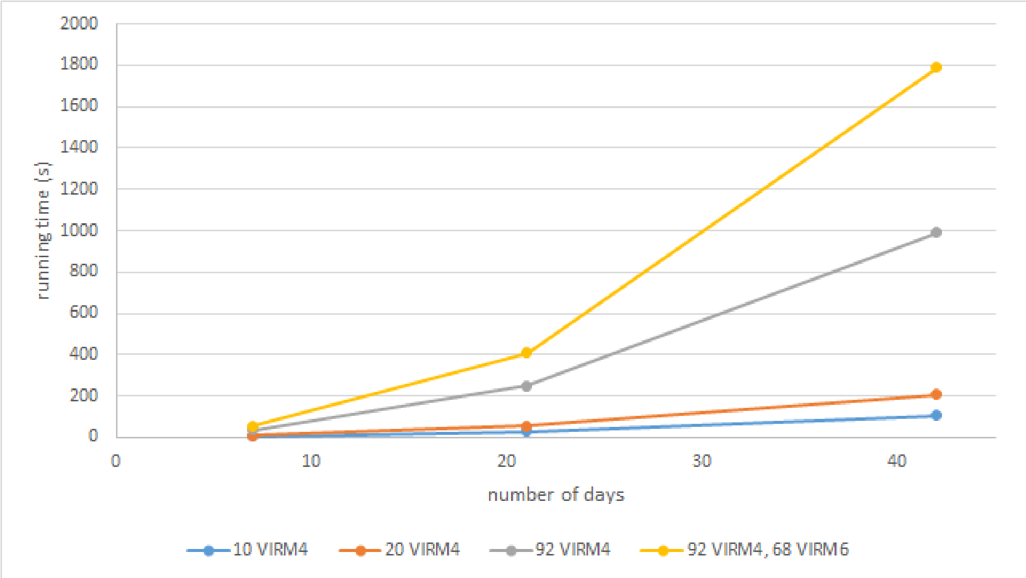



*Figure B.2: Influence of number of days in analysis on running time, for various numbers of rolling stock units considered.*

Figure B.2 reports the influence of the number of days in the analysis on the running time, for various numbers of rolling stock units. Both the length of the time horizon as the number of rolling stock appear to contribute significantly to the running time. It shows that for a time horizon of 42 days, the running time for the scenario with all rolling stock units of type VIRM4 and VIRM6 is (approximately) 1800 s (approximately 30 minutes). In many planning applications this is very acceptable. Note that, if the planning horizon is halved, to 21 days, the computation times are less than approximately 8 minutes.

The reported values in Figure B.2 are averaged over the scenarios with various values for $L^D_{max}$ and $\delta^D$. These parameters may in theory contribute to the problem's complexity and hence the running time. However, Figure B.3 demonstrates that the the parameter $L^D_{max}$ does not seem to influence the problem's complexity. Although not reported, the same holds for the value of $\delta^D$.

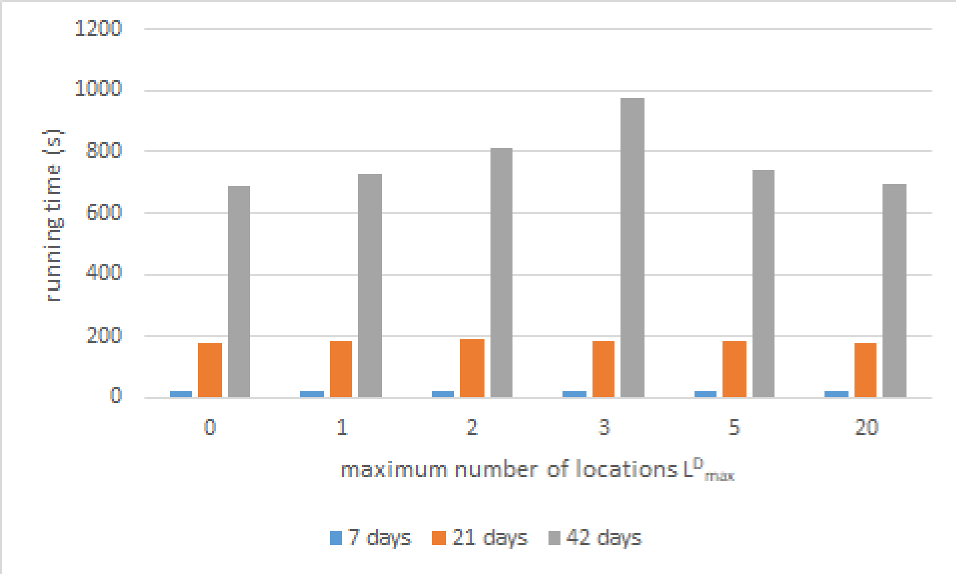



*Figure B.3: Influence of the maximum number of daytime maintenance locations $L^D_{max}$ on the running time.*

To be able to make some predictions on the running time beforehand, it may be worthwhile to look at the number of constraints and variables in the MIP formulation. These are typically the essential determinants of any problem's complexity. Figure B.4 demonstrates this relationship for all scenarios that could be solved in scenario batch 1. The running times may possibly increase exponentially with the number of constraints and variables.

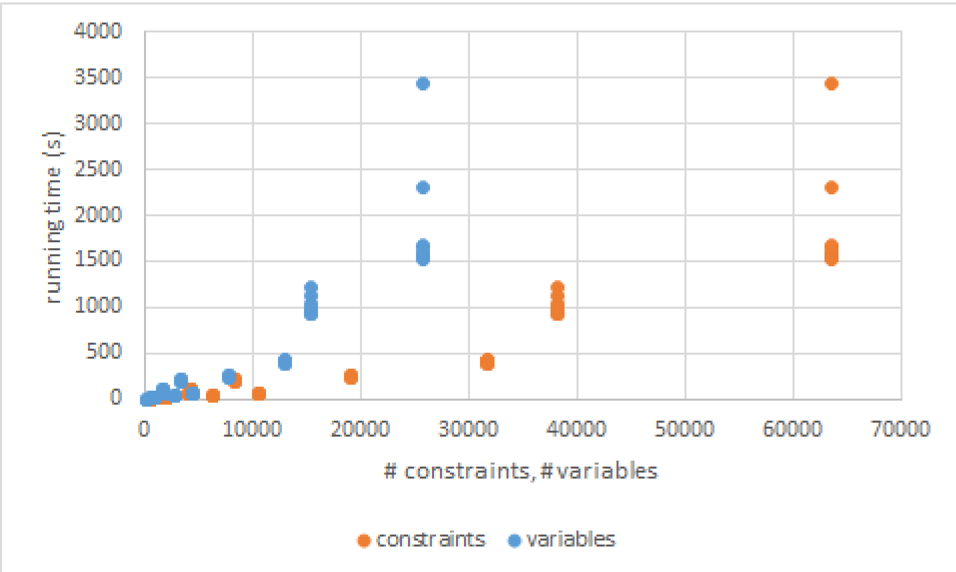



*Figure B.4: The running time in relation to the number of constraints and variables in the corresponding MIP formulation.*

The previously investigated scenarios contained at most all rolling stock units of type VIRM4 and VIRM6. However, in scenario batch 3 also an even larger scenario with all rolling stock units that are used for intercity lines is considered. The variant with $L^D_{max} = 30$ turned out to take the longest running time: 5089 seconds, approximately 1 hours and 25 minutes. For planning purposes, this may in many cases be considered an acceptable running time.

# Appendix C

# CMSLCP results

The current appendix provides additional **CMSLCP** results. Appendix C.1 presents a toy instance to help understand the workings of the **CMSLCP** and show that it functions appropriately. Appendix C.2 gives an overview of the capacities of the initial **MSLCP** solution considered in Chapter 8.

## C.1 APP demonstration on toy instance

To support the understanding of the **CMSLCP**, a small example is given of a toy instance in which the initial maintenance location capacity is violated, but where the **CMSLCP** finds a feasible solution in a next iteration. The naive cut generation method is used to generate cuts.

Table C.1 shows the jobs that need to be performed during this particular shift according to the initial **MSLCP** solution. It can be seen that jobs 3 and 4 both need to be performed between 13:22 and 14:48 and both take one hour. This cannot be performed by one maintenance team.

| job | mtnc. type A | mtnc. type B | release | deadline | duration |
|---|---|---|---|---|---|
| 1 | x | | 9:49 | 10:48 | 0.5 |
| 2 | | x | 13:12 | 16:48 | 1 |
| 3 | | x | 13:22 | 14:48 | 1 |
| 4 | | x | 13:22 | 14:48 | 1 |

*Table C.1: Jobs initially assigned to the a maintenance shift in the toy instance, resulting in a capacity violation.*

In the naive cut generation method, these four jobs together are added as a cut, which makes sure that in a next iteration of the **CMSLCP**, not all jobs 1-4 can be performed anymore. Table C.2 indicates the jobs assigned to the shift under consideration in the next iteration and shows that the initial job 4 has disappeared, meaning that the **MSCLP** solution in the second iteration does not assign maintenance to the maintenance shift corresponding to job 4 anymore.

| job | mtnc. type A | mtnc. type B | release | deadline | duration |
|---|---|---|---|---|---|
| 1 | x | | 9:49 | 10:48 | 0.5 |
| 2 | | x | 13:12 | 16:48 | 1 |
| 3 | | x | 13:22 | 14:48 | 1 |

*Table C.2: Jobs assigned to the maintenance shift in the toy instance after an iteration of the **CMSLCP**.*

Clearly, the new set of jobs (in Table C.2) can be performed by one team, and hence the **APP** results in a feasible solution. This solution is given in Table C.3. Since a solution has been found that satisfied the capacity constraints, the **CMSLCP** terminates.

| job | n | start time | end time |
|---|---|---|---|
| 1 | 1 | 9:49 | 10:19 |
| 2 | 1 | 15:48 | 16:48 |
| 3 | 1 | 13:48 | 14:48 |

*Table C.3: Final activity planning in the maintenance shift in the toy instance after running the* ***CMSLCP****.*

## C.2 Initial capacities

Table C.4 displays the initial capacities for the problem instance considered in Chapter 8. These capacities have been determined by applying the **APP** to each maintenance shift to which maintenance jobs were assigned. The **APP** was run using one and three maintenance team ($\overline{N} = 1, 3$). The former choice results in an infeasible solution for each maintenance shift that requires more than one maintenance team. The latter choice is able to detect up to three maintenance teams, which proves to be enough in the current case. The downside of the **APP** with three maintenance teams is that the running times are longer as a result of the higher number of variables in the optimization problem, although it becomes clear that in both cases the running times are lower than a few seconds and hence very acceptable.

