ERWIN ABBINK

Crew Management in Passenger Rail Transport

Crew Management in Passenger Rail Transport

Crew Management in Passenger Rail Transport

Plannen en bijsturen van rijdend personeel bij de spoorwegen

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Erasmus Universiteit Rotterdam
op gezag van de
rector magnificus

Prof.dr. H.A.P. Pols

en volgens besluit van het College voor Promoties.

De openbare verdediging zal plaatsvinden op
vrijdag 24 oktober om 13.30 uur

door

ERWIN JAN WILLEM ABBINK
geboren te Ede

ERASMUS UNIVERSITEIT ROTTERDAM

Promotiecommissie

Promotoren: Prof.dr. L.G. Kroon

Prof.dr. A.P.M. Wagelmans

Overige leden: Prof.dr.ir. J.W.M. Bertrand

Prof.dr.ir. H.W.G.M. van Heck

Prof.dr. G.T. Timmer

Co-Promotor: Dr. D. Huisman.

Acknowledgement

This thesis summarizes the research I have been working on in the past 17 years at Netherlands Railways. In these years I have worked with many people, who all contributed to this work and made it a pleasure doing this research.

First and foremost, I want to express my gratitude to my colleagues and supervisors, Leo Kroon, Dennis Huisman. Leo, thank you for initiating this research, your thorough reviews of this thesis and for being patient when there was no progress in my writing. I have good memories of the start of this research when you shared your knowledge on algorithms and modeling and played an essential role in the research presented in this thesis. Dennis, thank you for your innovation spirit, energy and focus on making progress in practice. Our regular discussions on organizational topics, strategy and also on the operations research topics stimulate me a lot. Thank you for the initiative and effort spend on winning the Edelman Award, which we would not have won without your contribution.

My third supervisor, Jo van Nunen, guided the start-up of this thesis, but unfortunately passed away and we did not get a chance to really work together, which I regret a lot. Albert Wagelmans, many thanks for stepping in as a supervisor and giving valuable feedback on both the structure and the details in my thesis.

I would like to thank NS, to be specific Wim Fabries, Hanna Brands and Erik Sigger for supporting me in writing this thesis and giving me the opportunity to spend valuable time on writing this thesis. Special thanks go to Wilmet van Dijk, Pieter-Jan Fioole, Joel van 't Wout, and Twan Dollevoet for working together on the development and proper usage of the CREWS system. I am pleased that Wilmet and Pieter-Jan are also willing to stand by my side as my paranymphs.

Also, I am grateful to my other fellow workers: Ramon and Ellis for challenging and reminding me (frequently) that I should complete my thesis, Hilbert for reading and correcting my thesis, and Gabor for helping me with the Latex details. Together with Merith, Thijs, Anneke, Timo, Jitske and Peter, it is fun to work at π!

During the years it was a pleasure to work together with the highly qualified people at Siscog, especially Antonio Frazao, Ricardo Saldahna, Luis Albino, and Jorge Roussado. My visits to Siscog were always very satisfying and focused on the quality of the system and the support it delivers to the planners. At the end of the day we also had many pleasant conversations and activities in which I got to highly appreciate the Portuguese culture.

I would also like to mention my co-authors Mateo Fischetti, Bianca van den Berg, Michiel Vromans, Gerrit Timmer, Anneke Hartog, Niek Wijgaards, David Mobach, and Eddy van der Heijden, with whom it was a pleasure to work.

Very important to me was Tjeu Smeets, as an initiator, sponsor and mentor of the work presented in this thesis. I learned a lot from Tjeu on the organizational forces that influence innovation and on creating an environment that challenges people to think out of the box, in order to improve the daily operations at NS.

My final words go to Tina. Thank you for supporting me in writing this thesis and for organizing our intense live with our three beautiful girls Sara, Eva and Julie. Your choice to quit your former job and to focus on the needs of our family gives us stability and without your support my thesis would not have been written.

Erwin Abbink
Utrecht, 2014

Contents

Chapter 1

Introduction

In 2001 Netherlands Railways (NS) struggled with the relation between management and the train personnel or crew, due to a new way of constructing the daily duties introduced by the management. These internal problems led to several days of a national strike having a big negative impact on the image of NS. NS plays a major role in the mobility in the Netherlands, and the strikes caused problems to the large number of daily passengers who are dependent on transport by train. Therefore, the strikes lead to a public discussion and there was a strong pressure on the management of NS to end the conflict.

The management made a deal with the works council, to find an alternative production model, which ended the strikes. The works council delivered a model describing a set of measurements ensuring variation in the duties, and the management accepted it. Before implementing the new kind of duties, the heavily discussed type of duties had to be operated by the crew members. This was unavoidable because the lead-time in the planning process was too long to change the operational plan within the available time. During this period of operation, the punctuality of the trains was below the agreed level, therefore the Dutch government sent the board away. After that, a new way of duty scheduling was introduced.

Several years later, in April 2008, NS won the INFORMS Franz Edelman Award for applying advanced Operations Research techniques for the development of a completely new timetable (including new crew schedules) from scratch and thereby having a big positive impact on Dutch society. This event received quite some media attention in Dutch newspapers. It even reached national television and radio news.

Less than one year later NS was in the news again, but now because NS' perfor-

mance temporarily dropped off, and the crew members tried to find support for their feeling that the transfer times between different trains to be operated was simply too small and that this was partly causing the operational problems. The management agreed to lengthen the transfer times and the schedules were adapted at short notice. Together with some changes in the timetable, the operational performance stabilized.

These three events show that NS, and more in particular crew planning, has received a lot of attention both within the company and in the Dutch media. NS is a company with a large public responsibility and many people have an opinion about its performance.

In all three events mentioned, decision support systems for crew planning played a major role. With advanced decision support systems, we [1] consider systems that provide all following levels of support to the user:

Levels of system support

1. Registration of schedules

   Support the registration of schedules in a database.

2. Distribution of schedules

   Support the extraction of the schedules from the database and the distribution to other systems.

3. Generation of scheduling elements

   Determine the elements that need to be scheduled, for example driver activities, based on information from other systems. In the NS case for example the crew activities that need to be scheduled based on the given timetable and the lengths of the trains.

4. Graphical presentation

   The basic version is the static presentation of the schedules on paper (printed) or on screen. The more advanced version is an interactive presentation, where the user can graphically change the schedules.

---

[1]In this thesis, "we" means the author, and possibly the co-authors of the papers on which parts of this thesis are based. While performing the research and writing the thesis, the author and several of the co-authors, were part of a R&D department within NS. When we mean NS in general we will use the term "NS" instead of "we". Similarly, we will use the term "R&D Department" when we mean the author(s) and the other colleagues within this department.

5. Validation of schedules

   The basic version of this functionality generates reports that indicate problems in the schedules. For instance, a problem is identified when two activities of the same resource unit overlap in time. On-line validation of schedules is more advanced; here violations of the planning rules are shown to the user directly, while changing the schedules.

6. Evaluation of schedules

   Key performance indicators are computed for the quality aspects of the schedules. This helps the user to evaluate the quality of the schedules.

7. Generation of schedules

   This functionality uses advanced algorithms, for example Operations Research techniques, to generate schedules. This generation is guided by the appropriate evaluation functions of the schedules generated.

In railway planning, automated support for the crew-planning problem has been emerging during the last decade. One of the reasons for this is the increased focus on efficiency due to the introduction of competition in public passenger and freight transport. Railway operations have to compete with automobiles and with airlines achieving a larger market share for middle- and long-distance train travelers. In the airline industry, it was already common to apply OR techniques for the crew-planning problem. Airlines experienced heavy competition and simply needed to improve the efficiency of their schedules with advanced techniques.

Next to competing with other modalities (like bus, car and airline), railway operators started to compete with each other. A stimulus for this was the directive of the European Union that required opening the national railway market in the 90's. These developments require the railway companies to raise their operational performance, and thereby the services to their customers. On the other hand, they must lower costs by operating more efficiently. Train crew is one of the expensive resources and, therefore, railway operators need to schedule the crew as efficiently as possible.

Crew planning is a challenging task. At NS, this was originally a task manually performed by planners, handling conflicting objectives. The first objective is to be efficient; given the tasks to be performed, the goal is to schedule the minimum number of crew. The second objective is robustness; create the schedules in such a

way that they have a maximum capacity for absorbing small delays and can be easily rescheduled when major disruptions occur. The last objective is the quality of work; personnel should feel that the duties have a nice combination of tasks, giving them variation in their work and enough time between tasks.

Applying Operations Research on this multi-objective planning problem at a complex railway operator like NS is not trivial. First of all, subjective criteria like quality of work have to be modeled in mathematical formulas. Second, the available solution methods have to be adapted to the complex instances of NS. Finally, the introduction of an advanced decision support system requires working methods for planners to be changed, different ways of management (e.g., choosing between different scenarios that are evaluated with the system), and also the train crew will experience changes to their daily work due to the impact such tools have on the plan itself. The application of decision support has a major impact on the performance and at the external image of the company. Overall this makes the crew-planning problem a very interesting research subject.

1.1   Crew Management

In this thesis, Crew Management is defined as the management process of the railway's train crew workforce. It is responsible for matching the available number of crew members with the amount of work that needs to be performed, in the strategic planning horizon, for the planning of the crew schedules in the tactical planning horizon, and for the real-time dispatching of the crew members in case of an unforeseen event.

In this thesis we focus on drivers and conductors; other crew members (at ticketing offices, the call center, mechanics, etc.) fall outside the scope of this thesis. In the remainder of this thesis, we will use the general term crew rather than the terms train driver and conductor.

NS currently operates about 4,700 trains on a working day. All these trains need a driver and several conductors (depending on the length of the train). NS employs about 2,700 drivers and 3,000 conductors. In Figure 1.1, we give an overview of the Crew Management process. It shows that the planning process gets three types of input (activities, resources and goals & constraints), and the product is the plan. We describe these elements in the remainder of this section.
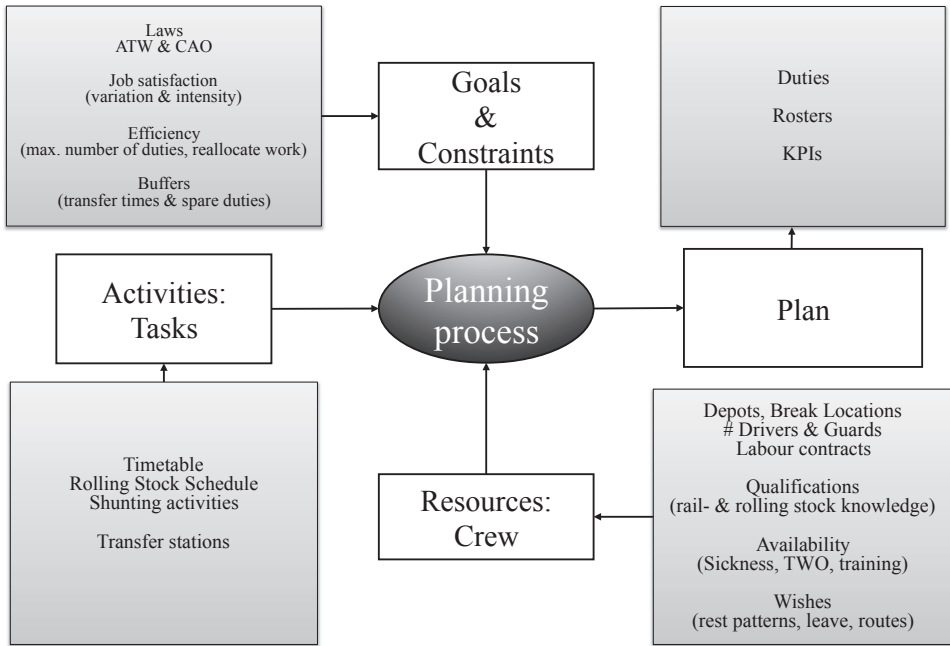
Figure 1.1. The Crew Management Problem

Activities A *task* is defined as a part of a train that has to be assigned to one crew member. Each train (or part of it) in the timetable requires a train driver and a number of conductors. The latter depends on the rolling-stock composition of the train. A crew member can be relieved at all major stations, and, therefore, every timetabled train results in about 3 tasks on average.

Resources Each crew member belongs to a specific crew depot. They perform duties with an average duration of 8 hours, which start and end at the depot. For performing their activities, they possess a certain set of qualifications. For drivers, these include route knowledge and rolling stock knowledge. Having route knowledge means that the driver is familiar with the route on which he has to drive the train. He knows where to expect the signals and where to adjust the speed of the train to specific circumstances in the rail infrastructure. Rolling stock knowledge relates to the specifics of the different types of rolling stock, and the related instructions on how to operate them. These include instructions for the conductors on opening and closing doors, and using the communication system on board.

Goals   NS considers three important goals in crew planning: (1) efficiency, (2) robustness, and (3) quality of work. Efficiency means that the total crew costs are as small as possible. Robustness of the crew duties, i.e., preventing propagation of delays via the crew schedule, depends on several elements, including the transfer times of the crews when transferring from one train to another. The quality of work is the perceived quality of duties by the crew members. This is addressed via labor rules and company agreements, for example, on the amount of variation in the duties.

Constraints   The most obvious constraints that hold for the Crew Management problem are the governmental laws such as the law on work times (in Dutch: Arbeidstijdenwet) and the law on driving times (in Dutch: Rijtijdenwet). These laws prescribe, for example, the maximum duty length, the maximum number of duties with tasks during night hours, rest time between duties and the maximum continuous driving time.

Second, there are collective labor agreements in which the company and its employees define additional constraints. Examples are the maximum number of late duties for crew members that are older than 50 years, duties starting early in the morning should not have a duration of more than 8 hours and the agreement that a meal break is considered being paid working time (included in the duty duration).

Third, there is an agreement on the variation, called "Sharing–Sweet–and–Sour." As the name Sharing-"Sweet-and-Sour" reveals, this set of rules aims at a fair allocation of the "sweet" and "sour" workloads among the 29 crew depots. "Sweet" represents the variety in routes and lines as well as work on intercity trains. "Sour" mainly represents the work on lines with a lot of anticipated passenger aggression and work on relatively old rolling stock. They aim at allocating the popular and the unpopular work as fairly as possible among the different crew depots. For instance, some routes are more popular than others, and intercity trains are preferred over regional trains.

Another example is the percentage of work on intercity trains. Of the work assigned to a depot for a week, at least 25% should be on the intercity trains. One can require every weekday to contain at least 25% of this work, but it is better to check this constraint for a complete week. For a more detailed description of these rules, we refer to Section 2.4.3.

Finally, there are constraints that are agreed with the management each year. Examples are the minimum transfer time between tasks that are not on the same rolling stock unit, stability in routes per depot, and the number of conductors per

train. These constraints vary with the focus of the management. If cost saving is the focus, transfer times can be reduced, and the number of crew is reduced. If the robustness is the focus, transfer times are increased.

Crew Management process   The Crew Management process starts with strategic or capacity planning. This phase deals with the long term availability and number of crew members, per depot. After capacity planning, the Crew Management process addresses the problem of assigning work to individual crew members, a complex task for each public transport company. Traditionally, this part of the Crew Management process is divided into three steps; crew scheduling, crew rostering and crew dispatching. NS is no exception in this and follows these process steps. For a more detailed description of these steps, we refer to Chapter 2.

The output of the planning process is the plan, which contains the assignment of tasks to the individual crew members. More specific, duties are produced which are assigned to the individual crew member via the roster.

When the crew duties and rosters are scheduled, they are operated. In operations, one has to cope with disturbances like delays, accidents, infrastructure breakdown and rolling stock defects. Handling these disturbances is called real-time dispatching. One or more duties can become infeasible to operate, due to such disturbances. Solutions have to be found quickly and need to be communicated with the crew. Crew dispatching is considered being a major problem in real time dispatching for passenger rail transport.

1.2   Motivation

This thesis gives an extensive overview on the Crew Management process, including crew scheduling, rostering and crew dispatching in the complex setting of NS. We will show that the problem at NS is not only complex from a combinatorial point of view, but also from an organizational perspective. Describing all aspects in detail will show that each mentioned problem is interesting by itself. Crew dispatching is, for example, considered being the bottleneck in dispatching. The current manual process limits the reaction time of NS to external events. Improving this process will lead to better service. Next to describing the individual sub-problems, we will also reveal the interaction between the sub-problems. In many other research studies, these aspects are not properly addressed or even neglected.

From an Operations Research (**OR**) perspective, this thesis describes several so-

lution approaches for solving parts of the Crew Management process. Column generation, Lagrangian relaxation, integer linear programs, set covering models, advanced local search and meta-heuristics are used to solve the problems. This wide range of techniques is necessary to get good solutions for the overall problem. This also indicates that the problem is very complicated.

The main focus of this thesis is not on the techniques themselves, but on the application of the techniques and the impact on the company. Both the impact on the Crew Management process and the impact on the quality of the produced plans are given. In general, the applied techniques have already been studied before. In most cases, the focus is on the techniques themselves and not on the application. In our opinion, this aspect is essential for achieving real support and business impact, and therefore, worth studying.

From a business perspective, we describe the process of implementing an innovative way of Crew Management from exploration to exploitation. Innovation is a hot topic in research and many papers are written about it. Innovation in the railway industry, which is known to be very conservative, is a relatively new research area. As NS is one of the early adopters of OR techniques, other companies can use the lessons learned.

The models and solution methods are applied to the Crew Management problem of NS, and the analysis is based on this. As a consequence, one can ask whether the results are applicable for other operators as well. Many of the results and findings can be generalized, so other operators, or even other (public transport) companies, can learn from them.

1.3   Research questions

This thesis deals with Crew Management for passenger railway operators, in particular at NS. As stated in the previous section, we address crew scheduling, crew rostering and crew dispatching. The main research question of this thesis is:

<blockquote>
"How to improve the Crew Management Process
in Passenger Rail Transport?"
</blockquote>

In order to answer this question, it is essential to understand Crew Management and its important aspects, as well as the supporting planning process in practice. In addition, we investigate the important criteria for assessing the quality of a crew plan.

We also review the available OR methods for supporting the Crew Management process. Based on the understanding of the Crew Management process and the available solution techniques, we consider which techniques are appropriate for supporting the Crew Management process and how these techniques are best used in practice. Finally, we show the effects of applying decision support on the management process, the management decisions, the quality of the plan, and the satisfaction of the crew. The main research question breaks down into the following sub-questions:

1. What are the important aspects of Crew Management?

2. How can we quantify the quality of a plan?

3. What are commonly used methods for supporting the Crew Management process?

4. How to apply these methods to the Crew Management process?

5. What is the impact of decision support on the quality of the plan?

6. What is the effect of decision support on the Crew Management process?

7. How to implement decision support in an organizational context?

1.4 Outline of this thesis

First we will give an outline of this thesis and, after that, we will give a guideline for reading this thesis.

Chapter 2 describes the railway management process at NS in detail.

Chapter 3 deals with Crew Scheduling. First, we address the initial application of a commercially available system, based on column generation and set covering, for solving a complicated labor conflict at NS. Second, we describe the aspects of implementing the system as a regular system used for supporting the Crew Scheduling process at NS. We describe how we implemented the company specific constraints and objectives, and how we improved efficiency using a heuristic. Finally, we describe the development of the algorithm called LUCIA together with a commercial software vendor.

In Chapter 4, we present two Mixed Integer Programming models for solving the Crew Rostering problem. We have tested the models on real-life instances and show that the models can be very useful.

Chapter 5 describes a possible approach for supporting the Crew Dispatching process. We present an actor-agent system that was developed for solving disturbances, in which case the driver's schedules become infeasible and have to be repaired. We compare the approach with an alternative approach presented in Potthoff et al. (2010).

In Chapter 6, we elaborate on an example of a change process NS went through, for implementing decision support systems. We describe the results of applying decision support systems at NS and relate lessons learned to literature found on the successful application of Operations Research in practice. We end this chapter by discussing current research topics.

In Chapter 7, we draw some final conclusions.

This thesis is written for OR practitioners, for OR scientists and for business oriented readers. We suggest that OR practitioners read the complete thesis. Business oriented readers are suggested to read Chapter 2, 6 and 7, and possibly Appendixes B and C. Operations Research scientists are suggested to read Chapters 3 to 5 in detail, which focus on modeling and solving the Crew Management problems.

1.5   Related Papers

This thesis is based on several reports and papers. Chapter 2 and Appendix C are partly based on the papers:

Kroon et al. (2009): KROON, HUISMAN, ABBINK, FIOOLE, FISCHETTI, MARÓTI, SCHRIJVER, STEENBEEK, AND YBEMA. *The New Dutch Timetable: The OR Revolution*. Interfaces.

Abbink et al. (2004): ABBINK, VAN DEN BERG, KROON, AND SALOMON. *Allocation of Railway Rolling Stock for Passenger Trains*. Transportation Science.

Chapter 3 and Appendix B are based on the papers:

Abbink et al. (2005): ABBINK, FISCHETTI, KROON, TIMMER, AND VROMANS. *Reinventing Crew Scheduling at Netherlands Railways*. Interfaces.

Abbink et al. (2008b): ABBINK, VAN 'T WOUT, AND HUISMAN. *Solving Large Scale Crew Scheduling Problems by using Iterative Partitioning*. Technical Report Report EI2008-03, Econometric Institute, Erasmus University Rotterdam.

Abbink et al. (2011): Abbink, Albino, Dollevoet, Huisman, Roussado, and Saldanha. *Solving large scale crew scheduling problems in practice*. Public Transport.

Chapter 4 is based on the paper:

Hartog et al. (2009): Hartog, Huisman, Abbink, and Kroon. *Decision support for crew rostering at NS*. Public Transport.

Chapter 5 is based on the papers:

Abbink et al. (2008a): Abbink, Mobach, Fioole, Kroon, Wijn–gaards, and van der Heijden. *Actor-agent based approach to train driver rescheduling*. In *Proceedings of the 20th Belgian-Dutch Conference on Artificial Intelligence (BNAIC)*.