| location | date | # jobs | $N = 1$ | | $N = 3$ | |
|---|---|---|---|---|---|---|
| | | | OF | time (s) | OF | time (s) |
| Alm | 10-04-2018 | 12 | inf | 0.16 | 3 | 0.34 |
| Alm | 11-04-2018 | 12 | inf | 0.16 | 2 | 0.33 |
| Alm | 12-04-2018 | 13 | inf | 0.16 | 2 | 0.54 |
| Alm | 13-04-2018 | 10 | inf | 0.14 | 2 | 0.38 |
| Alm | 14-04-2018 | 12 | inf | 0.15 | 2 | 0.38 |
| Alm | 15-04-2018 | 8 | inf | 0.13 | 2 | 0.25 |
| Alm | 16-04-2018 | 6 | inf | 0.12 | 2 | 0.26 |
| Bkh | 10-04-2018 | 5 | 1 | 0.14 | 1 | 0.19 |
| Bkh | 11-04-2018 | 8 | inf | 0.20 | 2 | 0.46 |
| Bkh | 12-04-2018 | 7 | 1 | 0.15 | 1 | 0.28 |
| Bkh | 13-04-2018 | 6 | 1 | 0.16 | 1 | 0.30 |
| Bkh | 14-04-2018 | 1 | 1 | 0.17 | 1 | 0.13 |
| Bkh | 15-04-2018 | 1 | 1 | 0.11 | 1 | 0.17 |
| Bkh | 16-04-2018 | 8 | 1 | 0.15 | 1 | 0.27 |
| Gn | 10-04-2018 | 12 | 1 | 0.24 | 1 | 0.73 |
| Gn | 11-04-2018 | 14 | inf | 0.17 | 2 | 1.02 |
| Gn | 12-04-2018 | 15 | 1 | 0.20 | 1 | 1.79 |
| Gn | 13-04-2018 | 24 | inf | 0.30 | 2 | 0.98 |
| Gn | 14-04-2018 | 13 | inf | 0.17 | 2 | 0.88 |
| Gn | 15-04-2018 | 14 | inf | 0.23 | 2 | 0.64 |
| Gn | 16-04-2018 | 11 | 1 | 0.19 | 1 | 1.14 |
| Lw | 10-04-2018 | 12 | inf | 0.20 | 2 | 0.34 |
| Lw | 11-04-2018 | 14 | inf | 0.20 | 2 | 0.48 |
| Lw | 12-04-2018 | 17 | 1 | 0.22 | 1 | 1.19 |
| Lw | 13-04-2018 | 14 | inf | 0.24 | 2 | 0.72 |
| Lw | 14-04-2018 | 13 | inf | 0.16 | 2 | 1.02 |
| Lw | 15-04-2018 | 13 | 1 | 0.18 | 1 | 0.54 |
| Lw | 16-04-2018 | 13 | inf | 0.22 | 2 | 0.56 |
| Zl | 10-04-2018 | 12 | inf | 0.27 | 2 | 2.68 |
| Zl | 11-04-2018 | 13 | inf | 0.21 | 3 | 2.44 |
| Zl | 12-04-2018 | 13 | inf | 0.27 | 2 | 2.77 |
| Zl | 13-04-2018 | 10 | inf | 0.28 | 2 | 0.53 |
| Zl | 14-04-2018 | 3 | 1 | 0.12 | 1 | 0.19 |
| Zl | 15-04-2018 | 1 | 1 | 0.11 | 1 | 0.15 |
| Zl | 16-04-2018 | 5 | 1 | 0.17 | 1 | 0.33 |

*Table C.4: Initially required number of maintenance teams for all daytime maintenance shifts, in a scenario with default maintenance durations and an **APP** model with 1 and 3 maintenance teams ($N = 1, 3$). Presented is the location and date of the shift, the number of jobs, the objective function (OF) value of the **APP** (or inf if it could not be computed) and the running time of the **APP** in seconds.*

# Appendix D

# Use case results

Chapter 9 presents a use case. In the generation of results for this use case, several scenarios have been tested. Not all have been presented in the main text.

In particular, two iterations of scenario generation have preceded the scenarios presented in the main text. Both she scenarios in the current section use short maintenance durations (of 30 minutes for type A and 60 minutes for type B) and are based on the small-scale version of the use case. The first scenario focuses on a scenario with the start of the daytime time window at 10.00. It is demonstrated that these settings result in undesirable assignments of maintenance activities to MOs. In the second scenario the start of the daytime time window is set at 07.00 and this problem is solved. However, this second iteration is still not desirable since it shows that assigned maintenance activities are too tight in their maintenance opportunities, leaving no slack time for operations.

Based on these two iterations, the choice was made to include in the main text scenarios with longer maintenance durations (45 minutes for type A and 90 minutes for type B) and a start of the daytime time window at 07.00.

The current appendix is structured as follows. Appendix D.1 presents the results for the first use case iteration and Appendix D.2 presents the results for the second use case iteration. In addition, this chapter contains some other extra information regarding the use case: Appendix D.3 gives details about the infeasibility of the rolling stock circulation of the large-scale use case scenario and Appendix D.4 gives a complete rolling stock circulation for 7 days and 24 rolling stock units, for reference.

## D.1 Iteration 1

One of the parameters in the use case is the start of the daytime time window. In Chapter 9 these are set to 07.00 and 19.00, respectively. However, in earlier iterations of the research process, the start of the daytime time window was set to 10.00. The rationale behind this first setting was that maintenance activities during daytime may only start after 10.00. A visualization of the resulting schedule for three rolling stock units and no locations for daytime maintenance is given in Figure D.2.

This figure indicates some problems related to setting the start of the daytime time window at 10.00. Consider for example Figure D.2a. It pictures the rolling stock circulation for a rolling stock unit for four subsequent days, with a maintenance activity of type B scheduled on the second day. This maintenance activity is planned in the maintenance opportunity in Rtd, starting on 9.55 and ending at 16.35. This maintenance opportunity is clearly during the day. However, recall that recall that a maintenance opportunity is classified to be during daytime if and only if it starts and ends during the daytime time window (on the same day). The maintenance opportunity under consideration starts at 9.55 and is therefore interpreted by the model as if it is a nighttime maintenance opportunity. This behaviour is undesirable since it does not correspond to realistic situations.

To choose better daytime time window parameters, notice the following. It may be better if the daytime window is chosen is such a way that maintenance opportunities are classified to be during daytime only if they start in or after the morning peak *and* end before or in the evening peak. Note that this implies that maintenance activities are classified nighttime only if either they start before the morning peak *or* end after the evening peak. This prevents the problem described above, where the maintenance opportunities that started *in* the morning peak (i.e. before 10.00) were classified as nighttime maintenance opportunities.

To choose better parameters for the daytime time window, it is helpful to investigate when the peak hour actually starts. The use of rolling stock units over the day is indicated in Figure D.1. The morning peak seems to be between approximately 07.00 and 09.00, and the evening peak seems to be between approximately 16.30 and 19.00. Therefore, in the scenarios used in Chapter 9, the start of the daytime window is adjusted to 07.00 and the start of the nighttime window remains unchanged at 19.00.

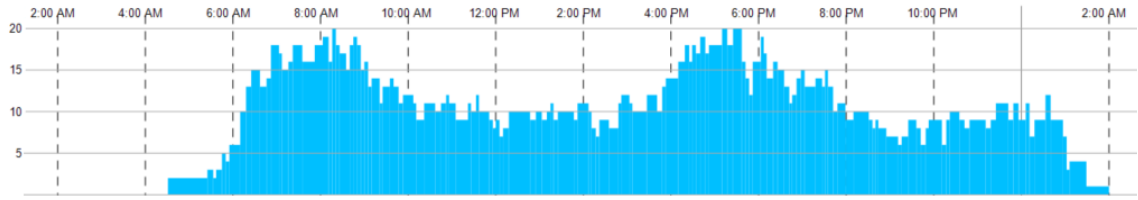
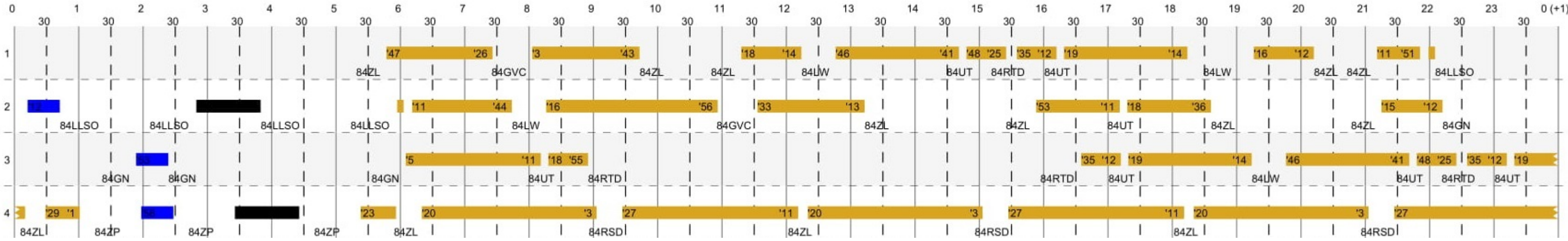
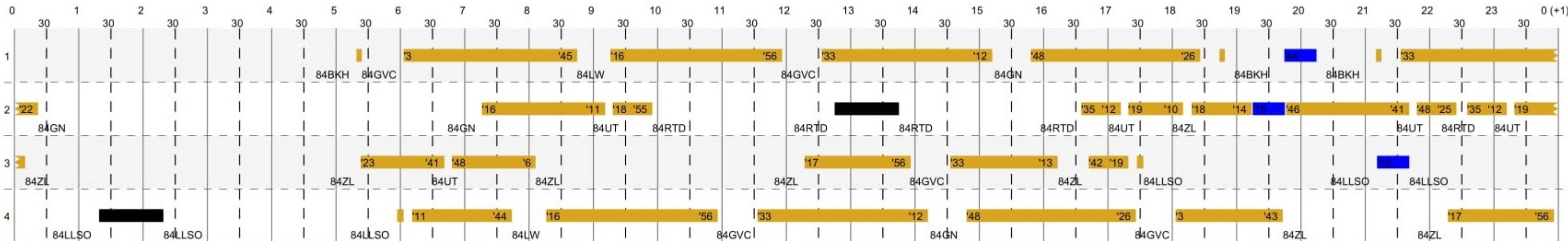



*Figure D.1: The number of rolling stock units in use over the day from 02.00 on 13-6-2020 until 02.00 on 14-6-2020 of the use case described in Section 9.2.1 (SMA, 2020b).*

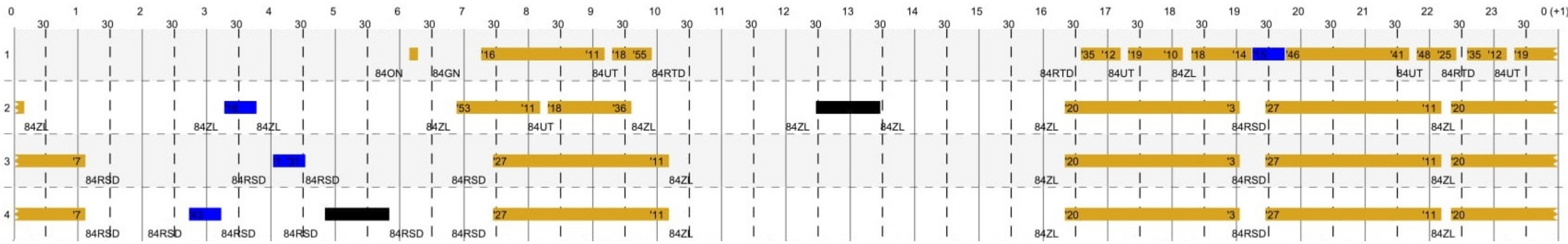
*(Text continues after figure.)*



(a) Rolling stock unit 1



(b) Rolling stock unit 2



(c) Rolling stock unit 3

*Figure D.2: Visualizations for an earlier iteration of the use case results (where the start of the daytime maintenance window starts at 10.00), of a maintenance schedule obtained by the **MSLCP** if no daytime maintenance is allowed. For each train, the rolling stock movements with location and time for each planned departure and arrival are indicated. The four rows presented correspond to the rolling stock circulation of a rolling stock unit for four subsequent days. The model determines MOs in which maintenance activities are executed, rather than their exact timing, and hence the visualized timing of maintenance activities should be interpreted with care. For reference, rolling stock units 1, 2 and 3 are listed in Viriato under vehicle id 3, 5 and 7, respectively.*

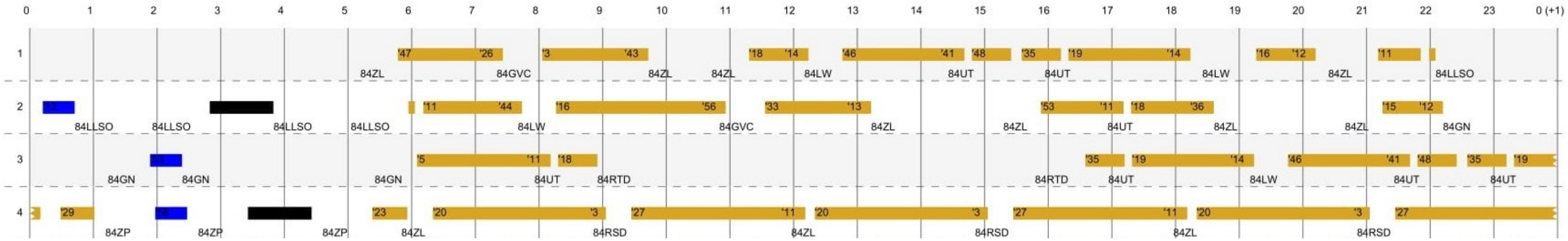
## D.2 Iteration 2

The second scenario uses short maintenance durations (30 minutes for type A and 60 minutes for type B) and a start of the daytime time window at 07.00. Results for this scenario are presented below. It becomes clear that, contrary to the first iteration presented in Section D.1, valid results can be produced: unlike the first iteration, maintenance activities are not anymore scheduled during daytime if the associated maintenance locations are not opened during daytie. However, this iteration unveils that many maintenance activities hardly fit in the maintenance opportunities, leaving no slack time during operations.

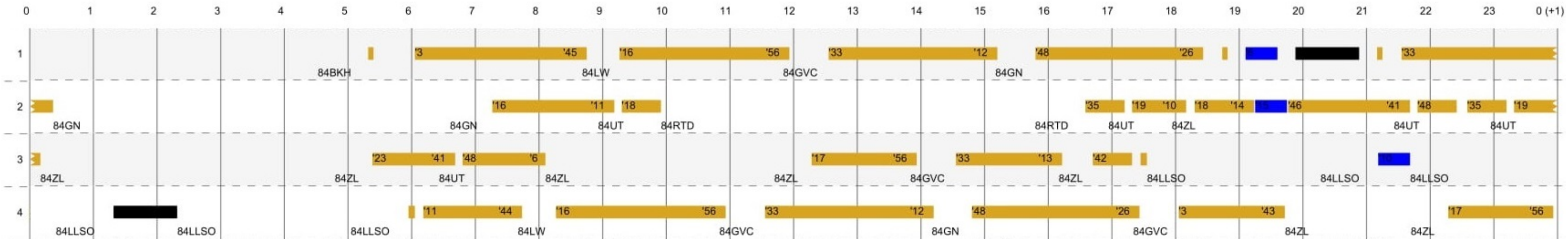
**Visualizations** Results were generated for 0, 1, 2, 3, 4 and 5 maintenance locations. Visualizations are given for the situation with 0 maintenance locations in Figure D.3 and for the situation with 3 maintenance locations in Figure D.5.

The current scenario with shorter maintenance durations is deemed to be less robust than the scenario with longer maintenance durations, used in the main text. In the current scenario, some maintenance activities were scheduled in such a way that it would just fit (see for example Figure D.5b, where the maintenance activities on day 1 and 4 in Gvc are used). In reality, however, this time may often be too short. Hence, taking into account extra time for maintenance resolves this problem, as the mentioned maintenance activities are not used in the second scenario anymore (see Figure 9.4b). Therefore the longer durations are used in the main text.

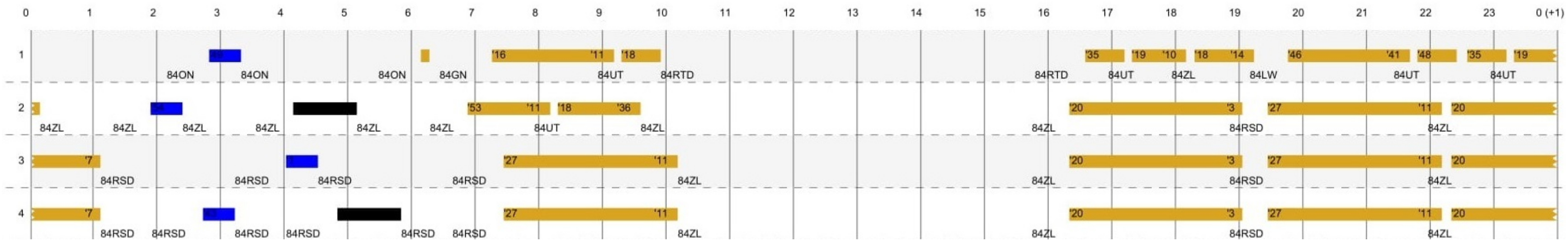
*(Text continues after figures.)*



*(a) Rolling stock unit 1*

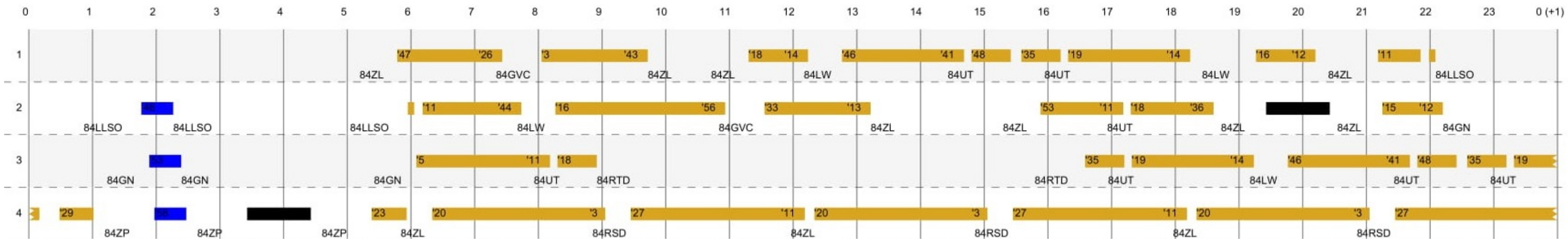


*(b) Rolling stock unit 2*

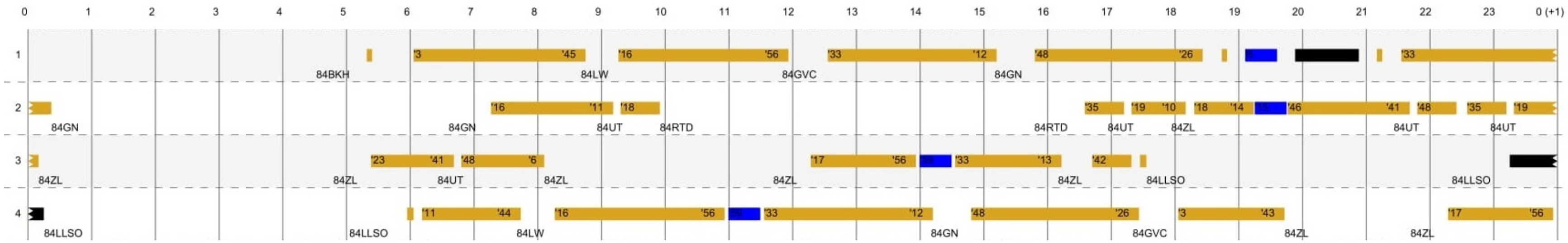


*(c) Rolling stock unit 3*

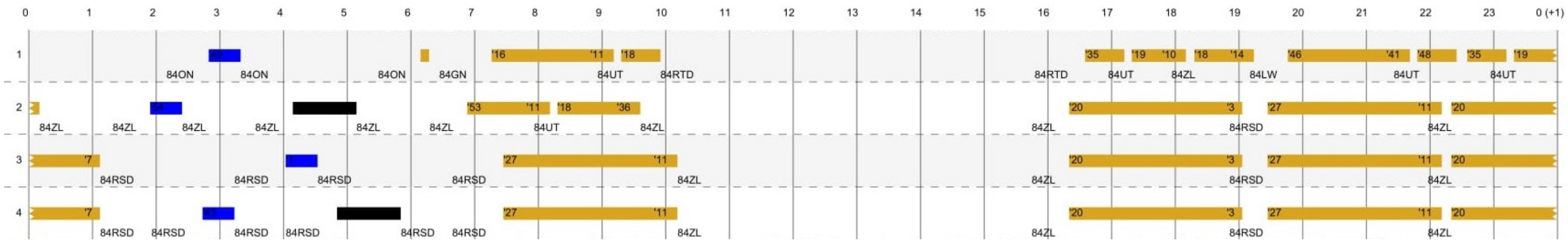
*Figure D.3: Visualizations for Tteration 2, of a maintenance schedule obtained by the* ***MSLCP*** *if no daytime maintenance is allowed. For each train, the rolling stock movements with location and time for each planned departure and arrival are indicated. The four rows presented correspond to the rolling stock circulation of a rolling stock unit for four subsequent days from 10 June 2019 to 13 June 2019. The model determines MOs in which maintenance activities are executed, rather than their exact timing, and hence the visualized timing of maintenance activities should be interpreted with care. For reference, rolling stock units 1, 2 and 3 are listed in Viriato under vehicle id 3, 5 and 7, respectively.*