Abbink et al. (2009): Abbink, Mobach, Fioole, Kroon, van der Heijden, and Wijngaards. *Actor-agent application for train driver rescheduling*. In *Proceedings of the Eighth International Conference on Autonomous Agents and Multiagent Systems*.

Mobach et al. (2009): Mobach, Abbink, Fioole, Kroon, van der Heijden, and Wijngaards. *Train driver rescheduling using task-exchange teams*. In *Proceedings of the Second International Workshop on: Optimisation in Multi-Agent Systems (OPTMAS) at AAMAS*.

Abbink et al. (2010): Abbink, Mobach, Fioole, Kroon, van der Heijden, and Wijngaards. *Real-time train driver rescheduling by actor-agent techniques*. Public Transport.

Chapter 2

Rail Transport Planning in the Netherlands

This chapter deals with railway planning in the Netherlands, with the focus on passenger transport. We start with a description of the organization of rail transport in the Netherlands. Then we continue with an overview of the railway planning problems, including the Crew Management problem.

2.1   Rail Transport Organization in the Netherlands

Netherlands Railways (Nederlandse Spoorwegen, or NS) was founded in 1837. In this year the Hollandsche IJzeren Spoorweg Maatschappij (HIJSM), started the construction of the first railway line in the Netherlands, between Amsterdam and Haarlem, which lead to the first train service in 1839. The HIJSM and the "Maatschappij tot Exploitatie van Staatsspoorwegen" (founded in 1860) merged into NS in 1937. Originally a private enterprise, NS became a non-profit, state-owned company in 1938, operating passenger and freight services and building and maintaining the railway infrastructure.

Because of European Union regulations and liberalization of the railway market, NS was split into several companies during the period 1995–2002. The state maintains ownership of the infrastructure because of its strategic value. ProRail, a nonprofit organization owned by the state, is responsible for maintaining and allocating the infrastructure. Other railway operators entered the market, and NS became a commercial operator, which still carries the name Netherlands Railways. NS decided

to focus solely on passenger transport and sold its freight operations to the German Railways (Deutsche Bahn).
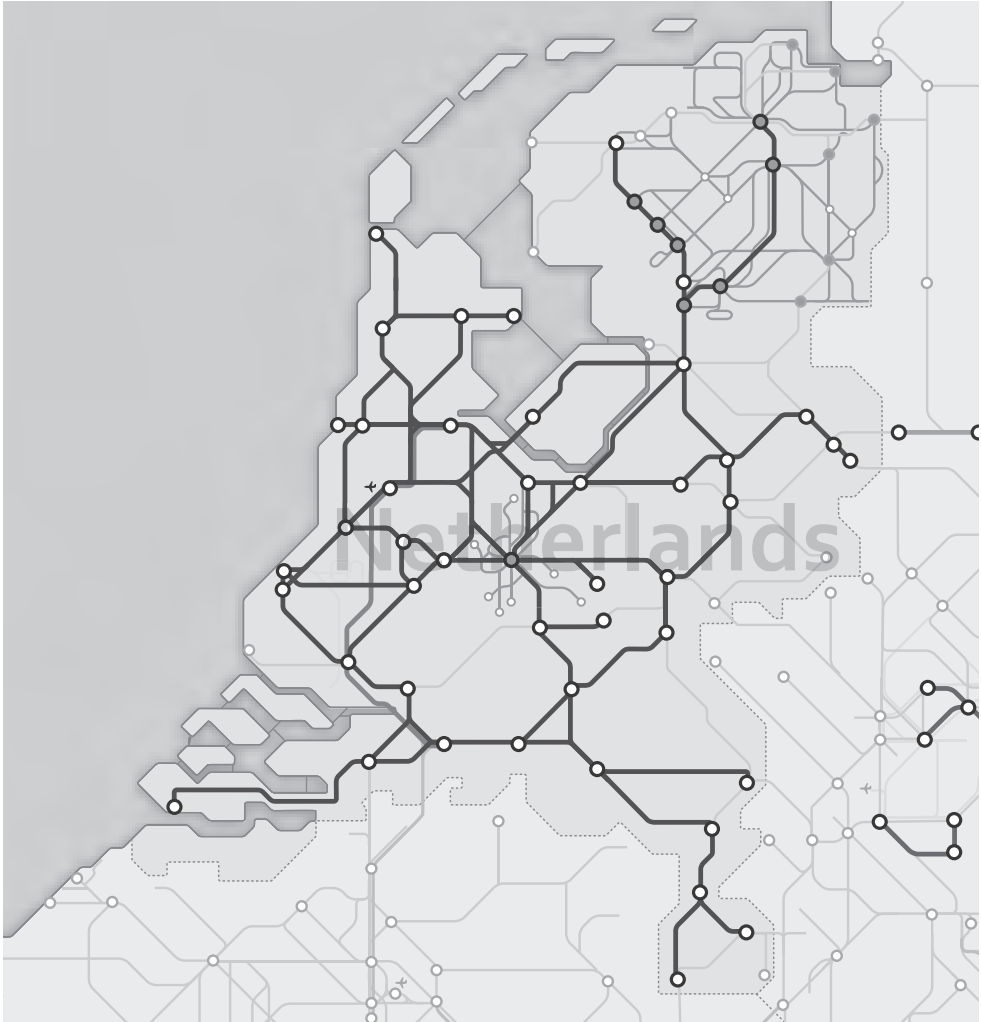
Figure 2.1. The Netherlands railway network (2013).

NS Reizigers (NSR) is a full daughter company of NS that is responsible for operating trains in the Netherlands. Currently, NSR operates passenger trains on all main lines. Those are the solid black lines in Figure 2.1. Five other operators run passenger trains on several regional lines. Those are the light gray lines in the figure. In addition, there are more than twenty freight operators. The dark gray lines in the north and the center of the Netherlands are the bus lines operated by the NS

daughter company Qbuzz.

NSR, which owns the license to operate passenger trains on all main lines, is by far the largest passenger train operator in the Netherlands. NSR[1] operates about 4,700 of the total number of 5,500 trains that run on the Dutch railway network on an average working day. NSR carries well over one million passengers per working day on the Dutch network and employs about 12,000 people (about 32,000 within NS in total). In the Netherlands, the total network, which is mostly double-tracked and electrified, is approximately 2,800 kilometers long. The four large cities, Amsterdam (Asd), Rotterdam (Rtd), The Hague (Gvc), and Utrecht (Ut) are all in the western part of the country. The main lines represent approximately 90 percent of the total passenger demand; the NSR operating revenues on these lines are, approximately, 2 billion Euros per year.

In 2013, about nine million different passengers traveled about 16.1 billion passenger kilometers by train. On average, each Dutch citizen travels approximately 1,000 kilometers by train per year. Figure 2.2 shows that the Dutch railway system
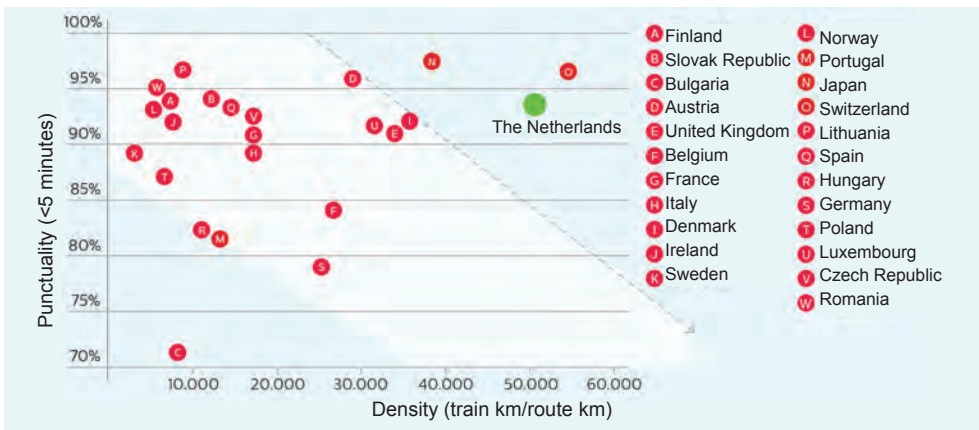
Figure 2.2. Punctuality versus density, NS Webpage (2013)

is one of the most intensively used networks within Europe and performs very well on punctuality.

Figure 2.3 shows that NS transports the most passengers per kilometer of the railway line in Europe, illustrating its efficient use of railway infrastructure.

Next to operating the passenger trains in the Netherlands, NS has broadened its view and has the ambition to become a multimodal European public transport and service company. Currently, NS also operates trains and stations in the United

---

[1]These figures relate to the year 2013
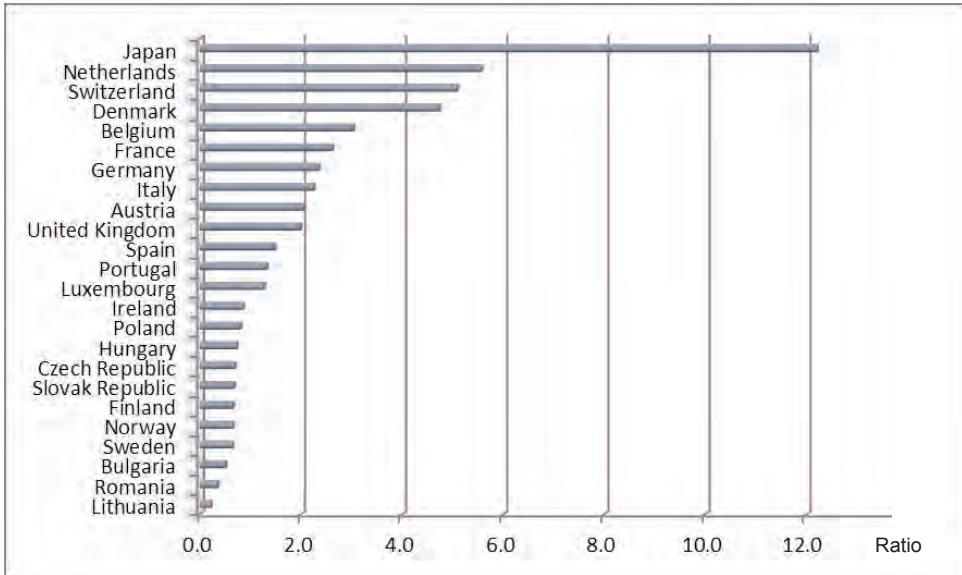
Figure 2.3.   Ratio of million passenger kilometers over line kilometers (World Bank 2011).

Kingdom. NS develops new commercial activities in and around stations, which has helped NS to become one of the largest Dutch retailers. In 2013, revenues of NS exceeded 4,6 billion Euros; about half of this is related to passenger transport in the Netherlands. In the remainder of this thesis, we will use the term NS for both the mother company as its daughter company NSR.

Passenger railway services are especially attractive for traveling between the large cities. For example, the NS market share of passenger movements between the four largest cities (Amsterdam, Rotterdam, The Hague, and Utrecht) is above 50 percent during peak hours, a period in which the road network between these cities is highly congested. If all commuters who currently travel by train switched to travel by car, these cities would become almost inaccessible. This would have a dramatic negative impact on the Dutch economy. Railway transport is environmentally friendly. The CO_2 emission per train-passenger kilometer is about one-third of the emission of an average car passenger kilometer; see NS Webpage (2013).

For these reasons, the Dutch government stimulates the growth of railway transport. However, it does not want to invest additional billions of Euros in new in-

frastructure, although its infrastructure is already very highly utilized compared with other countries. Building infrastructure is very expensive, time-consuming, and requires considerable amounts of land, which is unavailable in most places in the Netherlands. The solution for enabling growth of railway transport lies in developing better schedules for the operation of trains on the current infrastructure.

2.2 Railway Planning Problems

In this section, we give an overview of the railway planning problems at NS. There are several ways to categorize the railway planning problems. First, we discuss the commonly used framework proposed by Anthony (1965). He classifies planning (decisions) in three categories: strategic planning, tactical planning, and operational planning. We give a short introduction on the characteristics of each of these categories.

Anthony defines *strategic planning* as the process of deciding on the objectives of an organization, on the resources needed to attain these objectives and on the policies that are to govern the acquisition, use and disposition of these resources. An essential characteristic of strategic planning is that it is long term planning, and that, in general, strategic decisions have an impact over a long term horizon. Examples of strategic decisions are major capital investments in new production capacity and expansions of existing capacity, merger and divestiture decisions, determination of the location and size of new physical facilities, development and introduction of new products, and issuing of bonds and stocks to secure financial resources.

Tactical planning is defined as the process by which managers assure that resources are obtained and used effectively and efficiently in the accomplishment of the objectives of the organization. Tactical planning decisions have an impact on the medium term. Once the physical facilities have been decided upon, the basic problem to be resolved is the effective allocation of resources (e.g., production, storage, and distribution capacities and work-force availabilities) to satisfy demand, taking into account the costs and revenues associated with the operation of the resources available to the organization.

Finally, *operational planning* is defined as the process of assuring that tasks are carried out, effectively and efficiently. Operational planning is planning for the short term, as operational plans, generally, have a time horizon of several hours to several days. After making an aggregate allocation of the resources, it is necessary to deal with the day-to-day operational and scheduling decisions. Typical planning decisions

at this level are the assignment of customer orders to individual machines, the sequencing of these orders in the workshop, inventory accounting and inventory control activities, dispatching, expediting and processing of orders, vehicular scheduling, and credit granting to individual customers.

These three categories of planning differ in various dimensions and can be characterized (expressed in relative terms) as in Table 2.1.

Characteristics	Strategic planning	Tactical planning	Operational Planning
Objective	Resource acquisition	Resource utilization	Execution
Time horizon	Long	Middle	Short
Level of management involvement	Top	Medium	Low
Scope	Broad	Medium	Narrow
Source of information	External	External & Internal	Internal
Level of detail	Highly aggregate	Moderately aggregate	Low
Degree of uncertainty	High	Moderate	Low
Degree of risk	High	Moderate	Low

Table 2.1. Distinct Characteristics of Strategic, Tactical, and Operational Decisions (Anthony (1965))

We extend the categories with a fourth one: *Operations control*. Operations control is the process of supervising the execution of the plan and reacting to disturbances, in real-time. One could argue that this is the last phase of operational planning, but we think that the problem is in nature different from the operational planning and, therefore, we would like to create this fourth category.

Using the four categories we give a schematic overview of the planning phases of the rail transport planning processes in Figure 2.4.

At NS, strategic planning has a planning horizon of several years or even decades. In this phase, the offered service is determined in the form of the Line System. It is possible to build new rail-infrastructure, purchase new rolling stock and hire new crew.

The tactical planning phase has a horizon of 2 months up to a year. In this phase, the details of the plan are determined, but major changes in availability of resources are not possible. The offered service is determined in the form of the Basic Hour pattern and the assignment of rolling stock types to different lines. Also, a detailed
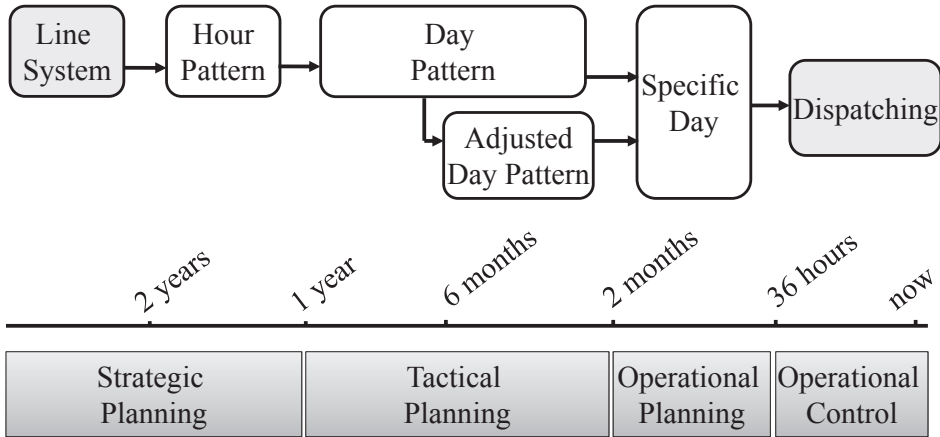
Figure 2.4. Planning phases at NS

plan is created for generic days, dealing with a generic Monday, Tuesday and so on.

The operational planning phase has a time horizon of 36 hours up to 2 months. Based on the plans for the generic days, for each calendar day an operational plan is made taking into account planned exceptions to the generic plan, e.g., planned maintenance work to the rail-infrastructure.

Finally, the operations control phase has a horizon of at most 36 hours, including real-time operations. In operations control, short term and unexpected changes are handled. For example, broken infrastructure, broken rolling stock, or sickness of an individual crew member can require modifications to the plan.

Next to the four planning phases, we decompose the railway planning problems by the resource involved. The most important resources in the setting of railway planning are rail-infrastructure (timetabling), rolling stock and crew. In general, these resources are not planned in a single step, due to the complexity of the overall railway planning problem. A common approach is to split the problem by resource and to solve the sub-problems sequentially as shown in Figure 2.5. This figure is representative for all planning phases. The solid lines indicate the sequential order in which, in principle, the planning steps are performed. The dotted lines indicate
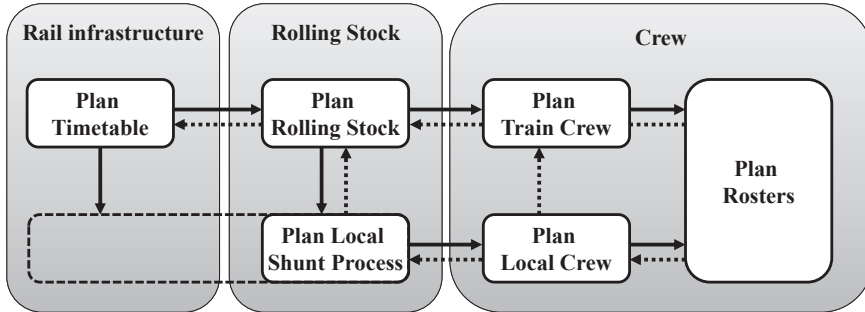
Figure 2.5. Planning steps at NS

that sometimes iterations are needed because a planning step requires a change to be made in a previous step.

Planning the usage of rolling stock is split into two steps: (1) planning the usage of the rolling stock units for train services, and (2) planning the local shunting processes. The extended dashed box around the "Plan Local Shunt Process" indicates that, although we consider this step to be part of planning the usage of rolling stock, there is a strong interaction with infrastructure usage. Each decision on where to store a rolling stock unit (locally in a shunt area), requires a train movement for shunting the unit to the right position.

For the usage of the crew, we identify three main steps: (1) assigning the tasks related to operating the train services to crew duties, (2) assigning the tasks related to shunting to crew duties, and (3) assigning the crew duties to individual crew members by creating crew rosters.

The identified problems are, usually, solved sequentially. To avoid going back and forth between the planning problems, each problem takes into account characteristics of the subsequent problem. For example, the required number of conductors on a train depends on the length of the train, which is determined during the rolling-stock scheduling. Therefore, minimizing the required number of conductors is part of the multi-criteria objective function in the rolling-stock scheduling problem.

In Figure 2.6, we use the planning phases (y-axis) and the three main resources (x-axis) to categorize the set of rail planning problems. In the remainder of this section, for each planning phase, we describe the planning problems related to each of the identified planning steps.
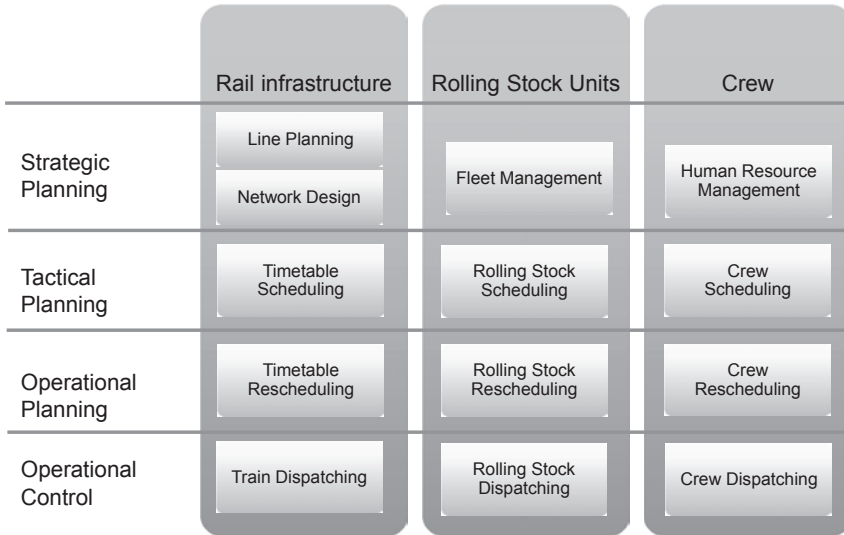
Figure 2.6. Rail Planning Problems

2.3   Strategic Planning

In strategic planning, the offered service is defined in the form of the Line System. This Line System is closely related to decisions on investing in new rail-infrastructure (network design), fleet management and human resource management. These issues are discussed in the following sections.

2.3.1   Line System Planning

The line system, which is the collection of all train lines, is the key input for the timetabling process, in which each train line has an origin station and a destination station. A frequency and a stopping pattern show the stations at which the trains on the line call: at all stations (regional trains), or at major stations only (intercity trains).

For example in Figure 2.7, there is a line from The Hague to Utrecht that runs once per hour and stops only at major stations. Afterwards, this line continues to Enschede. This type of line is called an intercity train line. Another line runs from

The Hague to Utrecht twice an hour (indicated with the thick line) and stops at every intermediate station. This second type of line is called a regional train line.
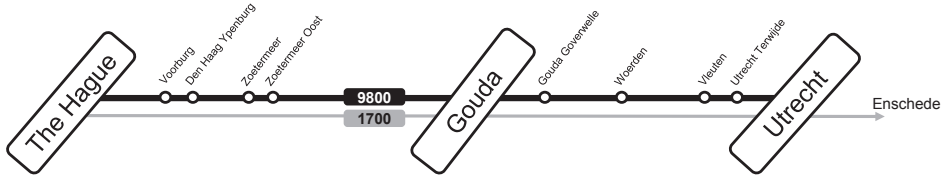
Figure 2.7. Line System The Hague – Enschede

A line system consists of all the individual train lines. Figure 2.8 shows the line system of 2013 in the Western part of the country. The dotted lines indicate train lines that are only operated during peak hours.