*(a) Rolling stock unit 1*

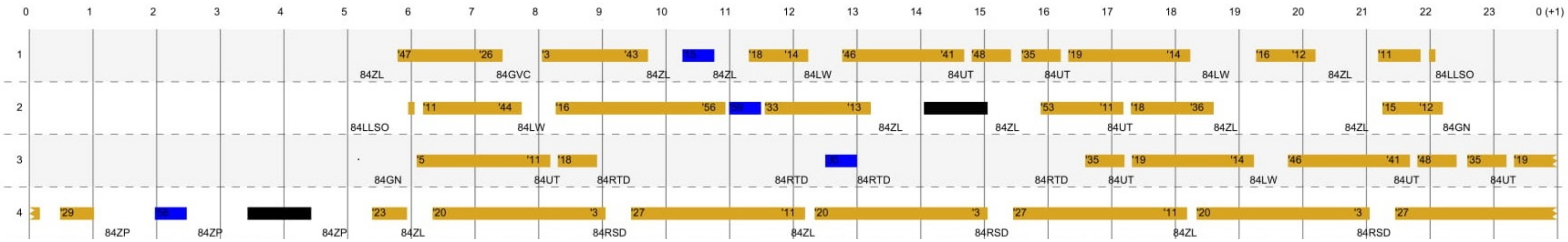



*(b) Rolling stock unit 2*

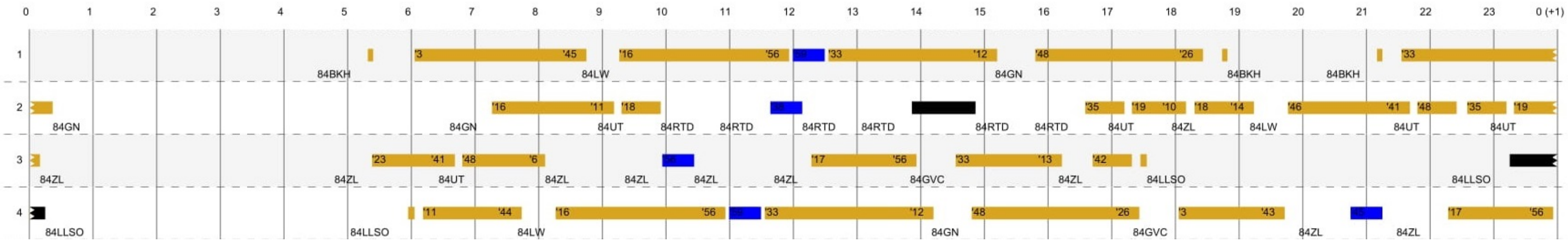



*(c) Rolling stock unit 3*

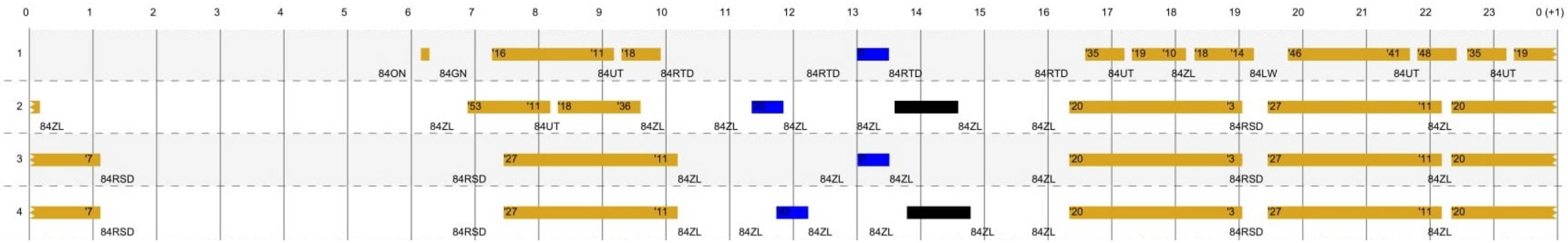
*Figure D.4: Visualizations for Iteration 2, of a maintenance schedule obtained by the **MSLCP**if one location can be opened for daytime maintenance. For each train, the rolling stock movements with location and time for each planned departure and arrival are indicated. The four rows presented correspond to the rolling stock circulation of a rolling stock unit for four subsequent days from 10 June 2019 to 13 June 2019. The model determines MOs in which maintenance activities are executed, rather than their exact timing, and hence the visualized timing of maintenance activities should be interpreted with care. For reference, rolling stock units 1, 2 and 3 are listed in Viriato under vehicle id 3, 5 and 7, respectively.*



(a) Rolling stock unit 1



(b) Rolling stock unit 2



(c) Rolling stock unit 3

*Figure D.5: Visualizations for Iteration 2, of a maintenance schedule obtained by the* ***MSLCP*** *if three locations can be opened for daytime maintenance. For each train, the rolling stock movements with location and time for each planned departure and arrival are indicated. The four rows presented correspond to the rolling stock circulation of a rolling stock unit for four subsequent days. The model determines MOs in which maintenance activities are executed, rather than their exact timing, and hence the visualized timing of maintenance activities should be interpreted with care. For reference, rolling stock units 1, 2 and 3 are listed in Viriato under vehicle id 3, 5 and 7, respectively.*

**KPIs** Table D.1 indicates the various locations opened in the current iteration of the use case, for each choice of the maximum number of daytime maintenance locations. It shows that if only one location can be opened, Gvc needs to be opened. If an additional location may be opened, Zl is a good choice (in addition to Gvc), then Rtd, and so forth.

| # locations | Gvc | Zl | Rtd | Bkh | Bkd |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | x | 0 | 0 | 0 | 0 |
| 2 | x | x | 0 | 0 | 0 |
| 3 | x | x | x | 0 | 0 |
| 4 | x | x | x | x | 0 |
| 5 | x | x | x | x | x |

*Table D.1: Locations opened in Scenario 1 for a varying number of maximum number of daytime maintenance locations that can be opened.*

Table D.2 gives for each solution the number of activities, the average hours of activity per day (both daytime and nighttime), the day share and the costs.

| | | | | costs (€) | |
|---|---|---|---|---|---|
| # locations | # activities | hrs. of activity | day share | optim. | cons. |
| 0 | 29.6 | 19.7 | 0.0% | 986 | 6,800 |
| 1 | 31.3 | 20.6 | 13.5% | 973 | 7,040 |
| 2 | 31.3 | 20.6 | 20.1% | 946 | 7,280 |
| 3 | 31.4 | 20.7 | 26.9% | 924 | 7,520 |
| 4 | 31.4 | 20.7 | 33.4% | 897 | 7,760 |
| 5 | 32.0 | 21.0 | 37.1% | 894 | 8,000 |
| mean | 31.2 | 20.5 | | | |

*Table D.2: KPIs for the six use case variants in Scenario 1. It shows for various choices for the maximum number of opened daytime maintenance locations the average number of activities per day, the average number of hours of activity per day, the percentage of hours of activity that are performed during the day, and the associated costs according to the optimistic method and the conservative method, defined in Section 9.2.4, in €.*

It can be observed in Table D.2 that the day share is increasing: when more locations can be opened during daytime, the proportion of hours of activity during daytime, relative to the total hours of activity, increases up to 21.4% for five locations opened. Notice that the largest increment occurs between the situation with no daytime maintenance and the situation with 1 location for maintenance.

Also, the costs have been computed according to the conservative method and the optimistic method. See Section 9.2.4 for a description on how these costs are calculated. It can be observed that the costs calculated according to the conservative method are much higher than in the optimistic method. The expected reason for this is that the computed solutions assign many 'small' workloads to maintenance locations. The occurrence of such small workloads contributes to a lot of costs in the conservative method. For example, if a small workload of 2 hours is assigned to a maintenance location on average, in total 8 hours need to be paid, leading to a tremendous cost increase. Moreover, although for an increasing number of daytime maintenance locations the costs in the optimistic method decrease a little (since relatively more hours of work are performed during daytime, and these hours are cheaper), the costs according to the conservative method increase. Two factors may contribute to this.

In the first place, some work may be moved to the day, but not enough to provide work to an employee for an entire day. Nonetheless, the costs are incurred for the entire day. In the second place, some work may be moved to the day, but this may not lead to a sufficiently large decrease of work during nighttime in order to lead to a decrease in employees needed during nighttime.

Moreover, the average number of activities per day is 31.2 and the average number of hours of activity per day is around 20.5. These numbers could be expected, in a similar way as indicated in Section 9.3.

The number of activities and the hours of activity appear to increase slightly with the maximum number of daytime maintenance locations. Since the goal is to minimize the number of nighttime maintenance activities, this may come at the cost of doing more maintenance activities in total since maintenance activities may be less efficiently spread. Clearly, still, the percentage of hours of daytime work increases up to 21.4 % for a scenario with 5 locations opened for daytime maintenance. Also, the average costs per day decreases as a result of the fact that more activities are carried out during daytime, which are cheaper.

Lastly, the workload distribution for various locations is indicated in Figure D.6.

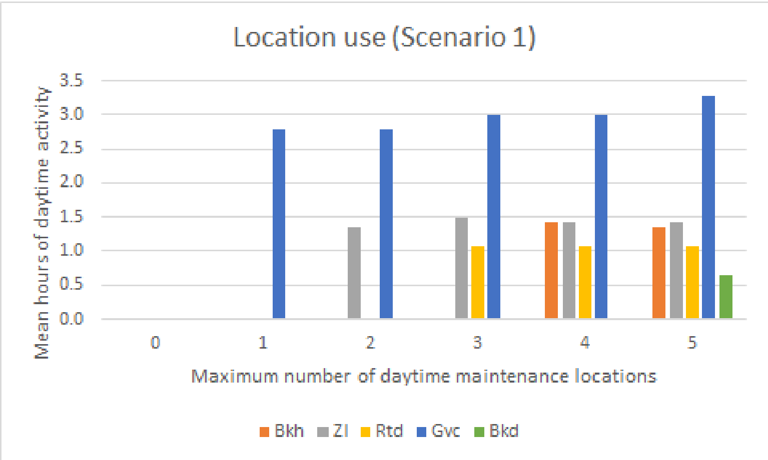



*Figure D.6: Hours of activity per location, averaged per day, for various values of the maximum number of locations for daytime maintenance, in Scenario 1.*

It shows that in Gvc approximately 2.5 hours of activity can be performed on average. Addition of the other locations results in a workload at these locations of less than an hour (on average). In many practical cases, these numbers are not considered to be substantial. However, it must be noted that the results presented here are only for the use case under consideration with 24 rolling stock units; more substantial workloads are expected when the entire network were considered.

## D.3 Infeasible rolling stock circulation

The large-scale scenario of the use case in Chapter 9 unveiled an occurrence of a rolling stock circulation for one particular rolling stock for which no feasible maintenance schedule could be made. D.3 displays an excerpt of the rolling stock circulation for this rolling stock unit for one day.

| id | departure | | | arrival | | | MO duration |
|---|---|---|---|---|---|---|---|
| | location | date | time | location | date | time | |
| 1 | Ut | 12-Jun | 19:03 | Rtd | 12-Jun | 19:40 | n/a |
| 2 | Rtd | 13-Jun | 0:56 | Ut | 13-Jun | 2:58 | 5:16 |
| 3 | Ut | 13-Jun | 3:11 | Gv | 13-Jun | 4:37 | 0:13 |
| 4 | Gv | 13-Jun | 4:44 | Gd | 13-Jun | 5:03 | 0:07 |
| 5 | Gd | 13-Jun | 5:10 | Rtd | 13-Jun | 5:26 | 0:07 |
| 6 | Rtd | 13-Jun | 6:05 | Ut | 13-Jun | 6:42 | 0:39 |
| 7 | Ut | 13-Jun | 6:49 | Gn | 13-Jun | 8:42 | 0:07 |
| 8 | Gn | 13-Jun | 9:18 | Ut | 13-Jun | 11:11 | 0:36 |
| 9 | Ut | 13-Jun | 11:18 | Rtd | 13-Jun | 11:55 | 0:07 |
| 10 | Rtd | 13-Jun | 12:05 | Ut | 13-Jun | 12:42 | 0:10 |
| 11 | Ut | 13-Jun | 12:49 | Gn | 13-Jun | 14:42 | 0:07 |
| 12 | Gn | 13-Jun | 15:18 | Ut | 13-Jun | 17:11 | 0:36 |
| 13 | Ut | 13-Jun | 17:18 | Rtd | 13-Jun | 17:55 | 0:07 |
| 14 | Rtd | 13-Jun | 18:05 | Ut | 13-Jun | 18:42 | 0:10 |
| 15 | Ut | 13-Jun | 18:49 | Zl | 13-Jun | 19:40 | 0:07 |
| 16 | Zl | 13-Jun | 19:45 | Gn | 13-Jun | 20:42 | 0:05 |
| 17 | Gn | 13-Jun | 21:18 | Ut | 13-Jun | 23:11 | 0:36 |
| 18 | Ut | 13-Jun | 23:18 | Rtd | 13-Jun | 23:55 | 0:07 |
| 19 | Rtd | 14-Jun | 0:05 | Ut | 14-Jun | 0:42 | 0:10 |
| 20 | Ut | 14-Jun | 0:49 | Amf | 14-Jun | 1:02 | 0:07 |
| 21 | Amf | 14-Jun | 1:07 | Bkd | 14-Jun | 1:10 | 0:05 |
| 22 | Bkd | 14-Jun | 5:32 | Amf | 14-Jun | 5:36 | 4:22 |

*Table D.3: Excerpt of the rolling stock circulation in the large-scale version of the use case. Each row represents a productive trip, with information about its departure and arrival. Between productive trips, maintenance opportunities are present. The duration of these maintenance opportunities are given in the last column.*

It can be observed that there are two long maintenance opportunities, one with a duration of 5 hours and 16 minutes and one with a duration of 4 hours and 22 minutes. These are the only maintenance opportunities that are able to fit a maintenance activity of Type A (with a duration of 45 minutes). The first maintenance opportunity ends at 13 June at 0:56. If maintenance of Type A would be scheduled in this maintenance opportunity, the next maintenance activity of Type A would need to be scheduled in a maintenance opportunity that starts less than 24 hours later. This means that the next maintenance opportunity would need to start before 0:56 on 14 June. However, as becomes clear from the current table, the next suitable maintenance opportunity of sufficient length starts at 1:10 in Bkd, which is too late. This example shows that the **MSLCP** tool can be used to quickly identify whether a feasible maintenance schedule can be made for a given rolling stock circulation.

However, to be able to still produce results, the rolling stock unit causing the infeasibility is left out of the analysis. In the current case, the infeasibility could be attributed to the rolling stock circulation of only one rolling stock unit. In other cases, the rolling stock circulation of multiple rolling stock units may be infeasible, which then all need to be left out of the analysis to be able to produce results. Note that this relates to adjusting the rolling stock circulation in order to be able to produce desired results, which occurs often in practice. A software tool like Viriato facilitates to do this in a graphical user interface, which may be beneficial and convenient in practical situations.

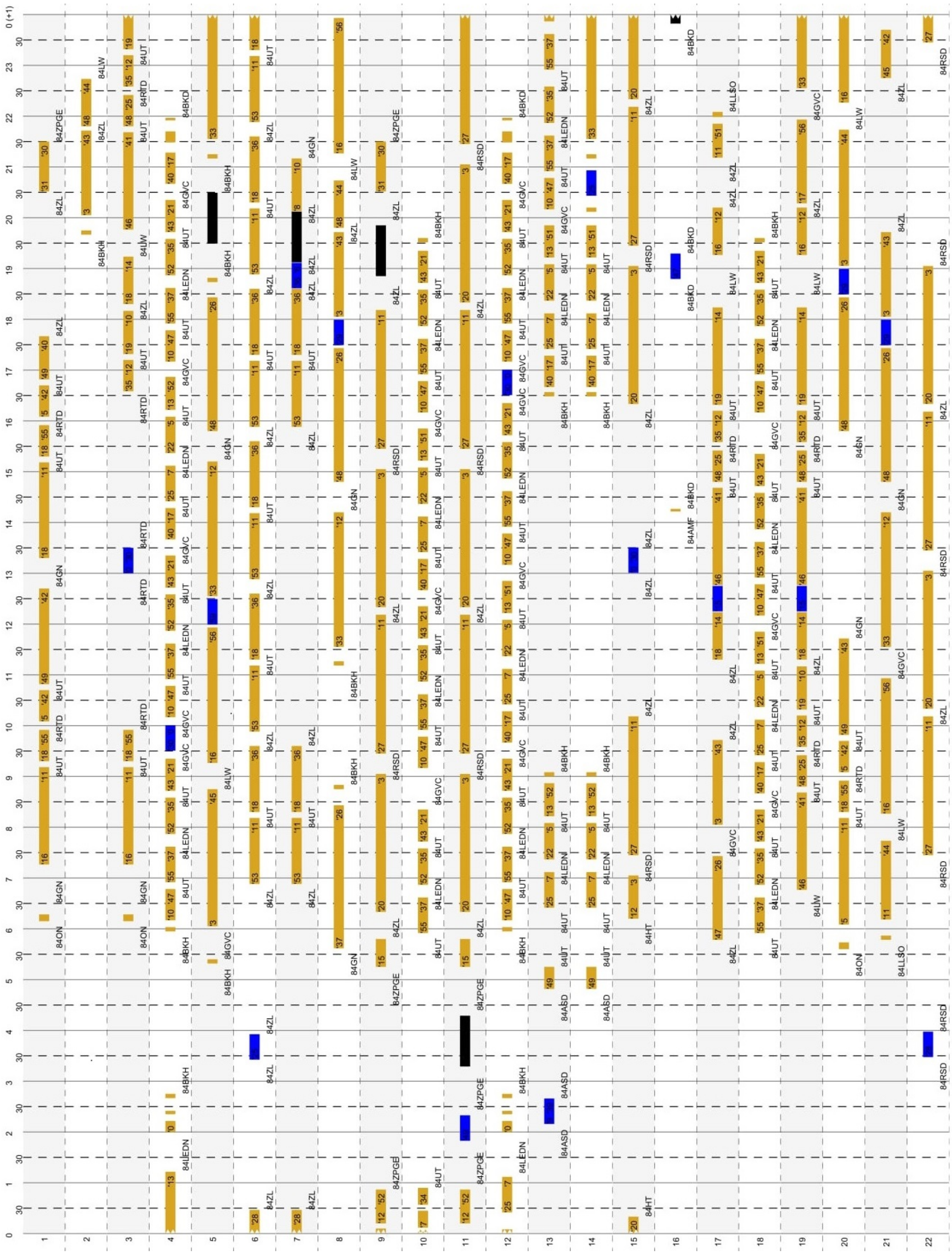
## D.4 Complete rolling stock circulation

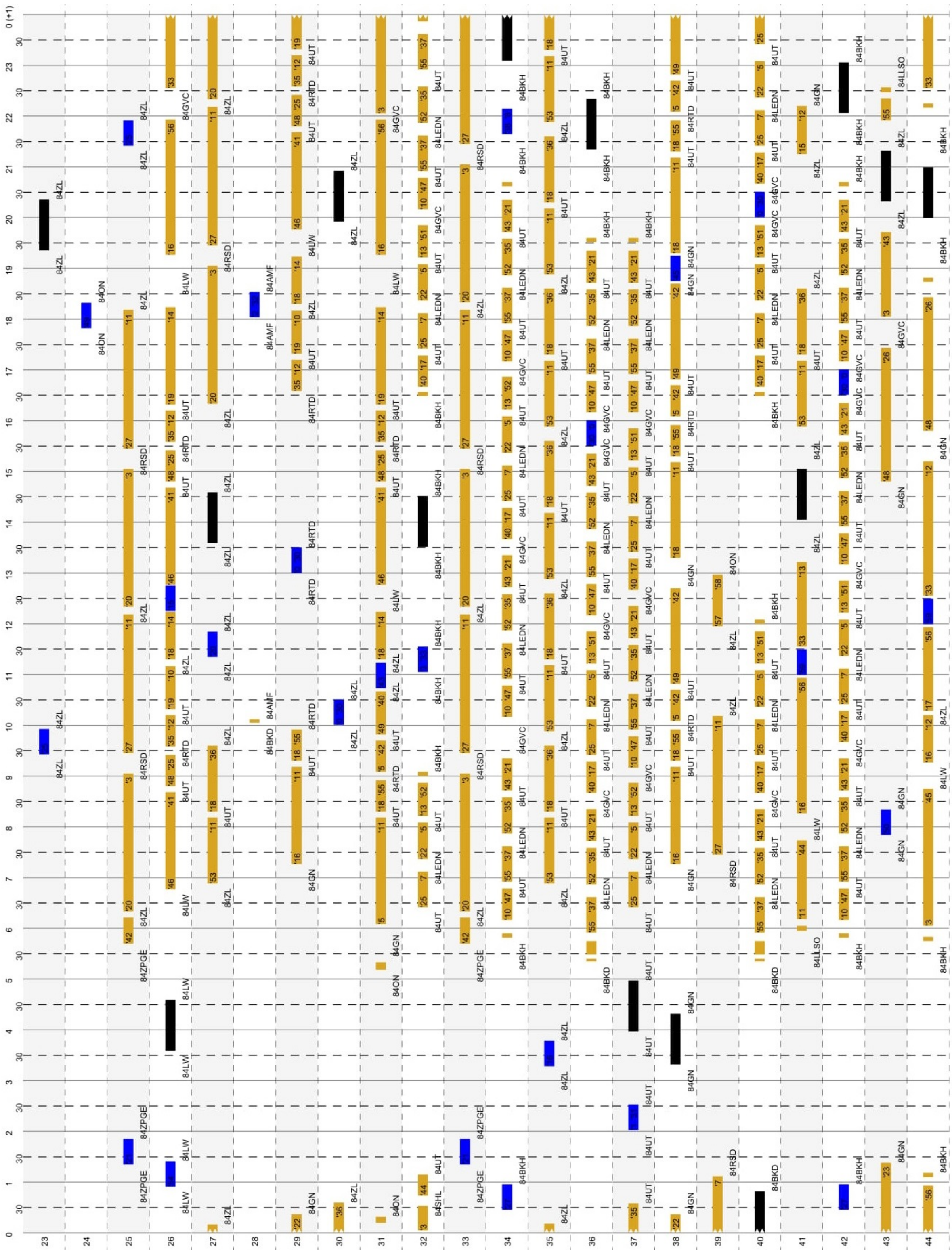
In Chapter 9, visualizations for rolling stock circulations were presented for various values of the maximum number of daytime maintenance locations $L^D_{max}$. For readability purposes, only 3 rolling stock units for 4 days were presented. However, as a reference, on the following pages the visualization of the complete rolling stock circulation for all 24 rolling stock units and 7 days in the analysis is presented for $L^D_{max} = 3$. This visualization is produced by Viriato.