2.3.2   Network Design

Network design explores possible expansions of the rail-infra, e.g., where to build new railway lines and switches, where to locate new stations for passengers and where to locate rolling stock maintenance depots.

In the Netherlands, this is the responsibility of ProRail. However, it closely interacts with Line System planning described in the previous section. Changed line systems might require another rail infrastructure. For example, doubling the frequencies of services between two stations might be dependent on doubling the tracks between those stations, or might require additional platforms. Opening a completely new station is an example where the network is changed, and the line system is changed as well, this to allow the train services to dwell in the new station.

2.3.3   Fleet Management

NS has several rolling stock types available for passenger transport. The two main classes to be distinguished are power-driven equipment and locomotive-hauled carriages. Power-driven equipment consist of train units that can move individually without a separate locomotive. Each rolling stock unit consists of a number of carriages that cannot be split from each other during the operations. Within each class, various rolling stock types can be distinguished. Power-driven equipment includes,
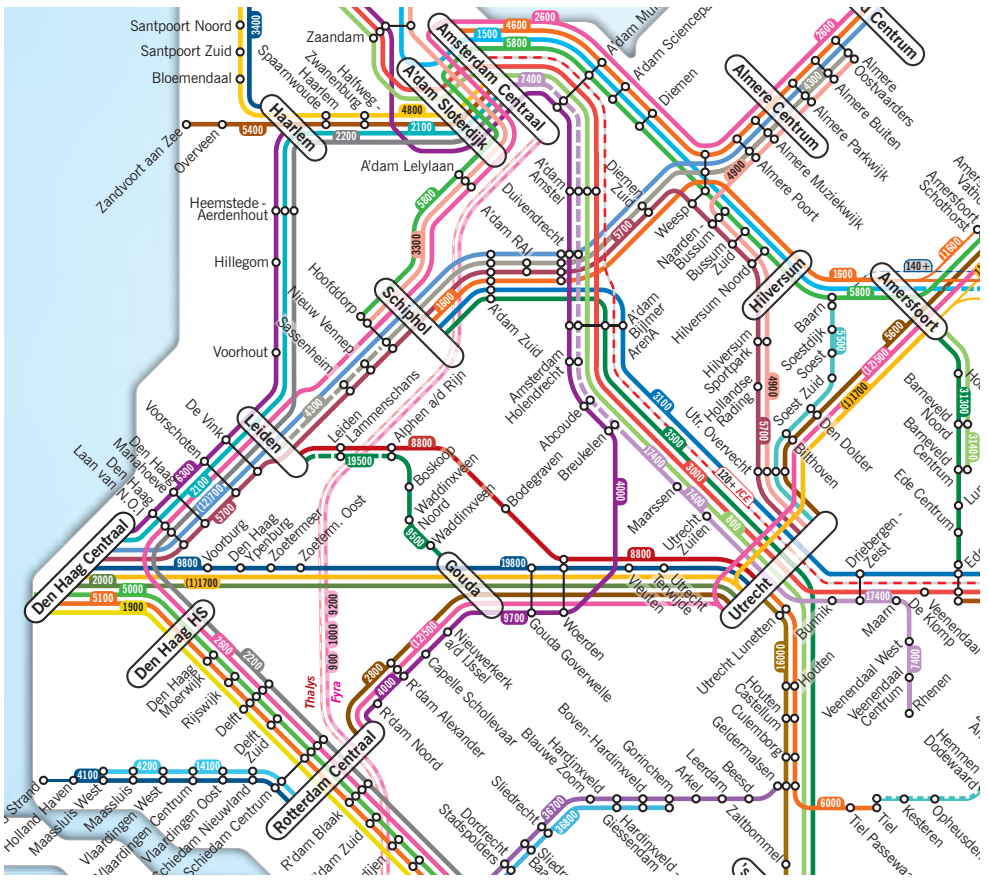
Figure 2.8. 2013 Line System western part the Netherlands

among others, *Koplopers* for the intercity trains and *Sprinter Light Trains (SLTs)* for the regional trains.

The train types can be subdivided subsequently into different subtypes. Koplopers, for example, can be subdivided into a subtype with three carriages and one with four carriages. Figure 2.9 shows an example. Similarly, SLTs can be subdivided into units with four carriages and units with six carriages. Train units of the same type can be combined to form longer trains with more seating capacity. Train units of different types generally cannot be combined in one train. The different subtypes within a type allow for more flexibility in the seating capacities of the trains. For example, with three- or four-carriage single-deck train units, all train compositions with a length of more than three carriages can be formed, except for compositions with five carriages. The maximum of a composition is, typically, 12 or 15 carriages.

Figure 2.9. This figure illustrates two single-deck rolling-stock units with three and four carriages, respectively

Fleet Management deals with strategic issues related to the long term availability of rolling stock. The horizon of this planning process is very long; it takes several years to purchase new rolling stock, and the lifecycle of rolling stock typically covers several decades. Based on long term demand forecasting models and scenario analysis, decisions are made with the aim to align the available capacity of rolling stock with the stochastic nature of the demand for passenger railway transportation.

Forecasts need to be made for both peak demand and off-peak demand, and for the ratio between first and second class passengers. The available capacity is not only based on the availability of rolling stock but also on the service level to be provided (e.g., what is the supplied seating probability). Next to that, the availability is directly related to the number of units that are in maintenance, on average, both in peak and off-peak periods. Therefore, the maintenance strategy is a key issue in rolling stock management.

In rolling stock management, decisions have to be made about which type of rolling stock to use, on acquisition and leasing of rolling stock units, on refurbishment and life-time extension of existing units, on selling of excess units, and on destruction of end-of-lifecycle units. When acquiring new units, decisions have to be made about the first and the second class capacity per unit, the length of the units and the technical characteristics like acceleration speed required for the timetable to be operated. In general, large units are rather inflexible to operate, and smaller units have a higher price per seat.

It also needs to be decided when to execute a decision. Additional units should not be available before they are required to be operated because this is a waste of money. Vice versa, if they are not available on time, the required capacity cannot be provided. The overall goal is to minimize the expected life cycle costs of the rolling stock, taking into account the required service to the passengers.

2.3.4 Human Resource Management

Human Resource Management deals with matching the required and available capacities of the crew depots on the long term. The available capacity of a depot can be changed by hiring additional crew, training crew so that they can perform additional activities, or (temporary) moving people to other depots, in general via secondment. Firing people is not common practice and reducing available crew is achieved by natural reduction of the number of crew due to retirement and crew that finds another job, inside or outside the company. Also, one can decide about the location of the depots. It might be useful to open or close depots at some point in time.

The required capacity per depot depends on both the timetable and rolling stock assignment, and on the labor agreements between management and crew. This capacity needs to be determined with long term forecasting models. Because trains are operated between multiple stations and the assignment of tasks to the depots is not fixed, one can influence the required capacity by the assignment of tasks to the depots where there is sufficient availability of crew.

It takes one (conductors) to three (drivers) years to fully train new crew members. This includes both theoretical training and practical (on-the-job) training. Together with the social aspects of hiring people, this implies that Crew Management has a long lasting effect on the capacities of the involved depots.

Quality aspects of Human Resource Management   We discuss the quality attributes of human resource management plans based on the 3 aspects introduced in Section 1.1: (1) efficiency, (2) robustness, and (3) quality of work.

The (expected) excess capacity, the difference between the provided and required capacity, is an important indicator of the quality of the plan. A plan is efficient when this excess capacity is low. This way unnecessary capacity costs are prevented. On the other hand, the robustness of a plan is high with a large excess capacity. The excess capacity can be useful when the required capacity turns out to be higher than expected. Of course, both the future provided capacity and the future required capacity are uncertain. The better the forecasts are, the better the quality of the plan will be. Only at the moment of operations, one can determine if the given capacity is sufficient and not too large.

The quality of work is the perceived quality of human resource management by the crew members. From the perspective of the crew members, it is also important that the available number of crew members is aligned with the required capacity. An excess of capacity means that the jobs of the employees might be endangered, and

with insufficient capacity, it might be required that the crew members postpone their holiday leaves, until the shortage is repaired.

2.4   Tactical Planning

In the tactical planning phase, the details of the plan are determined. Major changes in availability of resources are not possible. The Basic Hour pattern is created and rolling stock types are assigned to the lines. Also, a detailed plan is created for generic pattern days, including the timetable, the rolling stock schedules, the shunt plans and the crew schedules. Once a year, a new generic annual plan is created. This generic annual plan is modified about 6 times a year as a result of changes in the passenger forecasts and the availability of rail infrastructure and rolling stock.

2.4.1   Timetable Scheduling

The timetable describes the planned departure and arrival times of every train at every relevant station, including the routing over the network. Passengers can use it to plan their journeys, and it is the input for creating the plans for the rolling stock and crew. The timetable is scheduled in a few steps: The creation of the Basic Hour Pattern, routing trains through the stations, and 7x24 timetable planning. These steps are explained in the next paragraphs.

Basic Hour Pattern Scheduling

As stated, the timetable describes the planned departure and arrival times of every train at every relevant station. These time instants are called events. Other relevant events are the time instants at which the trains pass junctions, bridges, and other locations where coordination of train movements is required. Bridges, which need to be opened frequently for ships, are a particular bottleneck in the Dutch railway system because of the numerous waterways in the Netherlands.

NS uses a cyclic timetable. The basic idea is that the same timetable is repeated several times in succession. NS uses a cycle time of 1 hour. This means that trains leave every hour from the same platform at the same minute. For example, the intercity service from The Hague to Enschede leaves The Hague every hour at minute 24 from platform 4, at 8:24, 9:24, etc. Similarly, it arrives in Enschede about two and a half hours later.

This regular cycle of the timetable is highly appreciated by the passengers because it is easy to remember the departure and arrival times. Moreover, Dutch passengers have long been used to it, since NS was the first company to introduce such timetables, as far back as the 1930s. During the last few decades, many other European countries also introduced cyclic timetables. This makes it much easier to synchronize timetables between countries.

One possible disadvantage of a cyclic timetable is that, in particular periods of the day, there may be many trains serving a relatively low demand. NS prevents this by allowing some exceptions in the cyclic timetable. For example, between The Hague and Utrecht, NS does not operate four, but rather two intercity trains per hour in the late evenings.

An example of a one-hour timetable for the route Gouda–Utrecht in one direction is plotted in Figure 2.10. The horizontal direction represents the time scale, from
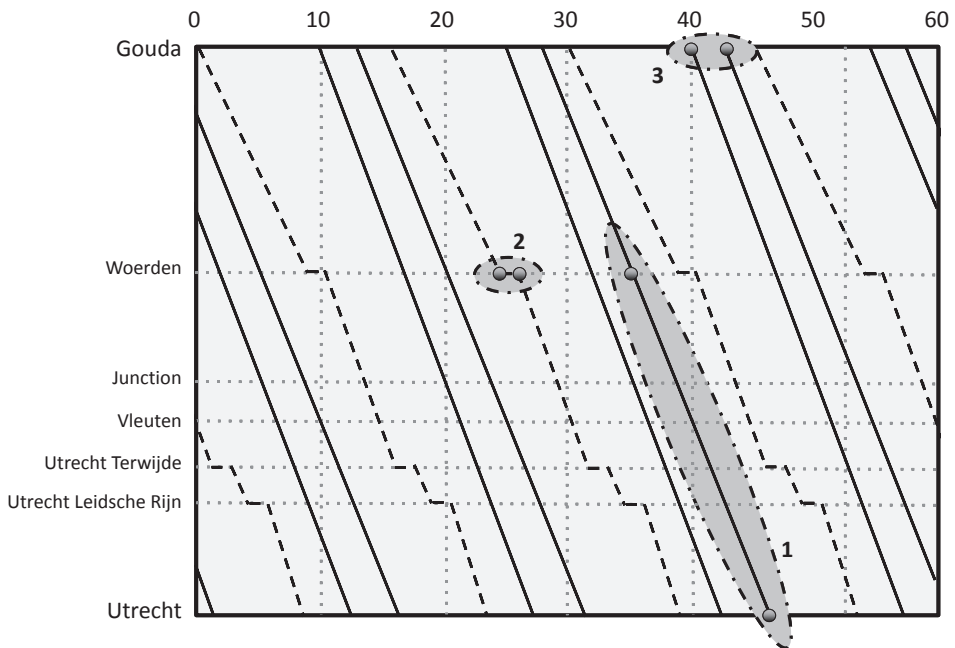
Figure 2.10. Basic Hour Pattern

0 to 60 minutes, and the vertical direction represents the place. On a large part of this route there are 2 tracks, one going in each direction, on other parts (e.g.,

between Woerden and Utrecht Leidsche Rijn) there are 2 tracks per direction, such that fast trains can overtake slow ones. On this heavily used route, there are 8 intercity services from Rotterdam and The Hague to Utrecht and beyond.

In the figure, for example, the leftmost train (with a solid line) leaves Gouda at minute 10 and arrives in Utrecht at minute 26. Indicated with dashed lines, 4 local trains are plotted. Notice that the slow trains leave Gouda just after an intercity train and arrive in Utrecht just before the next one. Finally, note that all trains are at the same location at minute 60 as at minute 0 due to the repetition of the timetable in the next hour.

While planning the event times, for all railway processes certain process times have to be respected, indicated with the ovals in the figure, where the time difference between two points are the planned process times, e.g., running time (1), dwell time (2), headway time (3). These must be selected in such a way that the resulting timetable is cyclic. The timetable for the whole country consists of individual timetables for every different route, which are connected to each other.

To increase the robustness of the timetable, one can increase the planned running times, dwell times, and headway times by time supplements. Time supplements in the running and dwell times absorb small disturbances in the real-time operations, allowing trains to recover from delays. Time supplements in the headway times, also called buffer times, reduce the propagation of delays from one train to another. Thus, time supplements and buffer times add to the predictability of the railway system. However, the downside is that extending time supplements implies an increased planned travel time for passengers which makes it less attractive for the customer. Also, it might require more rolling stock and crew, and thus influence the costs of operating the timetable.

Routing Trains through Stations

The timetable is not complete until all trains are assigned to a platform at every relevant station. Moreover, for each train, it must be decided how to route it through the stations. To give an idea of the complexity of this problem, Figure 2.11 shows the tracks of Utrecht Central Station. There are many platforms and routes to choose from. In the picture, it is shown how the intercity train from The Hague arrives at the station. Around the same time, there are also many other trains arriving from and departing to other directions. Trains running from one station to another or trains dwelling inside a station define the relationship between these events. Similarly, the headway time between two consecutive trains on the same route also defines a
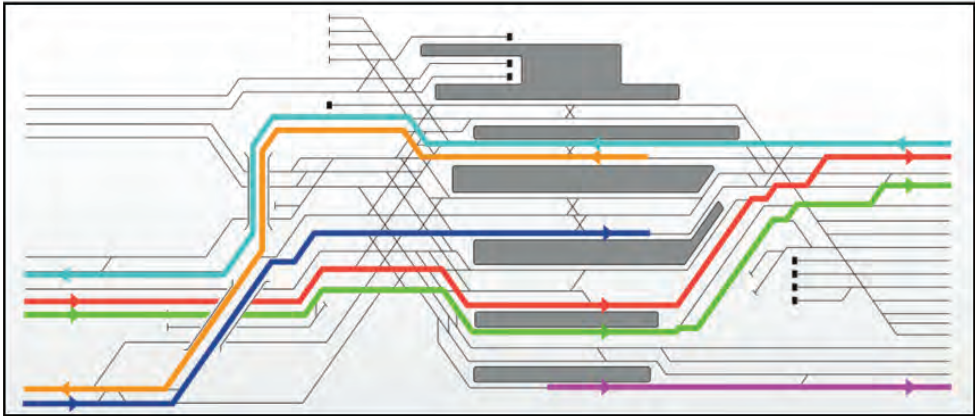
Figure 2.11. Example of some routes through a station

relationship between two events.

Stations form a bottleneck in the Dutch railway system because many trains from different directions come together at the stations, and the stations have limited infrastructure. Moreover, to provide good transfer opportunities for the passengers, trains preferably arrive (and depart) more or less at the same time and at adjacent platforms. Therefore, trains can easily hinder each other inside the stations, and finding appropriate routes for the trains through the stations is as important as determining their arrival and departure times. As long as these routes are undetermined, the timetable is incomplete.

7x24 Timetable Planning

In this timetabling problem, it is decided which trains, from the Basic Hour pattern, will be operated for each hour of the day, for all days of the generic week. During off-peak hours, some of the trains do not run, and during the night most of the trains are not operated. Therefore, the start-up of a day and the end of a day have to be decided. For each starting train, it has to be decided at which station it has its first departure, and at which station it will have its final arrival. Input for this planning is the cyclic hour pattern and the passenger demand, over the days and the week.

Next to fluctuations in demand, also maintenance of the infrastructure needs to be considered. The infrastructure manager has reserved several time slots in which for each pattern day, some parts of the network are available for maintenance work. This implies that trains have to be rerouted or must be (partly) skipped. Mostly these maintenance slots are allocated at the off-peak hours, so the impact on the

passengers is limited. Output of this step is a timetable that can be operated, and that serves as the input for the timetable of the calendar days.

2.4.2   Rolling-Stock Scheduling

The rolling stock scheduling is also performed in a few steps. First, an assignment is made of the number and types of rolling stock units to the separate lines, then the units are assigned to the planned timetable, and finally a shunt plan is created. These steps are described in the next paragraphs.

Rolling-Stock Assignment to Train Series

This assignment problem deals with finding the most effective allocation of the train types, subtypes, and units of rolling stock to a train series. A *train series* is defined as a single line or a subset of lines from the line system to which a fixed amount of rolling stock is assigned. This assignment takes into account that as many people as possible can be transported with a seat, especially during the rush hours. This assignment problem is described in detail in Abbink et al. (2004).

An adequate rolling stock capacity is one of the bottlenecks in the service provision by NS and it is important to look for the most effective allocation of the available rolling stock capacity among the trains, especially during the rush hours. This allocation is done after the line system and the timetable have been completed. Inputs are the (commercially) preferred train types per line. These preferences take into account the train category of the train series, the technical possibilities of the train types and subtypes, their running time characteristics, the expected number of passengers per train, and the passengers' preferences.

The trains that run in parallel, at one particular moment of the day, make up a so-called cross-section. The eight o'clock cross-section indicates all trains that run at 8:00h a.m., which is usually the busiest moment of the day. The circulation time of a train series, including the return times at the endpoints, and its frequency determine the number of cross-section trains of the train series.

For example, the regional train series 8800 shown in Figure 2.12, contains four eight o'clock cross-section trains, namely the trains 8820, 8822, 8829, and 8831. This is because the circulation time between Utrecht and Leiden and vice versa is about two hours and there are two trains per hour in each direction. The eight o'clock cross-section trains of all train series together make up the complete eight o'clock cross-section.
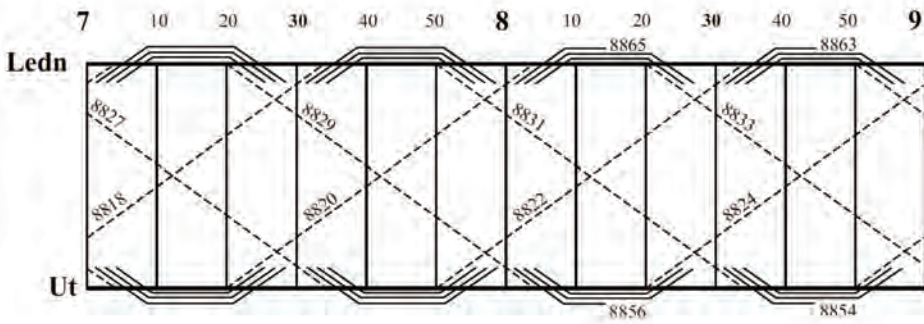
Figure 2.12. Part of the Rolling Stock Circulation on Utrecht
- Leiden

As a first step in the planning process of the rolling stock, the allocation of the rolling stock capacity to the trains in the eight o'clock cross-section is determined. Here, the idea is that if it is possible to determine an appropriate allocation of the rolling stock capacity during the morning rush hours, then this allocation will be appropriate during the other hours of the day as well. This is reasonable because, in practice, the required capacity during the evening rush is usually less than during the morning rush: The evening rush lasts longer than the morning rush, and it has a lower peak. The allocation of the rolling stock capacity to the eight o'clock cross-section trains is an example of bottleneck planning, where the aim is to maximize the effective rolling stock capacity.

The central question in this planning step is: *How many units of each train type and subtype should be deployed on each train, within the eight o'clock cross-section, to maximize the effective capacity for transporting passengers?*

First, the preferred equipment is to be taken into account as much as possible, when determining the allocation of equipment in a train series. It is, however, not always possible to deploy the most preferred equipment in a train series. An important reason for this is the limited availability of rolling stock. Hence, even if a particular type of equipment is preferred, often alternatives have to be chosen because not enough train units of the preferred type are available.

Secondly, the allocation is determined by the required capacity. Train units of different types usually have different capacities. Therefore, depending on the required capacities, one particular type of equipment is more suitable than another.

Furthermore, it is a strict requirement that the length of each train does not exceed the length of the shortest platform along the train's route. Hence, if a busy

train series contains a station with a short platform, then a train type with a large capacity per carriage (such as double-deckers) will have to be chosen.

Finally, there are restrictions for the regional train series regarding the running time characteristics. Especially on a train series with stops at a relatively short distance of each other, the allocated rolling stock should be able to accelerate and brake quickly.

On each train, at most one rolling stock type can be allocated because units of different rolling stock types cannot be combined into one train composition. Furthermore, on each train series, it is desirable to have as few as possible rolling stock types and subtypes: The latter may lead to increased robustness of the railway system because the adjustments by traffic control become much simpler. For each train series, the maximum allowed numbers of rolling stock types and subtypes depend on the number of cross-section trains in the train series.

After the allocation of the rolling stock has been planned for the eight o'clock cross-section trains, the rolling stock circulation is also planned for the other trains during the day. This is described in the next paragraph.

Rolling-Stock Circulations

This planning problem addresses the assignment of the rolling stock units to the planned timetable. Many European railway operators nowadays use electrical train units, which can drive in either direction without a locomotive, to operate most of their trains. As explained before, these units exist in different types (e.g., single-deck or double-deck) and subtypes.

The goal in scheduling rolling stock is to allocate an appropriate amount of the appropriate rolling-stock type to each train in the given timetable. NS must find a balance between three conflicting objectives in rolling-stock scheduling: (1) service, (2) efficiency, and (3) robustness.

In this context, *service* means offering as many passengers as possible a seat. The capacity of each train should be sufficient to transport the expected numbers of passengers. For example, an NS criterion is that each passenger must have a seat if the travel time is more than 15 minutes. This is because seat availability is an important factor that passengers use in deciding whether to travel by train or not.

Efficiency aims at minimizing the amount of rolling stock and the number of kilometers that the rolling stock need to perform, which implies that the capacity of a train should not exceed the demand too much. There is also a constraint on the number of train units available; during peak hours, most trains will simultaneously

require a large composition of units. A further complexity is that demand varies substantially during the day and on a line. For example, workdays have two peaks, one in the morning and one in the afternoon, with high travel demands in opposite directions. The rest of the day is the off-peak period with a lower, yet considerable, travel demand. To operate the rolling stock efficiently, NS addresses this demand-variation problem by adjusting the lengths of the trains during the day. Due to the choice of operating a cyclic timetable, the only way to adjust capacity is either to skip some (peak) trains completely or to adjust the length of the train.

Adjusting the length implies that units have to be shunted from and towards the shunting areas near the stations. These adjustments result in many shunting movements. NS addresses *robustness* by reducing the number of shunting movements and by having a line-based rolling-stock circulation. At the densely used infrastructure, shunting movements interfere with the regular train movements. Next to that, coupling and decoupling of units are operations that can be unsuccessful, due to technical problems, and can lead to delays. A line-based rolling stock circulation prevents spreading delays to the different lines. Next to that, NS prefers to have a single type of rolling stock per line because this simplifies recovery if there is a disruption in the real-time operations.

Determining the order of the different train units in a train is a difficult aspect of rolling-stock scheduling. In Figure 2.13, an example is given of a rolling-stock plan on the line The Hague/Rotterdam–Utrecht–Zwolle–Leeuwarden/Groningen. A white wagon in the left picture corresponds to a unit with three carriages (A) and a black wagon to a unit with four carriages (B). In this example, two trains depart from The Hague and Rotterdam with compositions AB and BA, respectively. The right character is the front unit of the train. Upon arrival in Utrecht, the trains are coupled onto each other within a few minutes; this results in a single train with composition ABBA. In Zwolle, the train is split again into one train bound for Groningen and one for Leeuwarden. Moreover, because the travel demand in the Northern part of the line is relatively low, the last train unit of the train is uncoupled in Zwolle. The example shows that it is necessary to know the order of the units in the train. The train arrives in Groningen and Leeuwarden with compositions BA and B.

Recently NS changed the structure of the line system, and this structure of combining and splitting trains is not present in the current timetable. However, to couple and uncouple train units, one still needs to know the order of the units in the train. One simply cannot detach the middle train unit at a station. Also, the front units cannot be uncoupled at a station where the train continues in the same direction.
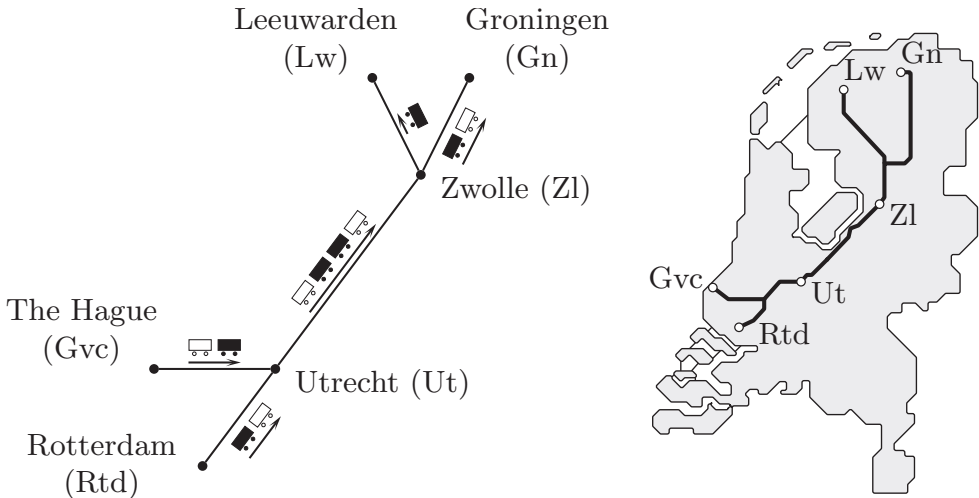
Figure 2.13.   A rolling–stock plan on the line The Hague/Rotterdam–Utrecht–Zwolle–Leeuwarden/Groningen.

Generally, there is not enough time in the timetable to perform this operation.

After the rolling stock circulation has been planned throughout the day and for each day of the week, it will also be balanced across the days of the week. On each shunting yard, the number of train units stored there during the night will be equaled to the number of train units that are required there on the next morning. This is accomplished by modifying the rolling stock assignment on the early morning trains or the late evening trains, or by adding some so-called dead heading trains.

Shunt Planning

As written in the previous section, trains that arrive at a station do not necessarily depart in the same composition. This requires that additional rolling stock units are transported toward (or from) the platform at which they depart (or arrive) to free the platforms for successive trains. These local transports are called shunting movements. Additionally, trains that have their final destination at a certain station need to be parked at the sidings of a station for the night. It is not allowed to leave

units at all platforms due to security reasons. In the morning, rolling stock units are repositioned again to the platforms.

A complicating aspect of parking the units is that there is limited capacity for placing the units at the shunt yard. Next to that, several operations like, for example, washing and cleaning the units, must be performed during the night. These operations require additional shunting movements. The shunting movements require both availability of the rail-infrastructure and a so called shunt driver. These shunting movements are planned in advance. Two important goals are taken into account.

The first goal is to minimize the number of shunting movements. This to improve the robustness of the plan, as explained in the previous paragraph. Next to this, shunting yards in the Netherlands are often located within the cities, and because of the noise generated by a shunting movement, NS would like to shunt as little as possible. Next to that, there are official limits on the number of movements per yard, imposed by the government. Additionally, each movement requires a driver, and there is a limited number of drivers available to perform the tasks.

The second goal is to maximize the robustness of the shunt schedule. This is dealt with by choosing the configurations and locations of the units at the shunt track such that the railway process can start-up in the morning with the smallest probability of disruptions. For each departing train, it is preferred to have a single shunting movement providing the required rolling stock composition. This requires that the compositions are already created at the shunt tracks. Next to that, the routing of the composition towards the platforms should be easy, without too many turnarounds. For a more detailed description on this problem, we refer to Abbink (2006).

2.4.3   Crew Scheduling

The crew scheduling process includes two major steps: (1) crew duty scheduling, and (2) crew roster planning. A duty starts and ends in a crew depot and describes the consecutive tasks for a single crew member. For each day, a set of anonymous duties is generated. Rosters prescribe how to assign the anonymous duties to individual crew members on consecutive days. Within NS, the duties are planned centrally for all crew depots, while the crews at the depots generate the rosters themselves. Both steps are described in the next paragraphs.

Crew Duty Scheduling

In this step, duties are constructed, where a duty is the work for one anonymous crew member on a single day. More formally, a *task* is the smallest amount of work that has to be assigned to one driver. At NS, a task typically contains one or two trips defined by the timetable. Here a *trip* is the movement of a train between a departure and an arrival, as introduced in Section 2.4.1. A *duty* is the work for one crew member from a crew depot on a certain day. It is inevitable that the crew scheduling step must be carried out centrally because most of the tasks could be assigned to several crew depots. Moreover, the crew costs are basically determined in this step.

Because a crew member can be relieved at all major stations, every time-tabled train, out of the mentioned 4,700 trains per day, results in about 3 tasks on average. A typical crew scheduling instance of NS, related to a single working day, requires assigning about 15,000 tasks to 1,000+ duties for drivers, and about 18,000 tasks to 1,300+ duties for conductors. Many labor rule constraints are defined per week, e.g., the average duty length per depot should be 8 hours. Therefore, it is attractive to solve the problem for a complete week instead of each day individually. As a consequence, this results in huge crew scheduling instances.

Figure 2.14 shows some examples of duties and duty rules. The figure shows several lines, each representing a duty. The first abbreviation (e.g., Asd = Amsterdam) on a line denotes the crew depot to which the tasks are assigned. The duty contains a sequence of tasks, the small line segments, mostly accompanied with a number of the train on which the task is performed. The abbreviations below the tasks show the stations at which the tasks end, or at which the next task begins. As can be seen the last station is equal to the crew depot again, because the duty has to start and end at the same station.

The duties have to fulfill a lot of requirements. The duty rules are illustrated with the gray rectangles. For example, there is an upper bound on the length of each duty, and there should be a meal break scheduled in each duty. Also, some time is needed for briefing and debriefing in the duties. And, the personnel base for which the duty is constructed, should have the knowledge on rolling stock and route knowledge, related to the tasks that are included in the duties. Also, there needs to be a minimum time between two consecutive tasks, the time needed to transfer from one train to another, including some buffer time for small delays. The small line beneath the tasks depicts that the tasks share rolling stock, which means that a single unit of rolling stock is used for the tasks. In this case a smaller transfer time
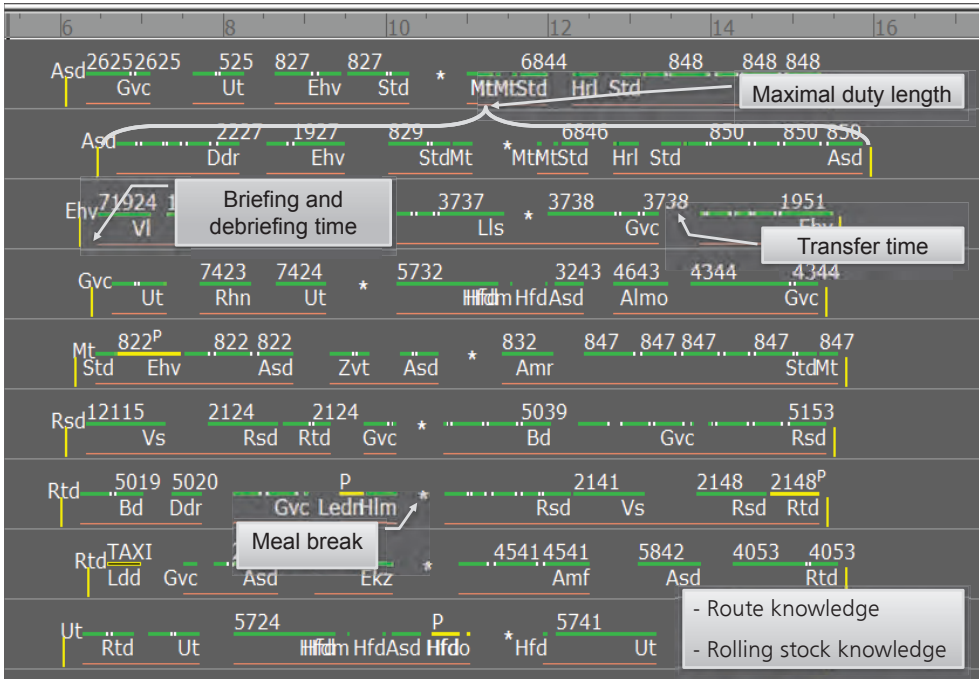
Figure 2.14. Duty examples

is allowed, because the second task will not depart before the first one arrives, even if the train is delayed.

In the crew duty scheduling problem that is solved for generating the generic annual plan, some rostering aspects are also taken into account.

Figure 2.15 shows some examples of duty roster rules. For example at NS, the average duty length over all duties on a crew depot should not exceed 8 hours. The reason is that, if this time is exceeded, then it is impossible to construct rosters where the average working time per week is equal to or less than 36 hours (in principle each full-time crew member works 9 days in two weeks). The number of night duties (duties with a working period between 1:00h and 5:00h) in a roster is also limited. This constraint should also be validated at a weekly basis.

Moreover, the work should be fairly spread over the different depots, to obtain a fair division of work over the week for the different crew members. The latter constraints are typical for the Dutch situation and are known as "Sharing-Sweet-and-Sour" rules. They aim at allocating the popular and the unpopular work as fairly as possible among the different crew depots. For example, some routes are more popular than others, and intercity trains are preferred over regional trains.
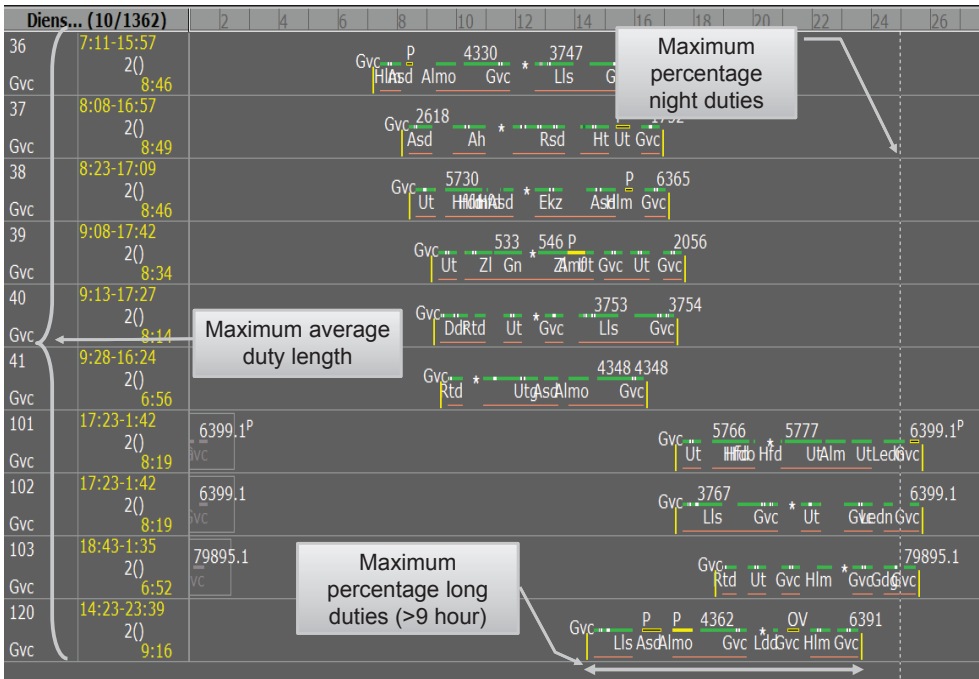
Figure 2.15. Duty Roster Rules Examples

The scheduling rules describe quantitative norms for dividing the "sweet and sour" workloads among the crew depots. For example, there is a lower bound for the number of unique railway track kilometers per crew depot, and there is an upper bound for the percentage of work per crew depot on lines with a lot of anticipated passenger aggression. To represent the requested fair allocation of the workload among the crew depots, the rules also contain upper bounds for the standard deviations over these percentages. Altogether, these rules are based on norms that can be checked objectively.

The norms for the standard deviations in the set of rules indicate that it is impossible to realize a 100 percent fair allocation of the "sweet and sour" workloads. For example, some lines are notorious for passenger aggression. When a crew depot is located at such a line, it will get a larger part of the work on that line. Other depots can work on such a line as well, but trains that start and end in a depot normally are assigned to the corresponding crew depot. This is because it would otherwise require a crew member to arrive (or depart) at the station by means of alternative (expensive) transport. At these moments of the day they cannot get there by train. In practice this means that some of the "sour" work can be allocated to other depots,

but not every depot will get an equal amount of "sour" work. Similarly, dividing the "sweet" workload fairly means that several crew depots have less "sweet" work than others.

The rules also prescribe a sufficiently high variety of work in the duties. To accomplish this variety, NS introduced the concept of repetition-in-duty (RID). NS divided the railway network into a number of routes, and, based on this division, defined the RID_d of duty d as follows:

$$\text{RID}_d = \frac{\text{\# routes in duty } d}{\#\text{different routes in duty } d}$$

The associated rule specifies that the overall average RID over all duties should be less than 2.7. For each crew depot, the average RID should be less than 3.0. In other words, on average each duty should contain a certain route at most three times.

In Figure 2.16, an example is given of different sets of duties containing the same set of tasks. A small line between two tasks indicates that they share the same rolling stock. The "*" indicates that there is a possibility for a meal break. The different gray tones indicate that the trains operate on different routes.
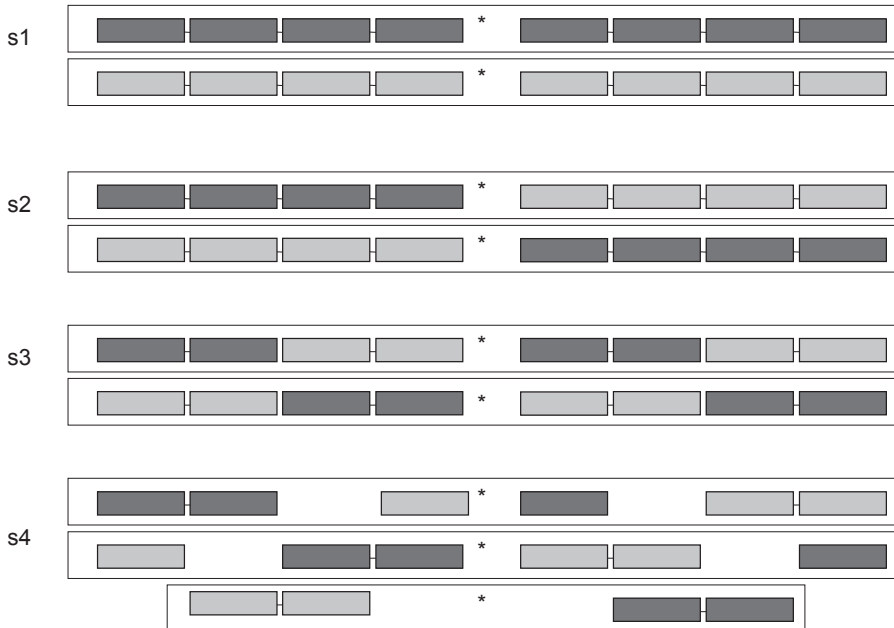
Figure 2.16. Possible Duty Configurations

The first solution $s1$ shows that two duties can cover the set of tasks in such a way that the crew member only operates on a single route. This is a good solution for robustness because possible delays will not be transferred to other lines. Also, it is efficient because the solution requires the minimum amount of two duties. Driving on a single line is considered to be monotonous work, therefore, this solution is not so nice from a crew perspective. Monotonous work not only gets boring, but is also considered to be dangerous. Studies have shown that experienced drivers, operating a line very often, anticipate on the signals, which increases the risk of passing a stopping signal.

Solution $s2$ scores better on the acceptability aspect because at least two different lines are operated in a single duty. The efficiency is on par with the first solution, and there is only a slight possibility of transferring delay to the other lines during the meal break.

The acceptability of solution $s3$ is better than that of $s2$ because the number of consecutive tasks on a single route is less. However, there is only a small transfer time between tasks on different routes, which makes the solution worse from a robustness point of view.

In solution $s4$, the acceptability is even better because the number of consecutive tasks on a single route is even further reduced. Also, the short transfers between routes are eliminated. However, this solution is less efficient because one additional duty is needed for this robust and attractive solution. This example illustrates the trade-of that can be made between the three major goals.

Quality aspects of crew duty scheduling We discuss the quality attributes of crew duty schedules, based on the 3 aspects introduced in Section 1.1: (1) efficiency, (2) robustness, and (3) quality of work.

Efficiency means that the total crew costs are as small as possible. These costs directly relate to the number of crew members needed to operate the schedules. In this aspect, a duty schedule is more efficient when fewer duties are scheduled to operate all tasks. Next to the number of duties, an indicator for the efficiency is the average working time within a duty. At NS, this is about 60%. This means that, on average, the total duration of the tasks in a duty is about 60% of the duty duration. The remaining time is spend on briefing and debriefing, transfers between trains and (meal) breaks. Because the amount of work is given by the timetable, a higher percentage of working time means that fewer duties are scheduled.

The *robustness* of the crew duties, i.e., preventing the propagation of delays via

the crew schedules, depends on several elements, including the transfer times of the crews when transferring from one train to another. Robustness is often addressed in the crew duty scheduling problem by using constraints. It is hard to define a value to robustness, which can be optimized, but it is clear that buffer time between tasks on two different trains will limit propagation of delays. Therefore, we include a minimum transfer time between two consecutive tasks in a duty.

The *quality of work* is the perceived quality of duties by the crew members. This is addressed via labor rules and company agreements, for example, on the amount of variation in the duties. In this section, we presented some examples of indicators, used to determine whether the duties have a sufficient quality of work. In Section 3.7, we discuss how these indicators can be included in the decision support systems.

Crew Roster Planning

In this step, rosters are created where duties are assigned to the individual crew members.

Rosters can be created in several ways: (*i*) A roster for individual crew members can be created where crew specific characteristics (e.g., their vacations) can also be taken into account, (*ii*) a bid line can be constructed on which individual crew members can bid, or (*iii*) a cyclic roster can be constructed. The first two rostering approaches are mainly used in the airline industry (see Kohl and Karish (2004) for an overview). However, many European public transport companies, including NS, use the concept of cyclic rosters. In the remainder of the paper, we refer to a cyclic roster if we use the term roster.