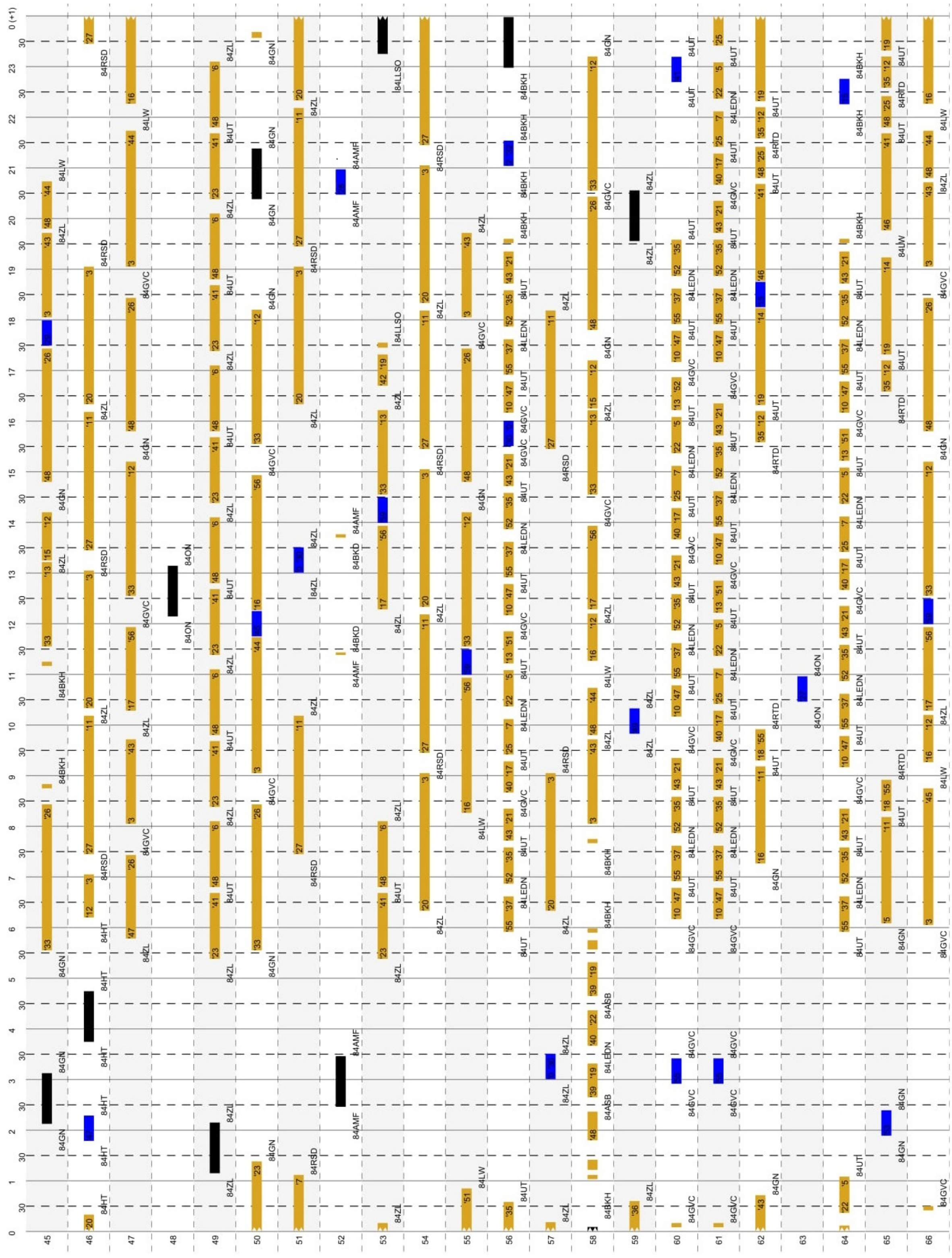
The visualization should be interpreted as follows. Each row corresponds to a number of activities for one rolling stock unit for a specific day. To also follow rolling stock units over multiple days, Table D.4 needs to be used. This table indicates for each rolling stock unit which row corresponds to a specific date.

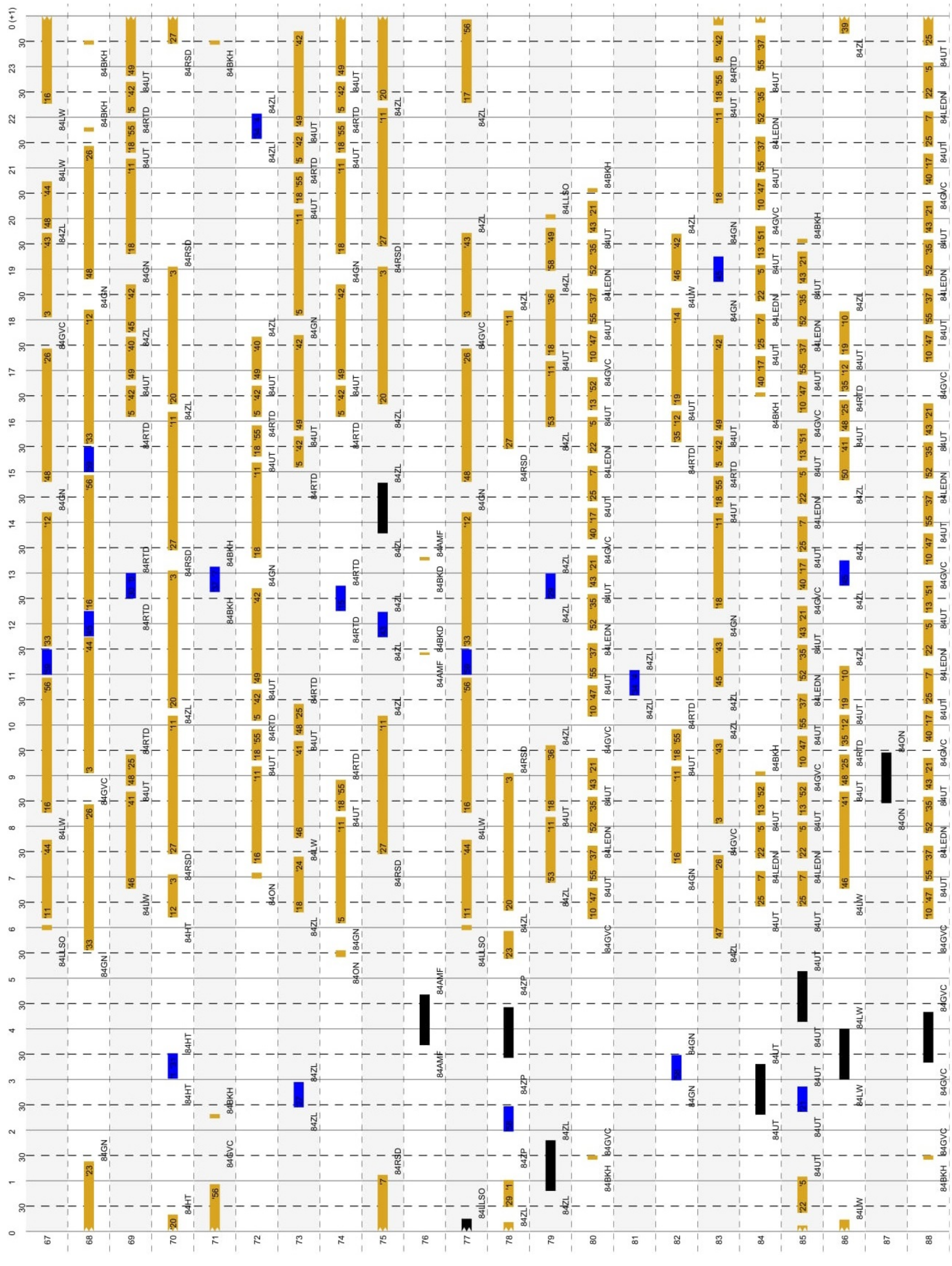
| veh. | 10-Jun | 11-Jun | 12-Jun | 13-Jun | 14-Jun | 15-Jun | 16-Jun |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 26 | 50 | 74 | 97 | 121 | 144 |
| 2 | 9 | 33 | 57 | 81 | 104 | 127 | 151 |
| 3 | 17 | 41 | 65 | 89 | 112 | 135 | 158 |
| 4 | 18 | 42 | 66 | 90 | 113 | 136 | 159 |
| 5 | 5 | 29 | 53 | 77 | 100 | 124 | 147 |
| 6 | 12 | 36 | 60 | 84 | 107 | 130 | 153 |
| 7 | 3 | 27 | 51 | 75 | 98 | 122 | 145 |
| 8 | 19 | 43 | 67 | 91 | 114 | 137 | 160 |
| 9 | 16 | 40 | 64 | 88 | 111 | 134 | 157 |
| 10 | 13 | 37 | 61 | 85 | 108 | 131 | 154 |
| 11 | 11 | 35 | 59 | 83 | 106 | 129 | 152 |
| 12 | 14 | 38 | 62 | 86 | 109 | 132 | 155 |
| 13 | 8 | 32 | 56 | 80 | 103 | 124 | 150 |
| 14 | 10 | 34 | 58 | 82 | 105 | 128 | |
| 15 | 20 | 44 | 68 | 92 | 115 | 138 | 161 |
| 16 | 1 | 25 | 49 | 73 | 96 | 120 | 143 |
| 17 | 6 | 30 | 54 | 78 | 101 | 125 | 148 |
| 18 | 21 | 45 | 69 | 93 | 116 | 139 | 161 |
| 19 | 15 | 39 | 63 | 87 | 110 | 133 | 156 |
| 20 | 7 | 31 | 55 | 79 | 102 | 126 | 149 |
| 21 | 4 | 28 | 52 | 76 | 99 | 123 | 146 |
| 22 | 22 | 46 | 70 | 46 | 117 | 140 | 162 |
| 23 | 23 | 47 | 71 | 94 | 118 | 141 | 163 |
| 24 | 24 | 48 | 72 | 95 | 119 | 142 | 164 |

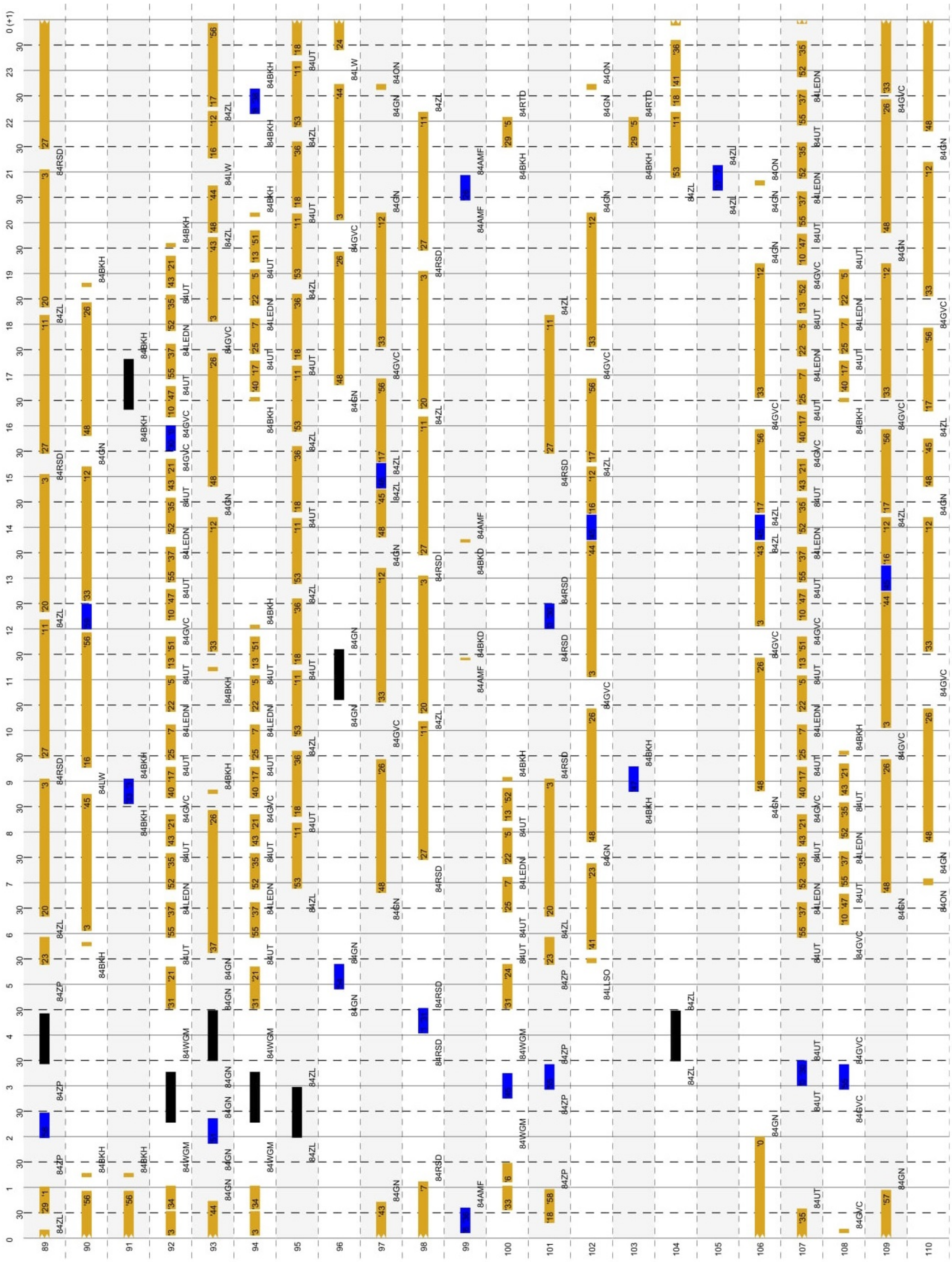
*Table D.4: Conversion table to interpret rolling stock circulation visualization from Viriato. Each row concerns a different vehicle. The columns refer to a day in June 2019. Each cell indicates the corresponding row of the rolling stock visualization in Appendix D.4 for a specific rolling stock unit on a specific day. For example, the rolling stock circulation for the first rolling stock unit is found on the subsequent pages in row 2 for June 10, 2019 and in row 26 on June 11, 2019.*