In the Cyclic Crew Rostering Problem (CCRP), rosters are created for a group of crew members, where crew members with the same characteristics (e.g., drivers, full-time employees, and same route knowledge) are in the same group. For such a group, one roster is constructed with a length in weeks equal to the number of crew members in the group. If a roster is of size k, then k indicates both the number of weeks and the number of crew members in the roster.

The input for the CCRP consists of a set of duties for each day of the week. Since the roster is cyclic, all weeks have the same duties. Furthermore, a roster has to satisfy many labor rules related to days off, working time, etc. Schematically, a roster can then be seen as a set of rows and columns, where the columns correspond to the different days of the week, and the rows correspond to the different weeks.

Table 2.2 gives an example of a roster for 6 weeks, which is to be carried out by 6 crew members. The letters in the example represent the type of duty scheduled

Wk	Mo	Tu	We	Th	Fr	Sa	Su
1	RES	E	R	L	WTV	R	R
2	E	E	E	E	E	E	R
3	R	L	WTV	R	R	E	E
4	E	E	E	E	E	R	R
5	L	L	WTV	R	L	N	N
6	N	R	R	L	L	L	L

Table 2.2. Small Roster Example

on that day: Early duties (E), late duties (L), night duties (N), rest days (R), WTV days (WTV), and reserve days (RES). These types will be explained in more detail later in this section. Crew member 1 starts in week 1 with the duties in the first row of the roster, while crew member 2 starts in the first week with the duties in the second row. These duties are carried out by crew member 1 in the second week. The remaining part of the roster can be explained similarly. Notice that the duties in the first row follow after the duties in the last row.

In the CCRP, one should take into account all kinds of constraints following from the labor rules and the collective labor agreements. Regular duties are categorized in three types; early, late and night duties, depending on the time of the day on which they start and end. Next to duties and days off, certain other types of days should be scheduled. These are the so-called WTV, CO and RES days. A WTV day is in principle a day off for a crew member. However, a crew member can decide to sell this day off and go to work. For the rostering process, this means that it should be possible to schedule a duty on a WTV day. A CO day is an extra day off that crew members get when they have worked a certain amount of time in the weekend, early in the morning or in the night.

If an individual crew member does not reach a certain level of CO time (e.g., due to illness), he should work on this day. A RES day is a day where a crew member should be available for work in order to replace other crew members that are ill or are on vacation. In the following, a roster day represents either a duty, a day off, a WTV day, a CO day, or a RES day. Below, we give some examples of rules that have to be taken into account in the CCRP: [2]

- The rest time between two duties is at least 12 hours if the first duty finishes before 2:00 a.m., otherwise it is 14 hours.

---

[2]For an extensive overview, we refer to Hartog (2005).

- The maximum working time per week is 45 hours. For a period of 13 consecutive weeks, the average working time per week is at most 40 hours.

- A day off is a period of at least 30 hours between two duties.

- At least once in the three weeks, there is a so-called *Red Weekend*. This is a rest period of at least 60 hours, starting not later than Saturday at midnight and ending not earlier than Monday 4:00 a.m.

- In a full-time roster of length n, there should be planned $2n$ days off and $n/2$ WTV-days.

- There are at most 10 duties per 4 weeks, and 32 per 13 weeks, which cover part of the period between midnight and 6:00 a.m.

The quality of a roster is determined by the order of the roster days and the variety in the duties. Some examples are:

- A series of duties after each other with the same type is preferred, where the type is either an early, late or night duty.

- Two or more adjacent days off (or CO-days) are preferred.

- WTV-days are preferably spread over the different days in the week and are preferably scheduled once in the two weeks.

- Similar duties should be spread over the roster. A lot of variety in the work with respect to routes, rolling stock types, etc. is preferred.

Since each duty belongs to a crew depot, the CCRP should be solved for each crew depot separately. To increase the influence of the crew members on the final rosters, the rosters are constructed locally at the different crew depots themselves. An important aspect for the crew members is a fair division of the popular and the unpopular parts of the work.

The most important rostering problem is to be solved each year at the beginning of a new timetable year, since then the basic rosters are constructed from scratch. During the year, the rosters are only slightly modified. The latter is less complex than constructing a roster from scratch.

Quality aspects of crew rosters   We discuss the quality attributes of crew rosters, based on the 3 aspects introduced in Section 1.1: (1) efficiency, (2) robustness, and (3) quality of work.

Efficiency means that the total crew costs are as small as possible. These costs directly relate to the number of crew members needed to operate the rosters. The efficiency of the rosters is quantified by the number of crew members assigned to the rosters to be operated. NS creates groups of a certain size, and creates cyclic rosters for these groups. The size of the group is equal to the number of crew members. Another indicator for the efficiency of the rosters, is the amount of work assigned to the rosters. If the available working time in a roster is 36 hours on average, and the assigned duties have an average duration of 35 hours, then it is clear that the efficiency could be improved by adding more work to the roster. Another option would be to reduce the size of the roster and the available working time.

For the rosters, the operational *robustness* can be improved by adding spare crew duties, to be used in case of disruptions in real time operations. A higher number of spare duties increases robustness, but, on the other hand, reduces the efficiency of the rosters because more crew is needed.

In crew rostering, the *quality of work* can be measured by computing the number of preferred and non-preferred patterns of consecutive days of work. In this section we presented some examples of patterns used in the crew depot Utrecht. Each pattern gets a score, positive for preferred patterns and negative for non-preferred patterns, and the goal is to maximize this sum of the scores.

2.5   Operational Planning

In operational planning, plans are created for calendar days. Since these plans need to be created for every calendar day, it would simply be too much work to create these plans from scratch, therefore only necessary changes are made to the generic plans.

These include changes due to rail infrastructure maintenance works and due to events, like sports events, which require additional capacity. Most of the changes are caused by maintenance work. The problems described in this section show a lot of similarities with the ones described in the previous section on tactical planning. Therefore, we mainly describe the differences with the previously described problems.

2.5.1   Timetable Rescheduling

In this planning phase, the timetable is updated according the required changes to the plan. Extra trains can be scheduled for events, or trains can be canceled because a part of the rail network cannot be used during maintenance work at those tracks. Sometimes it is sufficient to schedule another route for the train, so the service can still be provided. As stated, the timetable is updated, which means that only small changes are made, and the goal is to keep as much as possible of the original plan created in the tactical planning phase.

2.5.2   Rolling Stock Rescheduling

Due to the changes in the timetable, also the scheduled rolling stock duties need to be adapted. Assignments to canceled trains need to be removed, and added trains need a new rolling stock assignment. Also, due to changes in the timetable, it might be that the expected number of passengers per train changes as well, and a different seating capacity is required.

While creating the rolling stock duties, one of the goals is to minimize changes in the start and end balances (the number of units stored at a station during the night), and in the included couplings and uncouplings in the duties. Changes in these aspects require that the shunt plans need to be revised. Therefore, one of the goals is to limit the number of these changes.

As a result, it could be that the seating capacity and costs are not optimal, they can be improved when additional changes to the plan would be allowed, or even when plans could completely be rescheduled. The main problem in this is that shunt planning is still a very labor-intensive process.

It can happen that the shunting areas cannot be reached or are difficult to reach, due to maintenance work at the infrastructure. Either because the track that leads to the area is out of service, or because part of the route towards the station where the area is located is out of service. This implies that the capacity to park the rolling stock units can be limited. Therefore, rolling stock duties need to be adapted such that the number of units at a station, at the start and end of a day, does not exceed the number that can be parked.

For maintenance purposes, NS lets each rolling stock unit visit a maintenance facility on a regular basis, generally, after approximately 30,000 km. At the facility, the unit is checked and, if necessary, repaired. Next to that, preventive maintenance can be carried out. At NS, each day it is decided which units need to visit a maintenance

facility in the coming days, and how they are routed towards that facility. To prevent additional costs, this routing is done preferably with a minimum of additional dead-headings (train movements without passengers).

Usually, each rolling stock unit has been assigned to a chain of train movements during the planning of the rolling stock circulations. These chains determine the path through the network that a unit will operate during several days. Some of these chains pass by a maintenance facility in the coming days.

The problem is to assign units, that need maintenance, to a chain in which it is passing a maintenance facility at the right time interval, see also Maróti (2006). This assignment can be achieved by *swaps* of chains. If two units, of the same type, have an overlap of standstill time at the same station, they can be swapped. This means that they will operate the remainder of each other's scheduled chain of trains. Sometimes multiple swaps are needed to get the unit in an appropriate chain. Each swap, generally, requires some shunting at the involved station. If swapping is not possible, for example, in case of urgent repair, the unit can be transported by dead-heading towards the maintenance facility.

2.5.3   Crew Rescheduling

Due to the changes in the timetable and the rolling stock schedules, also the scheduled crew duties need to be adapted. Canceled trains need to be removed from the duties, and added trains need a driver and one or more conductors. Also, due to changes in the rolling stock schedules, it might be that the required number of conductors per train changes as well.

The crew schedules for the calendar days (e.g., Friday, October 3, 2014), are based on the schedules for the corresponding generic day (e.g. Friday). To limit changes to the rosters, the time intervals of the scheduled duties must not change too much. Therefore, duties can start early or end late, with a maximum difference of half an hour. These small changes in time generally fit in the rosters created in the tactical planning phase. Also, the goal is to change the contents of the duties as little as possible. Other tasks might need different route knowledge or rolling stock knowledge. This might require to re-assign a duty to another crew member in the roster. A change of contents of the duties might also lead to a violation of the "Sharing-Sweet-and-Sour" rules, introduced in Section 2.4.3, which is not allowed.

Included in the duties for the generic days are a set of reserve duties. These can be used in case additional duties are needed to accommodate the extra trains that need to be operated on a calendar day. These duties also can be used to limit the number

of changes to the other scheduled duties. Overall, modifications in the generic crew schedules for adapting them to the calendar days should fit in the rosters. Only if the CSP cannot be solved otherwise, the rosters may be modified. For a description on the Operations Research models used to solve this crew duty rescheduling problem, we refer to Huisman (2007). A modification of the roster might also be required due to absences of the crew members themselves. It might be that they request a leave, need to attend a training session, or become ill. In this case, the duties are reassigned to other crew members. For this reason, there are reserve duties included in the rosters. With one or more swaps, the duties can be reassigned to one of these reserves. In general, changes to the rosters are limited, and the crew members know their required attendance at work already a long time in advance.

Quality aspects of crew rescheduling We discuss the quality attributes of adapted crew schedules, based on the 3 aspects introduced in Section 1.1: (1) efficiency, (2) robustness, and (3) quality of work.

Efficiency means that the total crew costs are as small as possible. The number of crew members is already given in this planning phase. Therefore, the costs in this phase relate to the compensations paid for late modifications to the rosters. Therefore, the main focus is on producing the rescheduled duties in time, before additional compensation fees need to be paid to the employees.

The *robustness* of the crew duties, i.e., preventing the propagation of delays via the crew schedules, is handled similarly to the scheduling phase. E.g., minimum transfer times between two consecutive tasks are required during rescheduling as well.

The *quality of work* is the perceived quality of duties by the crew members. The original duties and rosters are considered to be of good quality. Therefore, one would like to limit changes to the duties and the rosters. During rescheduling, the objective is to limit the number of changes, computed by the number of tasks that changed in the duties and by the minutes that the start and end times of the duties shift in time.

2.6 Operations Control

After the planning phases, the daily plans are carried out in the real-time operations. Preferably, the plans are carried out exactly as scheduled. However, in the real-time operations plans have to be updated continuously in order to deal with delays of

trains and larger disruptions of the railway system. A disruption may be due to an incident, or a breakdown of infrastructure or rolling stock. On the Dutch rail network (with more than 5,000 daily trains), on average 10 disruptions of (part of) a route occur per day. Delays occur more frequently: About 450 trains experience one or more delays (≥ 3 minutes) per day. These delays lead to the removal of on average 10 train services per day.

The problems solved in operations control are very similar to operational planning, with the major difference that decisions need to be made in minutes. Moreover, the decisions need to be made under a lot of uncertainty; estimates of the time-span of a disruption are often not accurate.

2.6.1   Timetable Dispatching

For handling small delays, rules of thumb are in place. For example, it is predefined for connecting trains (trains for which NS offers a transfer to the customer), how many minutes a train should wait for the arrival of another delayed train. For more details on this problem, we refer to Dollevoet (2013).

Large disruptions, e.g., a complete blockage of a track, where several train services need to be canceled, require a fast response to prevent spreading the problems to other parts of the network. Therefore, predefined solutions are developed in advance. These predefined solutions describe an alternative timetable for the part of the network on which the disruption occurs. The problem is to select the right solution and to adjust this solution to the specific circumstances on the day of operations.

For example, in case of a blockage one can decide to operate regional train services dwelling at every station close to the disruption, and to skip the intercity train services between the larger stations at the ends railway line where the blockage occurs. When a train service starts and ends in a different station, this requires that the routing through the station is adapted, which is, generally, included in the solutions.

The predefined solutions do not include specific situations. For example, when there is a blockage where a train is involved in an accident, it might be needed to send a locomotive or other train to the location of the incident to tow the train away. Also, it might be useful to let the first intercity train, that runs after the blockage is finished, stop at stations where passengers have crowded due to the limited services during the blockage. These exceptions are not included in the predefined solutions and are handled by ad-hoc measures.

2.6.2 Rolling Stock Dispatching

For small delays, no changes are made to the rolling stock schedules. This means that the delays simply propagate to the next trains to which the rolling stock units are assigned, until the buffers in the schedules have absorbed the delays. Important in this are the buffers at the turning stations. In general, these buffers are sufficient to absorb the delays.

If a defect in a rolling stock unit is repaired within a few minutes, the train can depart with some minutes delay. If repairing a train takes more time, the unit needs to be put at a siding track and cannot be operated as planned anymore. If possible, a spare unit will replace the broken unit, otherwise, trains will be operated with smaller capacity.

When the timetable is changed due to a large disruption or canceled train, the rolling stock schedules need to be adapted, as well. Similar to the tactical planning phase, the schedules need to be adapted, fulfilling the operational constraints, with the general goal to minimize changes to the original plan. Changes to the plan may require changes to the local shunting plan, supplying and storing of the rolling stock units needed for operating the schedules. For more information on the rolling stock rescheduling problem, we refer to Nielsen (2011).

2.6.3 Crew Dispatching

Each working day, a crew member is scheduled to carry out a number of tasks, according to the planned duty assigned to him/her. A number of duties may become infeasible, due to cancellations and delays of trains or rescheduling of the rolling stock. A duty may become infeasible due to a time conflict (often caused by delays) and/or a location conflict (often caused by canceled train services). In both cases, a conflict occurs between two consecutive tasks in the duty.

Each day, about 2300 duties are carried out (1000 drivers and 1300 conductors). Furthermore, at any moment in time, the number of active duties at that moment is about 700. Dispatchers are responsible for rescheduling tasks among crew members so that all trains are manned with sufficient staff. Next to that, a crew member can become sick, which also leads to reassigning or rescheduling the duties. A number of spare crew members is available to resolve rescheduling problems.

An example of a reassignment is given in Figure 2.17. One task is canceled, depicted in black, due to a canceled train service. This makes it impossible to perform the dark gray task (C–D) in the first duty. The task can be inserted in another duty
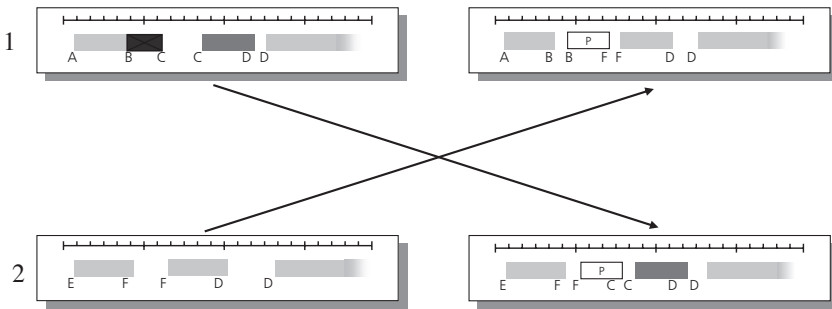
Figure 2.17. Example: Rescheduling of an infeasible duty

(requiring a positioning task depicted in white). This, in turn, requires the task from F to D in the second duty to be rescheduled. In this example, this task can be inserted in the first duty, basically swapping the two tasks.

In practice, rescheduling can be much harder than in the given example, requiring multiple changed duties. In the case of a large disruption, for example, a complete blockage of part of the railway network, the number of affected duties is large but a positive aspect is that they all have more or less a gap where inconsistent parts of other duties can be included. At both sides of the blockage, there are crew members that can exchange tasks with crew at the other side of the blockage. This property of the problem ensures that a solution can be found with only a limited number of changed duties.

Many of the constraints and rules that apply to the planning problems do not apply to the real-time crew rescheduling problem. There is no need to follow the "Sharing-Sweet-and-Sour" rules, transfer times can be minimal (even negative, which results in a train waiting for a late crew member), and duty duration can be extended. This makes the number of possible solutions sufficiently large to solve most of the problems.

Especially in the first phase after a disruption, when a train and its crew do not arrive at the planned station, there might be a shortage of crew for operating a departing train. There is no time to reposition crew members to this location in time, and therefore, a number of spare drivers are available at the important stations. In general, these stations are equal to the locations where the crew members are allowed to switch trains in the planning phase. There are more spare drivers than spare conductors, because a train is allowed to drive without a conductor or with fewer conductors, but can obviously not be operated without a driver.

Dispatchers are organized into four regions, where they are responsible for rescheduling train crew who currently reside in their region. Often dispatchers need to handle task-rescheduling recursion, which they can handle to a certain extent given the available time and resources. Typically, about five minutes are spent to resolve an inconsistent duty. Frequently, rescheduling problems are left "open ended" for later resolution by other dispatchers (often in another region). In larger disruptions, some trains simply cannot be operated, as dispatchers are busy rescheduling train crew, causing additional delays for passengers.

Quality aspects of crew dispatching We discuss the quality attributes of crew dispatching, based on the 3 aspects introduced in Section 1.1: (1) efficiency, (2) robustness, and (3) quality of work.

Efficiency means that the total crew costs are as small as possible. In real time operations, the number of crew members is given. Overtime due to arriving late at the crew depot, as a result of incidents and rescheduled duties, can be compensated. The indicator for this is the amount of additional working time, which should be minimized. Next to this, additional costs might occur when crew members need to be repositioned by taxi or other means of transport. In crew dispatching, the goal is to minimize these additional costs.

For *robustness*, also in crew dispatching the goal is to have sufficient transfer time. However, sometimes a dispatcher can choose to have a smaller (or even negative) transfer time if a train service has to be canceled otherwise. The use of spare crew is minimized in order to be able to react to future disruptions.

The *quality of work* is less of an issue in real-time dispatching. One tries to limit the number of changes, and also shortened transfers and meal-breaks are considered to be unattractive. However, the crew members understand that this is sometimes unavoidable.

2.7   Summary

In this chapter, we described the main railway planning problem and their important aspects. Sub-problems of the Crew Management Problem are described in the context of the other planning problems. The problems are illustrated by NS examples. This answers research question 1: *"What are the important aspects of Crew Management?"*

In addition to the general description of the problem, we elaborated on the important criteria for assessing the quality of a crew plan. In Sections 2.3.4, 2.4.3, 2.4.3, 2.5.3, and 2.6.3 we described the quality attributes of the crew plans. This answers research question 2: *"How can we quantify the quality of a plan?"*

Based on our knowledge of the problems and processes of other railway operators (most of the operators in Europe) we consider the situation at NS as representative for many other rail operators.

Chapter 3

Crew Scheduling

This chapter describes the development of decision support for the Crew Scheduling Problem (*CSP*) at NS. In Section 2.4.3, the CSP was described in general terms. The problem deals with assigning (driver and/or conductor) tasks, derived from the planned timetable, to anonymous duties. A duty is a day of work for one crew member. More formally we define the CSP as follows: *Given a set of tasks, find a feasible schedule with minimal costs such that all tasks are covered. In this context feasible means that all rules are satisfied.*

We will show that, because many rules deal with a whole week instead of a single day, it is more efficient to solve the crew scheduling problem (CSP) for a whole week as one optimization problem. Because we will consider instances for a whole week, the instances that we solve are a magnitude larger than one typically finds in the Operations Research (OR) literature. A typical crew scheduling instance of NS, related to a whole week, requires assigning about 90,000 timetabled tasks to 6,400+ duties for drivers and about 120,000 timetabled tasks to 7,600+ duties for conductors. NS is one of the first railway operators that successfully introduced advanced decision support systems in practice, for a CSP of this size.

This chapter describes the algorithms used to solve the problem and the applicability of these algorithms is illustrated by computational results. In this chapter, because of the complexity of the problem, we do not solve the problems to optimality. We consider a problem to be solved when the solution algorithm has found a solution and we can determine, based on the lower bounds, that the solution is near to the optimal solution.

3.1   Literature Overview and Related Work

Initially, research related to the application of Operations Research (OR) models and techniques (as introduced in the previous section) to solve crew scheduling problems dealt with the airline industry, see e.g., Barnhart et al. (1998), Desrosiers et al. (1995), Hoffman and Padberg (1993), and Wedelin (1995).

The airline industry has used such models to solve crew scheduling problems for many years. However, in the railway industry the sizes of the crew scheduling instances are, in general, a magnitude larger than in the airline industry. Moreover, crew can be relieved during the drive of a train resulting in many more tasks per duty than typical in airlines. In other words, the combinatorial explosion is much higher. The latter has made the application of these models in the railway industry prohibitive until the late nineties.

Developments in hardware and software enable the railway industry to use these models nowadays as well, see Caprara et al. (1997, 1999), Kohl (2003), Kroon and Fischetti (2001), and Kwan et al. (1999), among others. Fores et al. (2001) describe the work that has been done and used by several operators in the UK. In 2003, Kohl (2003) claimed to solve the largest Crew Scheduling Problem (CSP) in the world, namely the instance for one day of the German long-distance traffic. The instances that we will consider in this thesis are at least of the same magnitude.

The main optimization models and solution techniques used in these papers are introduced in Appendix A. In this appendix the Minimum Cost Flow problem and the Set Partitioning problem are described. Also, a basic introduction into Lagrangian Relaxation and Column Generation techniques is given there. These models and solution techniques are commonly used for crew scheduling problems and form the basis of the work performed at NS, to support the CSP.

3.2   Problem Formulation

As described in Section 2.4.3, a set of tasks is derived, based on the timetable and the rolling stock roster. We will formulate the CSP as a Set Covering Problem. Let T be the set of tasks to be covered. Furthermore, D denotes the set of potential duties. The subset D_t of D consists of the set of duties containing task $t \in T$. The binary decision variable x_d indicates whether duty $d \in D$ is included in the solution or not. Every duty d has positive costs c_d. Furthermore, let S be the set of additional constraints and let l_s and u_s be the lower and upper bound for constraint $s \in S$.

Finally, let w_d^s be the coefficient of duty $d \in D$ for constraint s.

Then the set-covering model, with additional constraints, can be formulated as follows:

$$\min \sum_{d \in D} c_d x_d \tag{3.1}$$

$$s.t. \quad \sum_{d \in D_t} x_d \geq 1 \qquad \forall t \in T, \tag{3.2}$$

$$l_s \leq \sum_{d \in D} w_d^s x_d \leq u_s \qquad \forall s \in S, \tag{3.3}$$

$$x_d \in \{0, 1\} \qquad \forall d \in D \tag{3.4}$$

Equation (3.1) is the objective function, which states that the sum of the duty costs is minimized. Constraints (3.2) guarantee that for each task t, at least one duty that contains this task is selected. It may sometimes be better to perform a task more than once. If, for example, the number of tasks going out of a crew depot differs from the number of tasks going into the crew depot on a day, over-covering is necessary. Moreover, even if over-covering is unnecessary, it may be efficient to allow over-covering. By allowing over-covered tasks, it can be that other tasks are covered easier, resulting in a larger decrease in costs than the additional costs for the over-covered tasks.

Besides the regular tasks that must be covered, the formulation also allows one to include a number of suggested or additional tasks in the model. E.g., one can add possible taxi trips to reposition crew members from one to another location. These taxi trips have higher costs, but might make the schedules feasible and/or more efficient. Tasks related to shunting activities are examples of suggested tasks. They can preferably be included in the drivers' schedules, but can also be performed by crew members locally at the stations, when the schedules become inefficient by including the suggested tasks.

These suggested and additional tasks may be covered with a duty, but they need not be covered. As stated, adding such tasks to an instance may be helpful for finding a feasible schedule or for improving the overall efficiency of the schedule. There is no constraint (3.2) corresponding to the additional tasks, which are considered only within the duty-generation module. For each additional task t, we have a dummy duty that covers task t only, and has a cost equal to the user-defined penalty for leaving task t uncovered.

The additional constraints (3.3) are not related to the individual duties (such

constraints are handled at the duty-generation level), but to certain forbidden com-
binations of duties. Often, these are constraints at the so-called depot-level. In these
constraints, one can see u_s as the availability of a certain resource and w_d^s as a param-
eter describing the amount of this resource that is used by duty d. These additional
constraints may be related to several issues, such as the number of duties per crew
depot, or the average length of the duties per crew depot.

For example, let K be the set of depots, if k_d and l_d denote the crew depot and
the length of duty d, respectively, and L denotes the maximum average length of the
duties of each crew depot (say, eight hours), then the following constraints guarantee
that the maximum average duty length for each crew depot $k \in K$ is respected.

$$\sum_{d:k_d=k} (l_d - L)x_d \leq 0 \qquad \forall k \in K \qquad\qquad (3.5)$$

The constraints (3.4) indicate that the decision variables are binary. The cost
coefficients c_d in (3.1) represent the fact that the main objective is to minimize
the number of duties required to cover all tasks. However, these coefficients also
handle additional penalties for discouraging undesirable characteristics of the duties,
such as an unfair allocation of the (Sharing-)Sweet–and–Sour workloads (see Section
2.4.3), uncovered suggested tasks, covered additional tasks, or positioning tasks. In
particular, we can use a penalty term for each unit of slack in constraints (3.2) to
reduce (or even forbid) positioning tasks (recall that, on a positioning task, a driver
or conductor is traveling as a passenger to the start of his next task).

3.3   Solution techniques

To solve the CSP formulated (as a Set Covering Problem) in the previous section, we
apply the technique of dynamic column generation in combination with Lagrangian
relaxation. As stated, for an introduction into these techniques we refer to Ap-
pendix A. This section describes how these techniques are used to solve the CSP.

For solving the CSP, the solution process usually consists of a duty-generation
module and a duty-selection module. The algorithm first generates a large set of
potential feasible duties. A duty is feasible if it satisfies all constraints at the duty
level, for example, the maximum duty length, the location, and the duration of the
meal break.

We apply column generation for duty-generation, which is frequently used for this
purpose. This technique does not generate all duties a priori, but on-the-fly during

the solution process. A reason to use this approach for generating the duties is that the set of feasible duties can be extremely large. Hence enumerating all possible duties a priori is not feasible.

The duties are generated by solving a resource constrained shortest path problem in an acyclic network $G = (N, A)$. In this network, the nodes (N) correspond to the tasks, and there are arcs (A) between two nodes if the two tasks can be assigned to the same crew member. Moreover, a path in the network corresponds to a feasible duty, when it starts (at source s) and finishes (at sink t) in the same depot, and all kind of additional constraints, such as duty length and break rules, are satisfied.

We define the costs of the arcs such that the total cost of a path is equal to the reduced cost of a duty. By finding the shortest, feasible path and checking whether its cost is negative or not, it is possible to check if there are still duties with negative reduced costs. If not, we stop the column generation procedure.

After the duty-generation, the duty-selection module aims at selecting a subset of these potential duties to cover each of the tasks with at least one of the selected duties, to satisfy the additional constraints at the depot-level, and to minimize the total costs. In other words, the algorithm solves the model (3.1) - (3.4) with the generated set of potential duties as input. All duties of a depot taken together must satisfy the additional constraints at the crew-depot level.

We use Lagrangian relaxation and sub-gradient optimization instead of linear programming to calculate the required dual information. Huisman et al. (2005), Huisman (2004), and Potthoff (2010) give the following reasons to use this approach to solve the CSP:

- For the CSP, a set of columns generated to compute a strong lower bound, is a set which can be used to select a reasonably good feasible solution.

- Lagrangian relaxation provides tight bounds for set partitioning type of problems (e.g., see Beasley (1990)).

- Using a linear programming relaxation is not a realistic option for some of the proposed models, because of the high number of constraints.

We organize the solution process in a sequence of iterations, in each of which the algorithm tries to obtain better and better solutions. Within each iteration, the algorithm iterates between the duty-generation module and the duty-selection module, in an attempt to update the best solution found. In this way, the algorithm typically considers several million of potential duties in the duty-generation phase,

and it constructs and evaluates thousands of alternative solutions during the selection phase.

3.4   Initial Solution Approach: TURNI

As described in Section 2.4.3, due to the complex set of labor rules, automated support in the crew scheduling process is necessary. Typical crew scheduling instances of NS produce set-covering instances that are much larger than the instances addressed in the literature so far, and they have many additional nasty crew-depot constraints. Even if we solve these problems for a single day, the resulting instances are extremely large.

The next three sections describe how we initially implemented a commercial software package named TURNI to support the CSP. After the initial implementation, we developed a heuristic to improve the results, which we describe in Section 3.5. Next, we started a new project where we joined forces with a commercial vendor to create a new algorithm called LUCIA, which resulted in a solution capable of solving the CSP instance as a whole for a whole week. We describe this algorithm and the corresponding results in Section 3.6. All the presented solutions are based on the application of dynamic column generation combined with Lagrangian relaxation as described in the previous section.

The first solution approach was to implement TURNI to support the CSP. NS has been using the automated crew scheduling algorithm TURNI since 2000 (until 2007, when it was replaced by LUCIA, see Section 3.6). NS successfully used the TURNI system in practice, see Abbink et al. (2005), and the business effect of using the system is discussed in Appendix B.

Because of the highly sophisticated algorithms included, TURNI solves extremely difficult crew scheduling instances in 24 hours of computing time on a personal computer. Therefore, NS could construct all crew schedules for all days of the week within just a few days. TURNI was developed by Double-Click sas [1], which has customized it several times to cope with the complex rules that govern NS crew schedules.

3.4.1   TURNI solution method

In the initial approach, we generated the crew schedules using the TURNI system. TURNI (Kroon and Fischetti (2001)) includes an algorithm specifically developed to

---

[1] www.turni.it

solve large crew scheduling problems. The model behind TURNI is a set-covering model as introduced in Section 3.2. TURNI solves the set-covering model by applying column generation combined with a Lagrangian-based heuristic. This Lagrangian-based heuristic, called CFT-heuristic, in which CFT stands for Caprara, Fischetti and Toth, forms the basis of TURNI (see Caprara et al. (1999)). The main characteristics of the heuristic are a dynamic pricing scheme for the variables, coupled with subgradient optimization and greedy heuristics, and the systematic use of column fixing to obtain improved solutions.

As described, the algorithm iterates between the duty-generation module and the duty-selection module. The algorithm deletes feasible duties that were generated in earlier stages and whose effectiveness turns out to be low during later stages of the process. The latter is done to keep the number of active duties manageable. These active duties are the potentially "good" duties. The duty-selection module heuristically looks for a solution for the overall model, based on the currently available set of feasible duties. The algorithm applies the generation and selection modules cyclically for a number of iterations, in an attempt to update the best solution found.

Thereafter, it activates a fixing procedure to select some duties that appear to be particularly efficient and to fix them as belonging to the final solution. Then the algorithm repeats the overall process on the tasks not covered by the fixed duties; it iterates the duty-generation and duty-selection phases for a while on the sub-problem (without the fixed duties), fixes new duties, and so on. It also applies a heuristic refinement procedure from time to time, in an attempt to improve the best solution by locally exchanging tasks among duties.

A relevant detail is the fact that in the TURNI output, not only the final set of created duties is presented, but also a large number of "good" duties are available, generated during the process. We will use this additional information for constructing sub-cases as described in Section 3.5.2.

3.4.2   Computational Results TURNI

Figure 3.1 and 3.2 illustrate the performance of TURNI on a real-life case involving 14,678 tasks related to all the work for the drivers on a typical Tuesday.

Figure 3.1 plots the number of duties in the current best solution (Heu. Sol.) and the current number of fixed duties (Fixed duties) over the CPU time.[2] The first fixing phase produces a solution involving 982 drivers after about 2 hours. Pass 1 ends after 4:06h and delivers an almost-optimal solution with 971 drivers. The

---

[2]hours on a PC Pentium 4 with 512 Mbyte RAM and a clock speed of 3.0 GHz
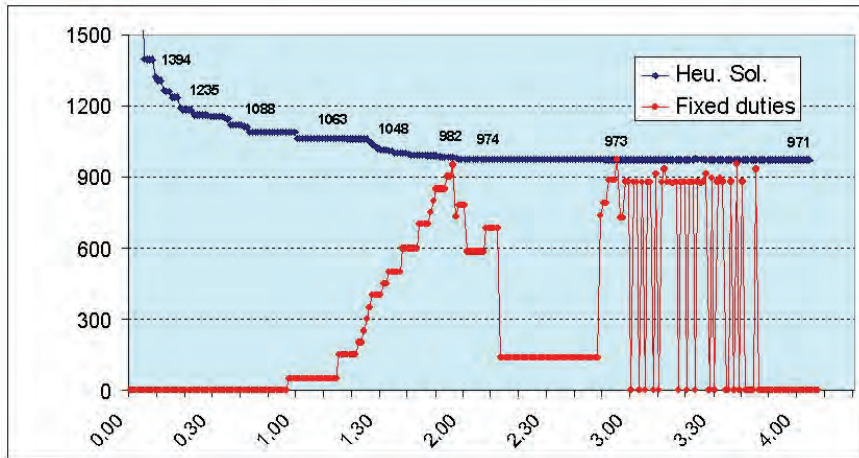
Figure 3.1. Computational results of TURNI

eventually best solution delivered by TURNI had 965 duties (only 0.6 percent less). This best solution was found in pass 3, after 14:47h of computation. The cost of this solution was slightly refined in subsequent passes. The results represented in the figure are typical for runs of TURNI on instances of comparable size.

Figure 3.2 shows the Lagrangian lower bound, given the generated set of duties, on the optimal solution cost over the computational time. This is shown for the same 14,678 task instance as in the previous figure. In this run, the solution cost is mainly related to the number of duties, but it also involved additional terms such as operational costs, and penalties for task over-covering. The value of the Lagrangian lower bound goes down quickly at the very beginning of the computation, when better and better duties are generated, and stabilizes after about 45 minutes, which confirms the effectiveness of the TURNI column generation procedures.

3.5   Second Solution Approach: Iterative Partitioning

As described in the introduction of this chapter, it is potentially more efficient to solve the CSP instance as a whole for the whole week. At the time that TURNI was implemented at NS, there was no available crew scheduling algorithm capable of solving such huge instances at once. A common approach to deal with these huge weekly instances, is to split them into several daily instances and solve those
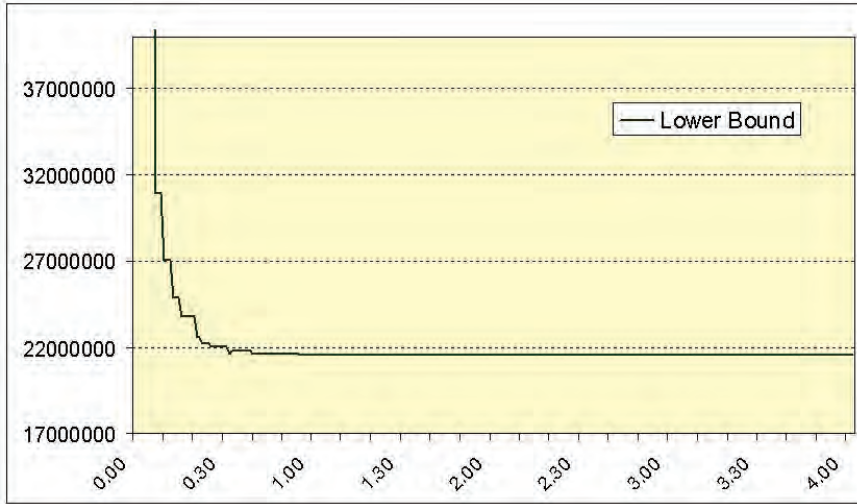
Figure 3.2. Lower bound of a typical instance with TURNI

separately. This was also the case in the initial approach. However, this section will show that this is inefficient.

In this section, we discuss several methods to partition huge instances into several smaller ones. These smaller instances are then solved with the crew scheduling algorithm TURNI. We compare these partitioning methods with each other, and we report several results where we applied different partitioning methods after each other. The results show that all methods significantly improve the solution.

3.5.1   Argumentation

NS, not being the developer, considered the TURNI software to be a black box where data is inserted, and duties are returned. During the initial years of using the software, we determined that although the system was capable of handling quite large instances, the results could be improved using the characteristics of the problem. This is based on two reasonings.

The first reasoning is that the global constraints should be validated on a weekly basis. The original method used a static partitioning of the complete problem into separate days of the week. The tasks can easily be assigned to a single weekday because there are almost no trains running through the night, so there is a natural moment in time to split the problem. Before a solution was computed, an estimate

was made on the effect that the sub-problem would have on the complete problem. For example, the maximum average duration of the duties for a crew depot is 8:00h. Originally, based on many years of planner's experience, this was replaced by a limit of 7:40h on the average duty duration of a working day and a limit of 9:30h in the weekend. In this way, the global average is still approximately 8:00h. However, it is obvious that applying these bounds in a rigid way does not guarantee optimality. We reason that planning for a whole week, and taking into account the real week constraints, will lead to a better overall solution.

The second reasoning is that, in some cases, the solution was improved if, for instance, the solution for one large crew depot was re-scheduled. For this, the duties and tasks for that depot are given to the TURNI software. The solution for this smaller problem is often better than the original solution. This also indicates that solving the larger instances is difficult for this initial implementation. Combining the two reasonings, we posed that we could improve the overall solution if we take the solution for one or more depots for the separate days and combine them into a case for the whole week.

Furthermore, we had the idea that it is good to consider several iterative combinations of depots in order to reduce the negative effect of optimizing over a sub-problem. We are also interested in the effect of varying sizes of the cases. The most important dimensions in scheduling are time and location of the activities. It seems natural to use these dimensions to partition the overall problem.

3.5.2   Partitioning methods

We have selected four methods for partitioning the CSP: (1) Weekday, (2) Geographical, (3) Line, and (4) Column Information based partitioning. In this section, we will describe the four different partitioning methods one by one.

Weekday partitioning

In this method, we create a sub-problem per weekday. All tasks belonging to the same weekday are combined in a sub-problem. For the Friday (as representative for a working day), the Saturday and the Sunday a separate solution is created. The advantage of this method is that it can be used without an initial solution. Because tasks of different weekdays cannot be scheduled together in a single duty at NS, this method is a good option to create an initial solution. In fact, this method was used as the only partitioning method in the initial approach.

Geographical partitioning

The primary geographical partitioning is the depot for which a duty is created and assigned to. After an initial solution has been created, we can combine all duties assigned to a depot for all weekdays. This results in 29 sub-problems, one for each depot. These sub-problems are very small and do not provide significant room for improvement. Therefore, we create some larger cases by clustering some depots based on the geographical location. For this, we split the country into a number of equally sized regions. We create small partitions where, on average, three depots are clustered, and we create large partitions where, on average, seven depots are clustered.

Line based partitioning

The railway product is defined by railway lines. Trains are operated along several railway lines at a certain frequency. For example, consider the 800 line[3] in Figure 3.3.

Figure 3.3. The 800-line with adjacent depots

---

[3]This line was operated in the timetable of 2007, currently the line plan has changed on this route.

This line passes through several depots, which can be grouped into one cluster. We have done this for four important long-distance lines, obtaining four clusters.

Partitioning based on column information

The last partitioning method we present is based on the information that is generated by the scheduling algorithm. As indicated before, TURNI uses a mechanism to rank duties according to their likelihood to be selected in an optimal solution. In this way, good duties are created that have a high probability to be part of the optimal solution. Duties that have no contribution to a good solution are removed from the set, while new duties that have a positive contribution are added. Therefore, the total set of duties is continuously improving. TURNI not only returns the duties that are in the final solution, but it also returns good duties which were generated during the solution process.

These duties give us additional information. If two tasks appear together in many potentially useful duties, it is likely that these two tasks will be assigned to the same duty in the optimal solution. If, on the other hand, two tasks (almost) never appear together in a duty, these tasks will probably be assigned to different duties in the optimal solution. Now, it is possible to give each pair of duties in the current solution a score that can be used as a measure for inserting a pair into a partition. This score is based on how often tasks from these two duties appear together in the set of all duties.

We calculate the score for each pair as follows. First, we count for each combination of tasks in these duties, say t_1 and t_2, the number of duties in the whole set that cover task t_1 and t_2. Then, we add all these numbers. In this way, we can construct a graph $G = (V, E)$, where the duties are represented by the vertices, and the edges represent a positive score. We define a weight $q(u, v)$ for each edge $(u, v) \in E$. This weight corresponds to the score calculated above. We want to find a partition of the vertices of G into k equal subsets $V_1, ..., V_k$, such that the total weight of the edges between different subsets is minimized, or more formally

$$\min \sum_{(u,v) \in E, u \in V_i, v \in V_j, i \neq j} q(u, v). \tag{3.6}$$

We use a generic algorithm for graph partitioning, based on Kernighan and Lin (1970), to solve this problem. For the details, we refer to Van 't Wout (2007).

3.5.3   Results iterative partitioning approach

First, we evaluated the different partitioning methods by running them after a base run in which we used the weekday partitioning. We set up several experiments, and tested the performance of a partitioning method based on the results of the base run. Next to that, we performed experiments in which all methods were applied sequentially. We terminated TURNI, solving a sub-problem, if there was no improvement anymore. All experiments were carried out on the same hardware[4].

The obtained results are based on both optimizing over the week constraints and on iteratively solving smaller sub-problems. To evaluate both aspects, we also performed runs in which we did not optimize over the week, but only applied the method on individual weekday cases. Furthermore, we wanted to evaluate the method for both drivers and conductors, so we made separate experiments for them.

In Tables 3.1-3.4, we present the results of the experiments. The first line corresponds to the different partitioning methods explained in Section 3.5.2 that we applied:

- Weekday partitioning (column "Day"),

- Geographical partitioning with about 7 depots together
  (column "Geo L"),

- Geographical partitioning with about 3 depots together
  (column "Geo S"),

- Line based partitioning (column "Line")

- Partitioning based on column information (column "Info").

The numbers in the columns corresponding to the different methods indicate the order in which the methods are applied. An empty cell indicates that the method was not used in the experiment. In the last two columns, we report the number of duties and the relative improvement compared to the base case. We choose to report the number of duties instead of the objective function, because the value of the objective function is mainly determined by the number of duties.

The first series of experiments, as reported in Table 3.1, deal with a driver's instance (all tasks to be operated by the drivers for a whole week) and a limit on the average duration of 7:40h for the weekday and 9:30h for the Saturday and Sunday.

---

[4]Intel Quad Core, 2.6GHz, 4Gb RAM

Day	Geo L	Geo S	Line	Info	#Duties	Av. Dur.	ΔDuties
1					6072	7:31h	-
1	2				5964	7:37h	-1.8%
1	2	3	4	5	5964	7:45h	-1.8%

Table 3.1. Results of a week instance for drivers with all con-
straints

In these experiments, we have not relaxed any of the global constraints meaning
that we have dealt with a real-world problem. Moreover, in these experiments, we
applied only the partitioning to the individual days. In other words, we have not
optimized over the week constraints. We can see that the partitioning method shows
a significant reduction in the number of duties (1.8%). Applying the sequence of par-
titioning methods does not improve the solution compared with only the geographic
partitioning.