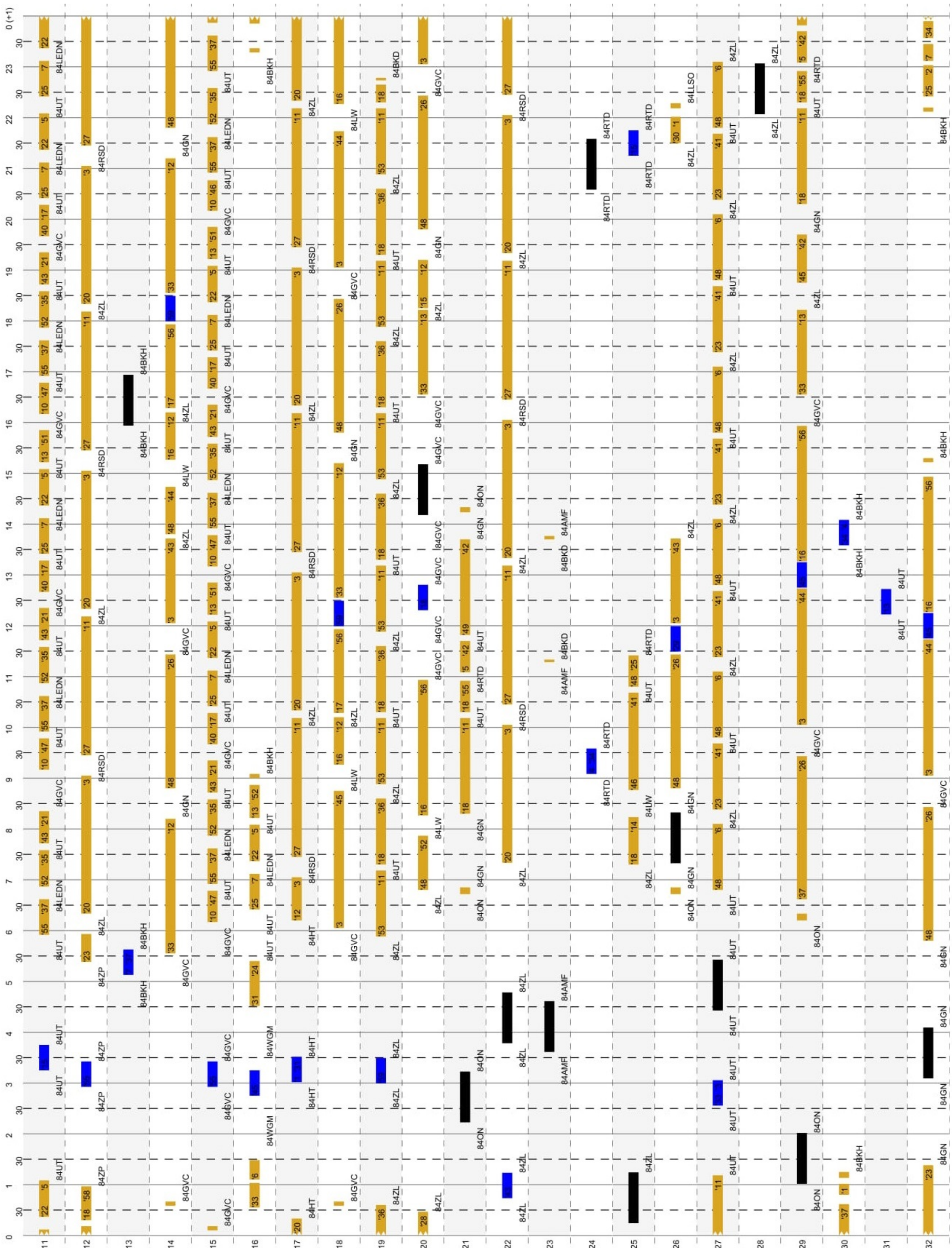


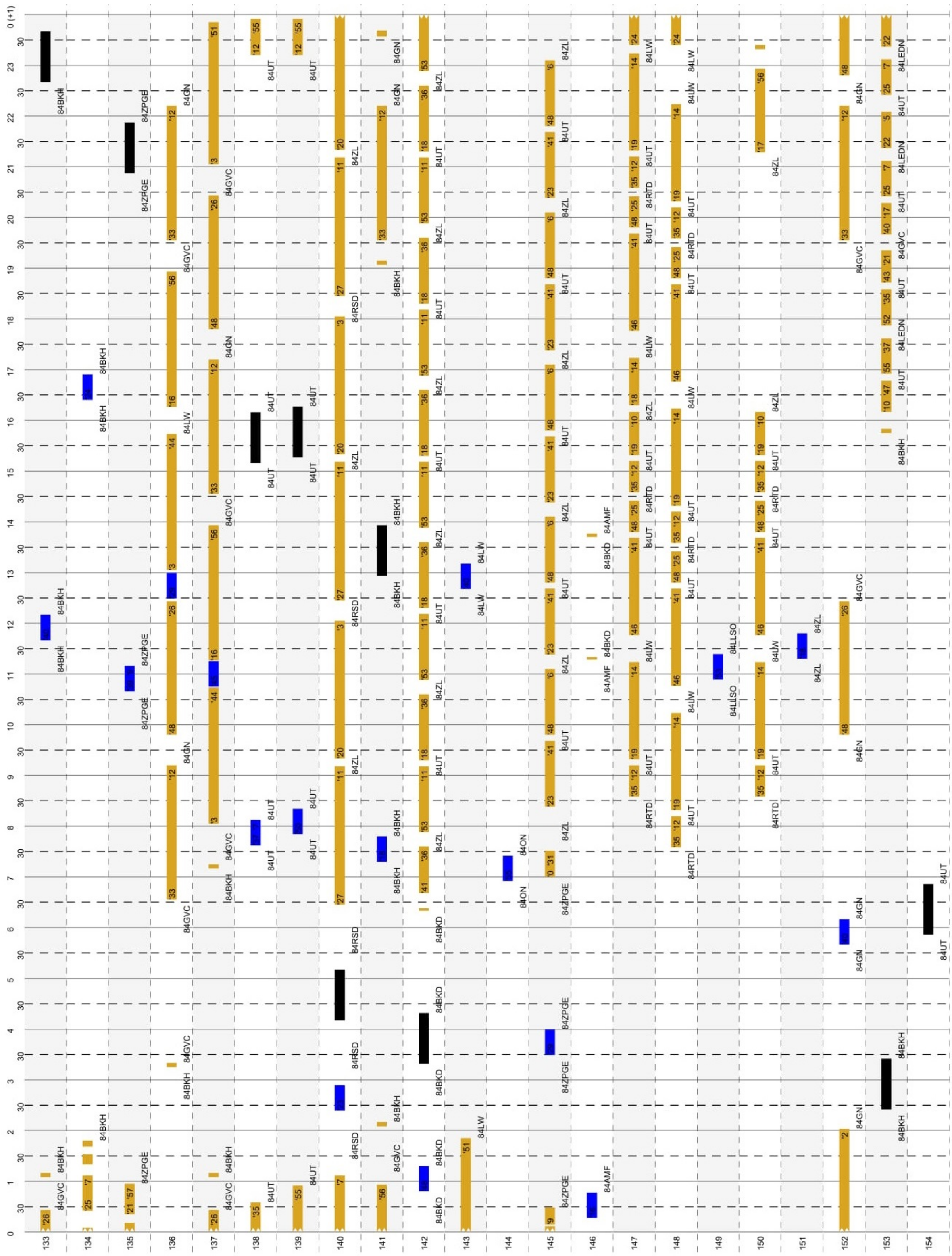


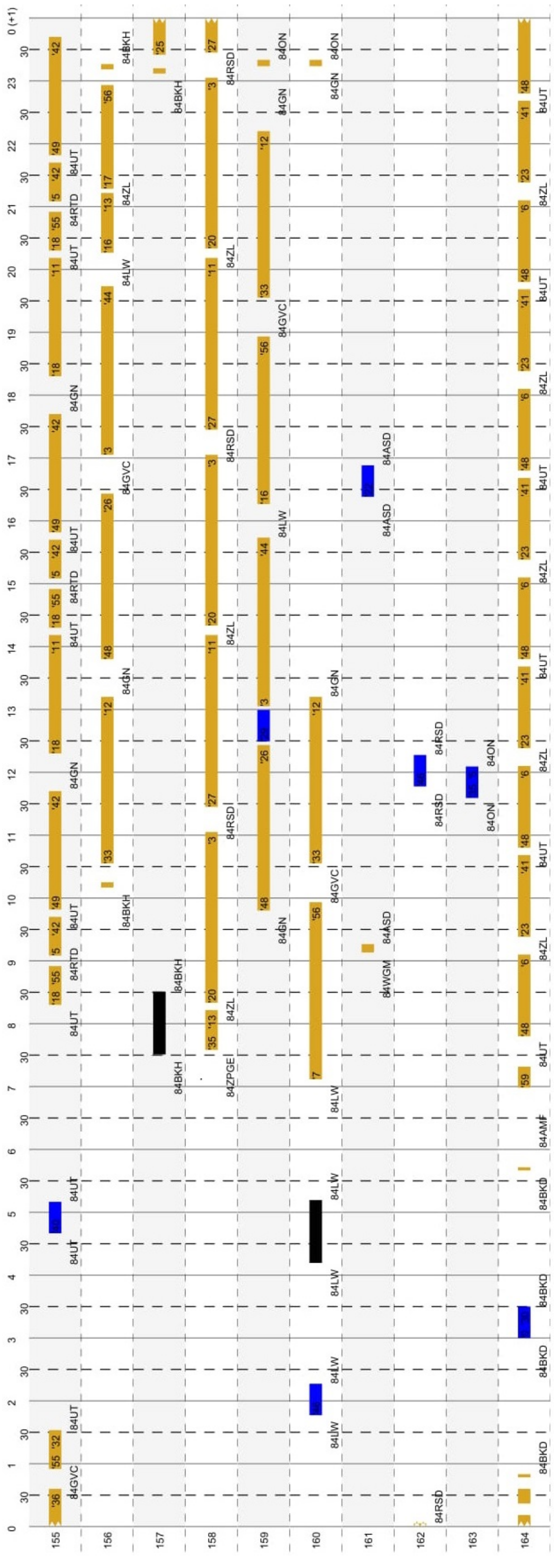












# Bibliography


Andrés, J., Cadarso, L., and Marín, (2015). Maintenance scheduling in rolling stock circulations in rapid transit networks. *Transportation Research Procedia*, 10(July):524–533.

Benders, J. (1962). Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4:238–252.

Blanc, P. (2020). Personal interview concerning maintenance processes at SNCF Voyages. Interview held on 18-2-2020. Summary given in the Appendix of the current thesis.

Bonne, H. (2020). Personal interview concerning maintenance processes at NMBS. Interview held on 31-1-2020. Summary given in the Appendix of the current thesis.

Canca, D. and Barrena, E. (2018). The integrated rolling stock circulation and depot location problem in railway rapid transit systems. *Transportation Research Part E: Logistics and Transportation Review*, 109(May 2017):115–138.

Clarke, L., Johnson, E., Nemhauser, G., and Zhu, Z. (1997). The aircraft rotation problem. *Annals of Operations Research*, 69:33–46.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2009). *Introduction to algorithms*. MIT press.

Deutsche Bahn (2018). Deutsche Bahn Integrierter Bericht 2018. https://www.deutschebahn.com/resource/blob/4045194/462384b76cf49fe8e c715f41e4a3202a/19-03-IB-data.pdf . Retrieved on 28-1-2020.

Dinmohammadi, F., Alkali, B., and Shafiee, M. (2016). A risk-based model for inspection and maintenance of railway rolling stock. In *European Safety and Reliability Conference (ESREL)*, pages 2016–29.

Eurostat (2020). Total length of railway lines. https://ec.europa.eu/eurostat/web/products-datasets/product?code=ttr00003. Retrieved on 23-1-2020.

Feo, T. A. and Bard, J. F. (1989). Flight Scheduling and Maintenance Base Planning. *Management Science*, 35(12):1415–1432.

Ford, L. and Fulkerson, D. (1956). Maximal Flow through a Network. *Canadian Journal of Mathematics*, 8:399–404.

Gopalan, R. (2014). The aircraft maintenance base location problem. *European Journal of Operational Research*, 236(2):634–642.

Gopalan, R. and Talluri, K. T. (1998). The aircraft maintenance routing problem. *Operations Research*, 46(2):260–271.

gov.uk (2020). Countries in the EU and EEA. https://www.gov.uk/eu-eea.

Hagberg, A., Swart, P., and S Chult, D. (2008). Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States).

Herr, N., Nicod, J.-M., Varnier, C., Zerhouni, N., and Malek Cherif, F. (2017). F., Joint optimization of train assignment and predictive maintenance scheduling. In *7th International Conference on railway operations modelling and Analysis (RailLille 2017)*, number April, pages 699–708.

Holtzer, J. (2020). E-mail correspondence about the objectives of NS. E-mail correspondence taken place on 30-6-2020.

Hooker, J. (2011). *Logic-based methods for optimization: combining optimization and constraint satisfaction*, volume 2. John Wiley & Sons.

Hooker, J. N. (2019). Logic-based Benders' decomposition for large-scale optimization. In *Large Scale Optimization in Supply Chains and Smart Manufacturing*, pages 1–26. Springer.

Huizingh, E. H. (2018). Planning first-line services on ns service locations. University of Twente. Master thesis.

International Union of Railways (2018). Railways, passengers carried. https://data.worldbank.org/indicator/IS.RRS.PASG.KM . Retrieved on 23-1-2020.

Janssens, S. (2017). Empirical analysis of service locations at NedTrain. Master thesis. Delft University of Technology.

Kravchenko, S. A. and Werner, F. (2009). Minimizing the number of machines for scheduling jobs with equal processing times. *European Journal of Operational Research*, 199(2):595–600.

Kuhlmann, H. (2020). Personal interview concerning maintenance processes at DB Regio Bavaria. Interview held on 18-2-2020. Summary given in the Appendix of the current thesis.

Maróti, G. and Kroon, L. (2007). Maintenance routing for train units: The interchange model. *Computers and Operations Research*, 34(4):1121–1140.

Mooren Ceng, F. P. and Van Dongen, L. A. (2013). Application of remote condition monitoring in different rolling stock life cycle phases. *Procedia CIRP*, 11:135–138.

NS (2017). Uitvoeren technische controles. Document TEC-4.001, NedTrain.

NS (2018a). Ns jaarverslag 2018. https://www.nsjaarverslag.nl/FbContent.ashx/pub_1001/downloads/v190228091452/NS-jaarverslag-2018.pdf . Retrieved on 28-1-2020.

NS (2018b). VVGB NedTrain. Various Microsoft Access data sheets containing all planned rolling stock movements for NS, for 10 different periods of approximately 8 weeks each, with start dates between 10-12-2017 and 9-6-2019.

NS (2019). Locaties langs het spoor. https://www.ns.nl/over-ns/treinonderhoud/locaties.html. Retrieved on March 4, 2020.

NS (2019). Specificaties dienstvoorzieningen, status augustus 2019. https://www.ns.nl/binaries/_ht_1568013570478/content/assets/ns-nl/over-ns/specificaties-dienstvoorzieningen.pdf. Retrieved on 05-03-2020.

NS (2020). EBITDA determinants for NS.

opendatasoft.com (2020). Trainline.EU - European Train Stations. https://public.opendatasoft.com/explore/dataset/european-train-stations/table/ , filtered on suggestable train stations only. Retrieved on 23-1-2020.

Rahmaniani, R., Crainic, T. G., Gendreau, M., and Rei, W. (2017). The Benders' decomposition algorithm: A literature review. *European Journal of Operational Research*, 259(3):801–817.

Sarac, A., Batta, R., and Rump, C. M. (2006). A branch-and-price approach for operational aircraft maintenance routing. *European Journal of Operational Research*, 175(3):1850–1869.

SMA (2016). Viriato - Software for railways. https://www.sma-partner.com/images/Downloads/Viriato_E_160907.pdf . Retrieved on 4-5-2020.

SMA (2020a). Rostering Algorithm Execution in Viriato.

SMA (2020b). Viriato. Vehicle statistics obtained from Viriato on 13-5-2020.

SMA (2020c). Viriato training - Zürich, 05.02.2020. Presentation slides from a Viriato training on 5-2-2020 issued by SMA.

Taha, H. A. (2011). *Operations research: an introduction*, volume 790. Pearson/Prentice Hall Upper Saddle River, NJ, USA.

Tönissen, D. D. and Arts, J. J. (2018). Economies of scale in recoverable robust maintenance location routing for rolling stock. *Transportation Research Part B: Methodological*, 117:360–377.

Tönissen, D. D., Arts, J. J., and Shen, Z.-J. M. (2019). Maintenance Location Routing for Rolling Stock Under Line and Fleet Planning Uncertainty. *Transportation Science*, 53(5):1252–1270.

Treinposities.nl (2020). Vertrektijden. https://treinposities.nl/vertrektijden . Retrieved on 23-1-2020.

Van Hövell, M. (2019). Increasing the Effectiveness of the Capacity Usage at Rolling Stock Service Locations.

Voet, T. (2020). Information provided by Tijmen Voet, Head of Planning. Retrieved on 24-2-2020.

Wagenaar, J. and Kroon, L. G. (2015). Maintenance in Railway Rolling Stock Rescheduling for Passenger Railways. *SSRN Electronic Journal*, pages 1–38.

Zomer, J. (2019). Verkenning mogelijkheden overdag behandelen. Technical Report September, NS.