Day	Geo L	Geo S	Line	Info	#Duties	Av. Dur.	ΔDuties
1					6180	7:39h	-
1	2				5969	7:54h	-3.4%
1	2	3	4	5	5944	7:54h	-3.8%

Table 3.2. Results of a week instance for drivers based on
different average durations

The second series of experiments (Table 3.2) deal with the same driver's instance
as in the previous experiments. The differences are that the constraints on variation
and number of duties are relaxed, and that we optimized over the whole week. We
can see, in these experiments, that the initial solution is worse (6180 versus 6072
duties), although several constraints are relaxed. It seems that this is difficult for the
TURNI algorithm. It is obvious that the solutions of the first series of experiments
are also feasible solutions for the second series of experiments. Because the initial
solution is worse, the potential improvement is at least 6180 – 5964 (3.5%), the best
solution found in the first series of experiments. The best solution is even better
than the best solution in the first series of experiments, which could also be expected
because some constraints are relaxed.

The third range of experiments shown in Table 3.3 is again the instance for drivers.
These experiments were similar to the previous ones except that we used a maximum
average duration of 8:00h for all sub-problems.

Day	Geo L	Geo S	Line	Info	#Duties	Av. Dur.	ΔDuties
1					6235	7:43h	-
1	2				6006	7:53h	-3.7%
1		2			6045	7:54h	-3.0%
1			2		6022	7:53h	-3.4%
1				2	6064	7:50h	-2.7%
1	2	3	4	5	5971	7:54h	-4.2%

Table 3.3. Results of a week instance for drivers based on an average duration of 8:00h a day

Compared to the previous range of experiments, we can see that the initial solution is worse because the wrong maximum average duration was chosen. We see that the partitioning method works very well giving an overall efficiency improvement of 4.2% after sequentially applying all methods. We can also see that the second partitioning method has the largest efficiency improvement if we apply it after the weekday instances. Unfortunately, although the relative improvement is larger, we do not get such a good solution as in the previous experiments. It seems that the bad solution of the weekday runs has its effect on the final solution. This indicates that making a good guess (based on the experience of the planners) on the average in the weekday instances has a significant effect.

Day	Geo L	Geo S	Line	Info	#Duties	Av. Dur.	ΔDuties
1					7432	7:49h	-
1	2				7339	7:56h	-1.3%
1		2			7318	7:55h	-1.5%
1			2		7335	7:56h	-1.3%
1				2	7331	7:54h	-1.4%
1	2	3	4	5	7335	7:57h	-1.3%

Table 3.4. Results of a week instance for conductors with all constraints

The last set of experiments, presented in Table 3.4, is about an instance for conductors (all tasks to be operated by the conductors for a whole week). This instance has 18,000 tasks compared to 15,000 for the drivers, because on several trains two conductors are required. In these experiments, we used the averages 7:40h for the weekday and 9:30h for the Saturday and Sunday as used in the planning of the previous years. Next to that, we have applied tight constraints on the number of duties per depot. In this case, we applied the optimization over the week constraints.

We can see that two methods outperform the solution where we apply all methods

sequentially. The best results are obtained by applying the partitioning into small geographically oriented sub-problems. It seems that because the conductor's case is larger than the driver's case, it helps to reduce the size of the sub-problems by using the small geographical partitioning, which creates the smallest sub-problems compared to the other methods.

Overall, we see that all partitioning methods give more or less the same results. For drivers, by applying them all after each other, the largest improvement can be gained. For conductors, it seems sufficient to apply only one of the partitioning methods. The partitioning based on small geographical regions (Geo S) gives good results in the conductors case. The best solution of the iterative partitioning methods was implemented in practice for the crew schedules corresponding to the timetable of the year 2007. Together with some other small improvements in the process, this led to an efficiency improvement of about 2%. For more details on the presented approach, we refer to Van 't Wout (2007).

3.6   Third solution approach: LUCIA

The (second) iterative partitioning approach showed an improvement of the results of TURNI. The method used was a very pragmatic one that could be realized on short notice. The third approach is based on the idea that a new algorithm, based on the state-of-the-art techniques, possibly gives better results than the second approach. NS started a joined project with Siscog [5], the provider of the crew scheduling system called Crews. The project aimed at developing an optimization model capable of solving from scratch large-scale duty scheduling problems, with results that would be comparable to the results presented in the second approach. The result of this project is an algorithm called LUCIA. This section is based on the paper Abbink et al. (2011).

3.6.1   LUCIA solution method

In this section, we describe the LUCIA algorithm that was developed to solve the CSP. LUCIA has been fully integrated in the CREWS package developed by SISCOG. The algorithm uses dedicated networks where nodes correspond to tasks, and arcs to connections between tasks. A path in such a network corresponds to a duty. The CSP is solved by a Lagrangian-based heuristic combined with column generation (see

---

[5] www.siscog.pt

Section 3.3), similar to the techniques used in TURNI. Because we were involved in the development of the algorithm, we can discuss some of the details of the algorithm.

There are two significant differences with TURNI. The first difference is that the system was designed to take advantage of the modern CPU architectures with multiple cores. The pricing problems (generation of the duties) per depot and parts of the master problem can be processed in parallel. The second difference is the applied concept of connections, which will be explained in this section.

Network of connections

First, we explain how we construct the networks, where paths represent all possible duties. We construct such a network for each crew base, taking the route and the rolling stock knowledge of a crew base directly into account. Since we are planning duties for a whole week, a task in this context refers to an activity that the crew member has to perform on a particular weekday. This means that in this network: (i) a train that runs from Monday to Friday corresponds to five nodes, one per weekday; (ii) the starting time of a task is the time elapsed between 0:00 of the first day of the week and 0:00 of the day where the task starts plus the clock time where the task starts. This also means that the network contains all tasks operated in a standard week.

Arcs in this network connect either: (i) nodes on the same weekday, (ii) nodes at the end of a weekday to nodes at the beginning of the next weekday, and (iii) nodes at the end of the week to nodes at the beginning of the week. The length of an arc is limited, as will be explained later. The network is typically continuous, because the subnetworks of consecutive weekdays may be connected to each other. The network is also cyclic, because it can contain cycles that take one or more weeks to complete.

The length of an arc in this network is limited. More precisely, two tasks i and j can only be connected by an arc if the time between them satisfies the following upper bound:

$$[(taskStartTime_j - taskEndTime_i) \bmod weekTime] \leq maxArcLength, \qquad (3.7)$$

where $taskStartTime_j$ is the starting time of task j, $taskEndTime_i$ is the ending time of task i, mod is the modulo operator, $weekTime$ is a constant representing the time elapsed in one entire week, and $maxArcLength$ is a parameter representing the maximum length allowed for an arc, which must always be smaller than the maximum duty length.

To check whether an arc between two nodes exists and is useful, we solve a shortest path problem in another network, which contains many more details. This is necessary because the minimal transfer time between two trains differs when a crew member is a passenger or is working on a certain train. In this second network, the nodes correspond to *tasks*. Here, a task is defined as the movement of a train between a pair of stations. There are arcs between two nodes if two tasks can be assigned to the same crew member and when there is enough transfer time between two tasks. Figure 3.4 shows a graph with five nodes and the corresponding arcs. A solution can be made with three duties (1,2,3), (4), and (5).
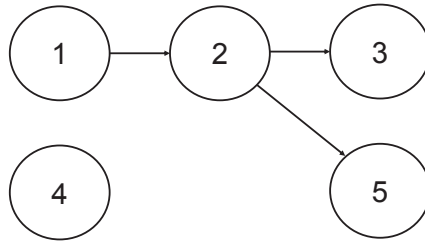
Figure 3.4. Basic Graph

In addition to this basic network, we introduce a copy of each node that represents the same task, but it is used as a positioning task (with the shorter transfer time). We illustrate this with an example (see Figure 3.5). In this example, six nodes are
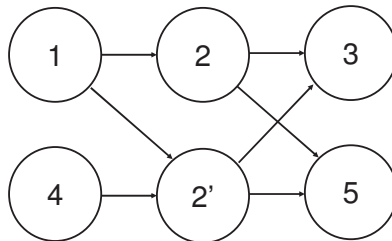
Figure 3.5. Graph with optional node

depicted. There is not enough transfer time between node 4 and node 2, so there is no connecting arc. Node $2'$ is added as an optional task, and there is an arc between node 4 and $2'$. By adding the optional task, the solution can be made with two duties: (1, 2, 3) and (4, $2'$, 5), due to the shorter transfer time for optional tasks. Without the optional task, it would be required to generate an additional duty. A drawback of this formulation is that the size of the problem increases significantly when the number of additional optional tasks is large.

A simple, yet very efficient alternative solution is depicted in Figure 3.6. We add
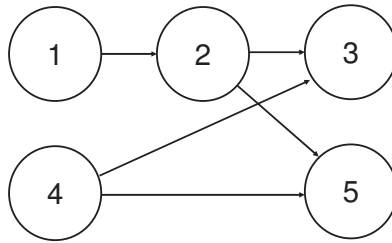
Figure 3.6. Graph with additional arc

a (positioning) arc between node 4 and node 5 (and between node 4 and 3) because we know that there is a possibility to get from the end location of node 4 to the start location of node 5 by using a positioning task. Note that we could also add an arc between node 1 and node 3, but due to the mechanism of allowing node 2 to be over-covered, this arc would be redundant.

This idea can also be found in the work of Rezanova and Ryan (2010). A special preprocessing procedure determines when such additional arcs are to be added. In theory, the process of generating these additional positioning arcs can be quite complex if we would consider all possible arcs. Arcs can be based on multiple task nodes and on other modes of transport like taxi and bus. We only need to add arcs between nodes with a limited time interval in between, namely two hours, due to the high frequency of train services. When the interval is large, and the distance is small, it will be possible to connect the nodes using the regular network. When the distance is large, there is not much use in generating connecting arcs because a long distance positioning task is not efficient. This limits the sets of nodes for which we try to generate the additional arcs.

For each set of nodes, we first try to find an arc with a meal break opportunity. Only if this is not found, we try to find an arc without a meal break opportunity. In order to further reduce the number of additional arcs, we also set the maximum number of rolling stock changes, in arcs based on multiple task nodes, to one. Adding arcs instead of nodes is computationally efficient and solves the problem of formulating differentiated transfer times.

Lagrangian heuristic combined with column generation

The main characteristics of the algorithm are a dynamic pricing scheme for the variables, coupled with subgradient optimization and greedy algorithms, and the systematic use of column fixing to obtain improved solutions. To tackle the large number of potential duties, LUCIA uses column generation to generate feasible duties ("columns"). These duties are generated by solving a resource constrained shortest path problem in a network of connections (see Section 3.6.1).

At the top level, as shown in Table 3.5, the algorithm obtains solutions based on a given network of connections by executing a loop (line 4), that alternates between column generation and optimization, that is fed by the columns stored in the pool *cols*, and the set Λ of Lagrange multipliers (line 3) related with the constraints (3.2) and (3.3).

```
1 algorithm SOLVE (network) returns sols
2   begin
3     (cols, Λ) ← GENERATEINITIALSETOFCOLUMNS(network)
4       return GENERATEANDOPTIMIZE(network, cols, Λ)
5   end
```

Table 3.5. The main algorithm

The generation of the initial set of columns is shown in Table 3.6. The algorithm starts by generating for each task in the network a column representing an infeasible duty that covers that task (line 3). These duties have a very high cost so that they are included in the solution only if there is no feasible way of covering the task. Next to that, the algorithm enters an iterative process where several steps are executed.

```
1  algorithm GENERATEINITIALSETOFCOLUMNS (network) returns(cols, Λ)
2     begin
3        cols ← GENERATESLACKVARIABLECOLUMNS(TASKS(network))
4        Λ ← (0, 0, . . . , 0)
5        while  Number of iterations without improving LBOUND(Λ, cols)
                         smaller or equal than 5 and
6                   Total number of iterations smaller or equal than 100 do
7           begin
8              cols ← cols \ CHOOSECOLUMNSTODELETE(cols, Λ)
9              Λ ← SOLVELAGRANGEANRELAXATION(Λ, cols)
10             aggregLevel ← 100% − (percentage of iterations done so far)
11             cols ← cols ∪ GENERATECOLUMNS(network, ∅, Λ, aggregLevel)
12          end
13       return(cols, Λ)
14    end
```

Table 3.6. Algorithm that generates the initial set of columns

First (line 8), uninteresting columns are removed from the column pool *cols*. Uninteresting columns are identified by having reduced cost larger than the maximum variable cost, i.e., $\max\{c'_{sj} \,|\, s \in S, j \in J_s\}$, that can be found in a potential duty. Then (line 9), the Lagrangian multipliers Λ are updated by solving the Lagrangian relaxation problem as in Caprara et al. (1999), but in an extended way, i.e., with the capability of handling additional constraints. Finally (line 11), the columns are generated (i.e., the pricing problem is solved) for the updated multipliers. Later in this chapter we will explain how this works.

In order to speed up the column generation process, tasks that share the same rolling stock are aggregated in a decreasing degree of strength during the iterative process. Aggregating a set of tasks means considering them (temporarily) as just one task, so that they all must be assigned to the same duty. By doing so, the number of nodes in the network reduces, as well as the number of arcs (up to 75%), and, therefore, the pricing problems become easier and faster to solve.

The loop that alternates between column generation and optimization is the most important part of the algorithm and is shown in Table 3.7. In the main loop, that terminates when the gap between the lower and upper bound becomes small enough, or when a set number of iterations is reached (line 5). After updating the Lagrangian multipliers in line 8, the optimization problem is solved for the current set of columns and multipliers (line 9) to obtain solutions. The optimization problem is solved with a heuristic that extends the Lagrangian-based heuristic described in Caprara et al.

```
1  algorithm GenerateAndoptimize (network, cols, Λ) returns sols
2     begin
3        sols ← ∅
4        fixedCols ← ∅
5        while uBound(sols) − lBound(Λ, cols) ≥ 1 and
6                 Total number of iterations smaller or equal than 300
7           begin
8              Λ ← SolveLagrangeanRelaxation(Λ, cols)
9              (Λ, sols) ← SolveoptimizationProblem(cols, Λ, fixedCols, sols)
10             fixedCols ← ChooseColumnsToFix(cols, Λ)
11             cols ← cols \ ChooseColumnsToDelete(cols, fixedCols, Λ)
12             while Number of sub-iterations smaller or equal than 10 do
13                begin
14                   cols' ← GenerateColumns(network, fixedCols, Λ, 0%)
15                   cols ← cols ∪ cols'
16                   Λ ← SolveLagrangeanRelaxation(Λ, cols)
17                   if |cols'| < 5 then exit while
18                end
19          end
20       return sols
21    end
```

Table 3.7. Algorithm that combines generation and optimiza-
tion phases

(1999), with the capability of handling additional constraints.

Before switching to the column generation phase, the most interesting columns
fixedCols are fixed (line 10), to reduce the complexity of the problem, and uninterest-
ing columns are removed from the column pool *cols* (line 11). The size of *fixedCols* is
at most the number of tasks divided by 200, and contains, among the columns with
reduced cost smaller than 0.1, the ones that have the smallest reduced cost. The
uninteresting columns are obtained exactly in the same way as in the generation of
the initial set of columns. The column generation (line 14) is executed several times
in a secondary loop where the multipliers are updated (line 16) to reflect the new
columns that were added to the container.

Column generation

The generation of columns (i.e., the resolution of the pricing problem) is done by an
algorithm that uses, as a starting point, the solution of the shortest path problem
over the network of connections. As in Huisman (2007), the algorithm looks for a

number of variations over the shortest path (including the shortest path itself), and it collects the ones that have negative reduced cost and are feasible. A path is feasible if it satisfies some additional constraints such as the meal break constraint.

The fact that the network is cyclic poses a problem to the shortest path algorithm. However, this problem is easily handled by solving the pricing problem over different (acyclic) parts of the network (subnetworks), instead of doing it over the entire network. Splitting the pricing problem in parts, solves the cyclic network problem, and opens a door for running the algorithm in parallel. A natural way of dividing the network in parts is by weekday. This means that each of the seven subnetworks is associated to a particular weekday and that the columns generated inside it correspond to all duties that start on that weekday (and therefore belong to that weekday).

Following these guidelines, we defined the subnetwork of weekday j as the part of the overall network containing all sign on and sign off nodes and the tasks (nodes) that fit inside the following time intervals:

$$\left[dayStart_j, \ dayEnd_j + maxDutyLength\right], \ \text{if } j = 1, ..., 6 \tag{3.8}$$

$$\left[dayStart_1, \ maxDutyLength\right] \cup \left[dayStart_7, \ dayEnd_7\right], \ \text{if } j = 7 \tag{3.9}$$

where: $dayStart_j$ is the time elapsed between the start of the week and the time where weekday j starts, $dayEnd_j$ is the time elapsed between the start of the week and the time where weekday j ends, and $maxDutyLength$ is the maximum time length allowed for a duty. From this definition, we can conclude that the subnetwork of a weekday overlaps with the subnetwork of the next weekday, in a time length that is equal to $maxDutyLength$. These overlapping areas contain tasks that belong to both weekdays (can be covered by duties of both weekdays). This way of splitting the pricing problem in parts, with overlapping areas, ensures that all possible duties are included, and there is no loss of optimality. By looking at the definition, it is also possible to confirm that the subnetworks are acyclic, as required beforehand.

Solving the pricing problem is usually a very time consuming process, due to the size of the network. In order to obtain very significant performance improvements, we run the algorithm in parallel. The number of parallel threads depends on the number of cores available. If there are enough cores, there can be a thread for each weekday and crew base. In this case, each thread would run the pricing algorithm over a subnetwork corresponding to a given weekday with a sign in and sign off node corresponding to a given crew base. As will be shown in the next section, with this improvement, we are able to compute a solution for the largest instance within 30

hours. Without running the algorithm in parallel, it would take much more time, which is not desired (see Section 2.4.3).

3.6.2   Computational Results LUCIA

In this section, we illustrate the performance of LUCIA on the real-life cases of NS. First, we describe the effect on the computation time of applying parallel processing in the algorithm. Second, we compare the results of LUCIA with the previous approaches. Finally, we discuss the results of applying the algorithm on the week instances of the CSP.

Effect of Parallel Computing

The algorithm was designed to take advantage of the multi-core architecture of modern CPUs. To show the effect of using multiple cores, we have performed three sets of tests on one, two, and four of the available CPU-cores.
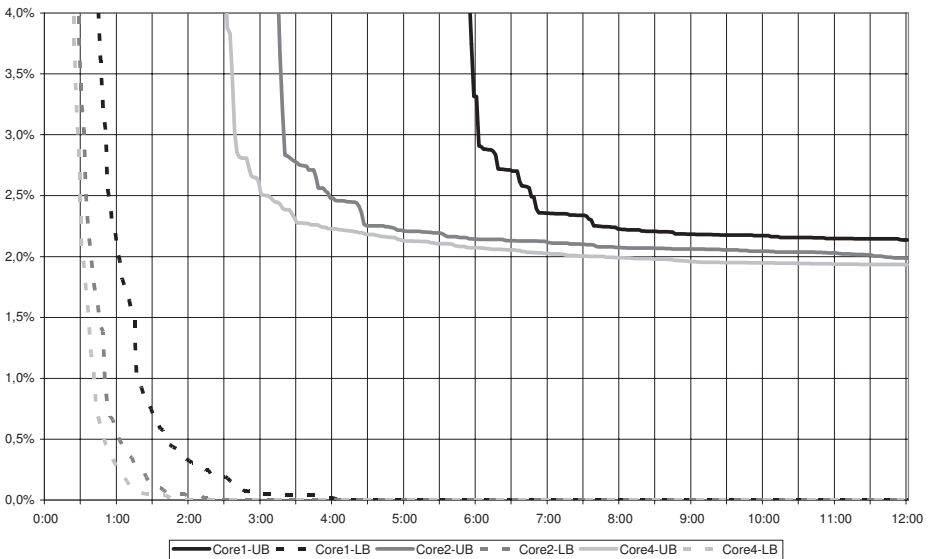
Figure 3.7. Effect of Parallel Computing

Figure 3.7 plots average current best lower and upper bounds (in percentage of the best lower bound) over the computation time[6] for the test instances. The dotted lines show the decrease of the Lagrangian lower bound on the optimal solution costs. This lower bound is dependent on the set of generated columns and, therefore, it is not exact.

In the first phase, the algorithm generates a large amount of columns. With one CPU-core, this takes about four hours. After this, the algorithm applies its heuristic, to find a feasible solution, and within 24 hours, the algorithm has found a feasible solution, with an upper bound (the solution value) which is within 2% of the best lower bound found.

The figure shows that it requires 11:32h to get the same result with two CPU-cores. Using four CPU-cores, the algorithm reaches a gap of less than 2% within 7:26h of computation time. The graph shows that, after seven hours, very good solutions are generated with the four CPU-cores. From that moment, the average gap is only 2%, and the solution does not significantly improve anymore.

When we compare the three different runs, we can see that doubling the processor capacity speeds-up the solution process with 50 percent. Quadrupling the capacity speeds-up the solution process with 68 percent. This large improvement can be explained by the fact that a large part of the algorithm is parallelized. There is a limited contribution of adding additional CPU-cores, because some parts of the algorithm cannot be split. When these parts are computed, only one single CPU-core is active, and the others are idle. For the LUCIA algorithm and the instances of NS, it seems sufficient to use four CPU-Cores per instance. The results show that the algorithm is capable of generating solutions within short computational times.

Comparison with Previous Approaches

Next to the computation time discussed in the previous section, we will discuss the quality of the solutions. To evaluate if the LUCIA solutions are at least as good as the approach described in the previous section, we created ten benchmark cases as shown in Table 3.8.

We want to evaluate the method for both drivers and conductors, so we make separate cases for them. The column Crew Type indicates drivers ("D"), conductors ("C"), or both ("D + C"). The column "Case Size" indicates the size of the problem.

100% means that all tasks to be assigned to the drivers or conductors are included. 50% (drivers) indicates that about half of the set of tasks is included. We construct

---

[6]hours on a PC Xeon Quad Core, 4Gb RAM and a clock speed of 2.6 GHz

	Crew Type	Case Size	Day	Transfer Time	Meal Break Time
BC1	D	50%	Thursday	0:20h	0:30h
BC2	D	100%	Thursday	0:20h	0:30h
BC3	D	100%	Thursday	0:15h	0:30h
BC4	C	100%	Thursday	0:20h	0:30h
BC5	C	100%	Thursday	0:15h	0:30h
BC6	C	70%	Thursday	0:15h	0:30h
BC7	C	30%	Thursday	0:15h	0:30h
BC8	C	100%	Sunday	0:15h	0:30h
BC9	C	70%	Sunday	0:15h	0:30h
BC10	C	30%	Sunday	0:15h	0:30h

Table 3.8. Benchmark cases

the case by taking all tasks that run along a part of the network. "70% of the conductors" relates to only including one (first or head) conductor per train. The other 30% are the tasks related to the additional conductors (second or third) per train. NS can decide to plan them separately, such that these tasks can be assigned to personnel with lower qualifications, e.g., so called service personnel. When a duty contains at least one task for a first conductor, it should be operated by a fully qualified conductor. Planning them separately is less efficient, but can be useful when there is a shortage of fully qualified conductors.

We also varied in the day of the week, minimum duration of the transfer time and of the meal break. All cases were constructed with global labor rule constraints and variation constraints.

All experiments were carried out on the same hardware. Table 3.9 shows the results for the benchmark cases. The column "TURNI" shows the number of duties

	# TURNI	# IIP	# LUCIA	Δ
BC1	439	436	437	1
BC2	837	830	822	-8
BC3	833	825	824	-1
BC6	850	835	832	-3
BC7	315	315	314	-1
BC8	760	747	739	-8
BC9	616[7]	616	615	-1
BC10	141	141	138	-3
Total	4791	4745	4721	-24

Table 3.9. Results of the benchmark cases

generated by TURNI, after a 24h of computation time. In fact, four parallel runs were performed, each with some minor adjustments in the internal solution strategy. The results of the best run are presented. The column "IIP" shows the results of applying the iterative improvement procedure as described in Section 3.5. This procedure started with the best solution, presented in the "TURNI" column, and was given 24h of computation time to improve the solution. The column "LUCIA" shows the result of applying the new algorithm, with a computation time of 48h. Note that this computation time is much longer than practically used at NS. The focus of the benchmark was on the quality of the solutions and we want to be sure that convergence is reached by the algorithms. The last column shows the improvement that is achieved by applying LUCIA.

The results of the benchmark cases BC4 and BC5 are not in the table because TURNI could not find a solution without constraint violations. This would make it unfair to compare them. LUCIA did find a feasible solution for all the benchmark cases. Overall, we conclude that we reach an improvement of at least 0.5% (24 duties) on the benchmark cases.

Results Week Instances

In this section, we present the improvement we reach by solving the CSP for a whole week. As shown in the previous section, the results of the LUCIA algorithm already had a high quality for the individual weekday instances. In the presented second approach, we already addressed the issue of the global (week) constraints. However, in this second approach the problem was not solved as a whole, as we present here.

To compare the results of optimizing the CSP instances per weekday with the results of optimizing the CSP as a single instance, we have created two cases: (i) all tasks for the drivers and (ii) all tasks for the conductors. With these two cases we vary the number of applied constraints. All experiments are carried out on the same hardware[8]. For each experiment addressing a single day instance, we let the algorithm run for 30 hours. After these 30 hours, we see that convergence has been reached and that the solution does not improve anymore. For the experiments addressing the week instances, we let the algorithm run for 7*30 hours to give it the same computation time as the separate runs. In all cases the best solution was found before half of the available computation time has been used. After this, no improvement was found by the algorithm.

---

[8]Intel Octo Core (2x4 Cores), 3.0GHz, 8Gb RAM

First, we made an experiment without global constraints. In the second experiment, we added the constraints on the duration (D) of the duties. To be specific, the constraint of 8:00 hours on the maximum average duration of the duties and the constraint saying that no more than 5% of the duties can have a length of more than 9:00 hours. In a third experiment, we added the constraint on having a minimum of 40% and a maximum of 70% of work on preferred trains (PT) per crew depot. As explained in Section 3.7, incorporating a minimum and maximum percentage we can satisfy the constraint on the standard deviation. Also, notice that when we solve the single day instances, all global constraints should be satisfied for each individual day. In this way, the constraints are per definition satisfied for the whole week as well. We have created these three experiment instances for both drivers and conductors.

Tables 3.10 and 3.11 show the results for each of the instances. The first two columns indicate the type of the problem. The third column shows the results for the individual day cases, and the fourth column shows the results for the week instances.

Case	Constraints	Indiv. Days	Week	Delta	Relative
D 1	none	22907166	22962526	55360	0.2%
D 2	D	23079190	22882933	−196257	−0.9%
D 3	D & PT	23403709	23097736	−305973	−1.3%
C 1	none	27608478	27738766	130288	0.5%
C 2	D	27863687	27790539	−73148	−0.3%
C 3	D & PT	28229258	28094205	−135053	−0.5%
C+D 1	none	50515644	50701292	185648	0.4%
C+D 2	D	50942877	50673472	−269405	−0.5%
C+D 3	D & PT	51632967	51191941	−441026	−0.9%

Table 3.10. Results Week Optimization, Costs

Case	Constraints	Indiv. Days	Week	Delta	Relative
D 1	none	5198	5221	23	0.4%
D 2	D	5393	5324	−69	−1.3%
D 3	D & PT	5489	5400	−89	−1.6%
C 1	none	6388	6428	40	0.6%
C 2	D	6588	6552	−36	−0.5%
C 3	D & PT	6715	6636	−79	−1.2%
C+D 1	none	11586	11649	63	0.5%
C+D 2	D	11984	11876	−105	−0.9%
C+D 3	D & PT	12204	12036	−168	−1.4%

Table 3.11. Results Week Optimization, Number of Duties

Table 3.10 shows the results in terms of the value of the cost function and 3.11 shows the results in terms of the number of generated duties.

The first set of experiments shows that solving the whole week as a single instance gives a worse result. In theory, the solution can be better because the solver can decide to plan a task in a duty of a different day than the day it was assigned to in the runs for the individual days. However, the experiments show that the increased size of the problem makes the problem harder to solve which results in a worse solution. The negative effect is larger for the conductors, which can be explained by the fact that these instances are larger than the drivers instances, and more difficult to solve. The other experiments show that solving the whole week as a single instance gives a better result than solving the single weekday instances. The positive effect for the conductor's instances is somewhat smaller than for the drivers.

When comparing the different sets of experiments, it is shown that adding constraints (like duration and preferred trains) leads to a more expensive solution in terms of costs and number of duties. For the week instances, there is an increase of 387 duties between the first and third experiment. This is about three percent. For the individual day instances, this increase is equal to 618 generated duties (about five percent).

The positive effect of solving the problem as a single instance, can be explained by the fact that in the week instances, the algorithm has the possibility to have a larger average duration for the duties in the weekend, as long as it compensates this with shorter duties during the weekdays.

It is not shown in the table, but when we inspect the solutions in more detail, this is exactly what happens for some crew depots. For other crew depots, the system decides to have longer duties during the weekdays and to have shorter duties during the weekend. For the total set of crew depots, we see that the average duration gets longer in the weekend and also the related percentage of long duties (with a duration of more than 9:00 hours), is higher than during weekdays.

Having short duties during the weekdays is efficient because, in railway operations, there is generally a peak–hour pattern for the commuters. During these peak-hours, there is a higher need for crew than during the rest of the day and this need can be covered efficiently with short duties. In the weekend, there is no peak-hour pattern, and it is efficient to create long duties. The positive effect of this is the fact that the number of duties during the weekend is reduced. Crew members in general like to have the weekend off, so for them, to operate relatively fewer duties during the weekend is attractive.

Table 3.12 shows the number of duties per day for the third set of experiments, with constraints on duration and preferred trains. It shows that the efficiency gain is not only achieved on Saturday and Sunday, but also on the other days of the week. We see that, for two days the solution is worse, in number of duties.

Case	Type	Mon	Tue	Wed	Thu	Fri	Sat	Sun
D 3	Individual	816	813	805	821	838	736	660
	Week	802	796	807	815	835	709	636
	Delta	−14	−17	2	−6	−3	-27	-24
C 3	Individual	1047	1046	1045	1046	1038	797	696
	Week	1032	1040	1036	1054	1030	769	675
	Delta	−15	−6	−9	8	−8	−28	−21

Table 3.12. Results Per Individual Day

Overall, the results show that the gain in solving the week as a single instance gets larger when there are more global constraints that apply for the whole week. Even though the problem has more global constraints, in some cases, the algorithm is still able to find better solutions. With the presented approach, we are able to solve the whole week instance with global constraints that are to be applied in the NS case. The overall effect is that we gain an efficiency increase of about 1%.

3.7   Practical Details

This section elaborates on a few specific issues that illustrate the complexity of implementing a crew scheduling algorithm in practice. The issues, that we present, relate to the Sharing-Sweet-and-Sour rules (see 2.4.3), and address the issue of generating attractive duties for the personnel. The examples and the results are created with the LUCIA algorithm introduced in Section 3.6

Route Variation

Driving different routes is considered to be attractive by the drivers. At NS, there is a labor rule on the minimum number of different routes per depot. This rule implies that we need to find a mechanism to get some depots on routes that they would not operate otherwise. For example, the aim is to get depot Emmen[9] on the route Utrecht-Rhenen. Without guiding the system, it generated solutions in which this depot passes Utrecht, but no tasks towards Rhenen are included. Another

---

[9]Currently this depot is not operated by NS anymore.

variation rule is the minimum percentage of preferred work per depot. Because depot Utrecht has a large amount of non-preferred work, it helps when another depot operates on non-preferred work starting in Utrecht. On the route Utrecht-Rhenen, the corresponding tasks are categorized as non-preferred work. Assigning depot Emmen to work on this route gives them an additional route and reduces the amount of non-preferred work of depot Utrecht.

A first solution is to select specific tasks on this route and tell the system that only Emmen has the required knowledge to operate these tasks. This works fine when the number of these requirements is small. Another option is to add a constraint. For example, let K be the set of depots and R be the set of routes. If k_d and r_d denote the crew depot and the existence of route r in duty d, respectively, and V_{kr} denotes the minimum number of duties containing route r, for each crew depot, then the following constraints guarantee that the minimum number of duties containing required routes for each crew depot is respected:

$$\sum_{d:k_d=k} r_d x_d \geq V_{kr} \qquad \forall k \in K, r \in R \tag{3.10}$$

Recall that x_d indicates whether duty d is included in the solution. The advantage of adding these constraints is that the algorithm has the possibility to select the tasks that are best for the overall solution. This is illustrated in Table 3.13. The

Case description	Cost	#Duties	Increase
1: None	38448930	917	0%
2a: Task Assignment	38633970	917	0.5%
2b: Route Constraints	39038637	925	1.5%
3a: Task Assignment	39061230	925	1.6%
3b: Route Constraints	39146680	932	1.8%

Table 3.13. Approaches for Route Assignment

first column describes the experiment instance. Case 1 is the instance without task assignment or route variation constraints. Case 2 is the instance with about 150 routes that need to be assigned to a certain depot, either by a manually created fixed task assignment, or by an additional constraint. In Case 3, we create an instance for about 300 routes that need to be assigned to a depot. The second column shows the solution costs, the third column shows the number of generated duties, and the fourth column shows the increase in solution costs in relation to Case 1.

The experiments show that using fixed task assignments gives better results than

adding additional constraints. Although the second method has a larger solution space and, therefore, theoretically can give better solutions than the first method, the algorithm does not find this better solution.

The results show that when the number of constraints increases, the relative difference between the two methods decreases. Although the number of generated duties in the third case is much higher, compare for instance 3b in relation to instance 3a, the difference in the cost function is not so large. This can be explained by the fact that the average duration of the duties is somewhat larger (3 minutes) in instance 3a and that there are more additional costs. For example, these additional costs can be costs related to taxi tasks. This indicates that the tasks with fixed assignment can be covered, but that the corresponding duties have a relatively large duration and that some additional taxis are needed to cover the tasks.

In practice, creating the fixed task assignments is a time consuming task and it is preferred to use the method of adding constraints. The experiments indicate that there is still some improvement possible in the algorithm in handling the additional constraints.

Standard Deviation on Aggression

One of the Sharing-Sweet-and-Sour rules is that the computed standard deviation on the percentage of aggression work should not be higher than 11.5. This is not a linear constraint, and, therefore, it is difficult to explicitly deal with it in the optimization model. We choose to use a work-around on this issue.

The standard deviation σ can be computed using the formulation in equations (3.11–3.13).

$$\sigma = \sqrt{\frac{\sum_{k \in K} v_k}{D}} \tag{3.11}$$

$$D = \sum_{k \in K} n_k \tag{3.12}$$

$$v_k = n_k(a_k - \bar{a})^2, \quad \forall k \in K \tag{3.13}$$

Formula (3.11) computes the weighted standard deviation. In formula (3.12), the total number of duties D is computed. v_k is the contribution of depot $k \in K$ to the value of the standard deviation, where a_k is the percentage of aggression work of the depot. The average percentage of aggression work $\bar{a}$ can be considered a constant because the amount of aggression work is given as input.

When a solution is generated by the algorithm, we compute the corresponding

standard deviation σ. If this is too high, we look at the contribution v_k of the depot to this value. For the depots that have a high contribution, by a high (or low) value of a_k and a high number of duties n_k, we change the maximum (or minimum) percentage of aggression work per depot u_k (or l_k) in the algorithm. Note that the default value of the maximum percentage of aggression per depot is 50%, based on the corresponding labor rule.

We illustrate this using the case BC5 as introduced in Section 3.6.2. In a first experiment, we ran this case with all values for the upper bound on the aggression constraint per depot set to 50%. The lower bound is set to 0% for all bases, and is not included in the table. The results are shown in Table 3.14. The first four columns (after the column with the depots), show the results of an initial run. The last row of the column v_k shows the value of the standard deviation (12.1), which exceeds the maximum labor rule value (11.5). We see that three depots (Amr, Asd and Lls) have a large contribution to the standard deviation because they have a relatively high value v_k and a relatively large amount of duties.

The columns u_k^1 to v_k^1, show the results of a second run, with adjusted values of the upper bounds on aggression work. We lowered the maximum value for the three depots (Amr, Asd and Lls) to respectively 35%, 25% and 43%. The contribution of Lls is the highest, so we subtract about 5% of the realized aggression value to obtain a new upper bound. For the depots Amr and Asd, where the contribution is smaller, we subtract about 2.5% of the realized aggression value to obtain a new upper bound.

Without running the new instance, we can compute if the standard deviation is valid when all contributions remain the same, except for the depots with a new upper bound. For these depots, we use the upper bound value in the computation of the standard deviation. If the result is feasible, we run the new instance. Note that there is no guarantee of finding a feasible solution because the aggression values of the other depots will be changed as well. Part of the aggression work assigned to the three depots mentioned will have to be reassigned to other depots.

In this run, the value of the standard deviation is allowed (smaller than the maximum value of 11.5). One can see that the aggression work has been reassigned to neighboring depots. For example, Hfdo (geographically close to Asd) has a much higher percentage of aggression work. The number of duties is similar to the number of duties in the original run, which implies that the constraint on the standard deviation is not so tight.

Another option would be to set the lower bound L and upper bound U to an equal value for all depots. We have performed two experiments with this method.

Depot	u_k	n_k	a_k	v_k		u_k^1	n_k^1	a_k^1	v_k^1
Ah	50%	43	6.15	3553		50%	43	10.97	738
Amf	50%	41	18.58	457		50%	41	18.31	419
Amr	50%	39	37.26	**18911**		**35%**	38	33.3	**12568**
Asd	50%	91	27.62	**13948**		**25%**	92	24.01	**7281**
Ddr	50%	20	15.06	1		50%	20	15.46	2
Ehv	50%	63	6.54	4768		50%	64	7.58	3633
Ekz	50%	16	15.2	0		50%	16	15.98	12
Emn	50%	10	0	2322		50%	10	0	2284
Es	50%	27	3.69	3602		50%	27	3.67	3536
Gn	50%	30	8.15	1508		50%	31	8.58	1323
Gvc	50%	93	9.48	3085		50%	90	9.54	2796
Hdr	50%	21	33.19	6767		50%	21	36.39	9506
Hfdo	50%	27	29.52	5506		50%	27	38.71	15033
Hgl	50%	21	3.19	3049		50%	21	1.09	4130
Hlm	50%	21	27.12	2964		50%	24	28.41	4243
Hn	50%	15	36.69	6902		50%	15	37.09	7244
Hrl	50%	21	7.42	1284		50%	24	5.47	2232
Ht	50%	17	6.46	1310		50%	17	9.39	557
Lls	50%	48	48.53	**53196**		**43%**	48	42.9	**37059**
Lw	50%	23	2.07	3989		50%	22	3.61	2911
Mt	50%	31	5.81	2756		50%	30	4.57	3335
Nm	50%	39	11.52	540		50%	40	11.44	540
Rsd	50%	44	5.35	4303		50%	42	6.41	3182
Rtd	50%	77	16.42	107		50%	77	18.03	655
Ut	50%	105	13.42	348		50%	105	11.88	1098
Vl	50%	22	1.09	4405		50%	20	2.29	3289
Vs	50%	13	16.09	9		50%	13	14.83	1
Zl	50%	45	4	5685		50%	45	3.18	6409
Zp	50%	19	2.07	3295		50%	19	0.08	4294
Total		1082	15.2	**12.1**		variable	1083	15.2	**11.4**

Table 3.14. Example Standard Deviation

In Experiment 3, we set the values to $L = 5$ and upper bound $U = 45$, and in Experiment 4, we set the values to $L = 8$ and upper bound $U = 40$. Table 3.15

Experiment	L	U	D	s
1	0%	50%	1082	12.1
2	0%	variable	1083	11.4
3	5%	45%	1076	11.1
4	8%	40%	1089	9.8

Table 3.15. Overview Experiments on Aggression Work

shows the results of the experiments.

The experiments show that a correct value of the standard deviation is found by setting the lower and upper bound to a value that is larger respectively smaller than required on a depot level. Experiment 4 shows that setting these constraints too tight gives a more expensive solution than can be achieved by some fine-tuning. When there are significant changes in the timetable, this fine-tuning can be easily repeated.

Standard Deviation on Preferred Trains

The constraint on the maximum value (10.5) of the standard deviation on the preferred trains can be handled in a similar way as the aggression work. We have created four experiments based on the drivers instance BC3. Experiment 1 is the base run with the lower bound set to the labor rule value of 25% per depot. In Experiment 2, we have increased the value of the lower bound, or lowered the upper bound, for the depots with the highest contribution to the standard deviation. Experiment 3 shows the result of increasing the lower bound to 35% per depot, and setting the upper bound to 75%. Experiments 4 (and 5) show the result of increasing the lower bound to 42% (40%) per depot, and setting the upper bound to 72% (70%). Table 3.16 shows the results of the experiments.

Experiment	L	U	D	s
1	25%	100%	822	13.2
2	variable	variable	826	10.5
3	35%	75%	821	11.1
4	42%	72%	822	9.6
5	40%	70%	829	9.3

Table 3.16. Overview Experiments on Preferred Trains

The first experiment shows that the standard deviation on the percentage of preferred trains per depot is too high when the lower bound is set to 25%, the required minimum per depot. In experiment 2, we have fine-tuned the lower and upper bounds, and get a valid standard deviation and a solution with 826 duties. In experiment 3, we have increased all lower bounds to 35% and lowered the upper bounds to 74%. This gives an improvement of the standard deviation, but not enough to satisfy the labor rule. In experiment 4, we have set the lower bounds to 42% and the upper bound to 72%. With this experiment, we get a valid standard deviation and a solution with 822 duties. The results of experiment 5 show that setting the lower bound to 40% and the upper bound to 70% is too tight because there are more duties generated than in experiment 4 which also gives a valid value for the standard deviation. Because the lower bound is less restrictive than in experiment 4, the upper bound seems to be too tight.

Overall, the method of experiment 4 gives the best result. Next to the low number of duties, an attractive aspect of this method is that the preferred work is spread evenly over the depots, which is in line with the philosophy behind the rule.

3.8   Conclusions

In Section 3.1, we addressed the research question 3: *"What are commonly used methods for supporting the Crew Management process?"* In particular, we have answered the question for the duty scheduling process. We have formulated the problem using set-covering models and have described commonly used mathematical programming techniques for solving the problem, in particular dynamic column generation.

In this chapter, we have shown that these techniques can be applied to the research problem, although this is certainly not trivial. During the years, we have improved the application of techniques by adding an iterative heuristic to improve the solution of the algorithm and we have found a good way to handle positioning tasks. Currently we are able to solve instances for a whole week.

The presented work shows that solving the whole week can be done with the current state-of-the-art techniques and that it leads to a significant improvement. We tried several alternative approaches to achieve the required variation with the aim to increase the crew satisfaction and we have selected the best approaches. This answers research question 4: *"What is a good way of applying these methods to the research problem?"*

The techniques presented in this chapter are commonly used in practice for solving crew scheduling problems. We show that, after several adjustments and fine-tuning, the extremely difficult instances of NS can be solved. We are aware that other European railway companies are using these techniques for solving their cases. Using the adjustments and fine-tuning, other railway companies could also use these methods to improve their schedules.

Chapter 4

Cyclic Crew Rostering

In the previous chapter, we elaborated on decision support for the crew scheduling problem where anonymous duties are created. Per crew depot, these duties need to be assigned to individual crew members. This is the so called *crew rostering problem*, which is the topic of this chapter. NS uses a cyclic rostering approach. The Cyclic Crew Rostering Problem (CCRP) was introduced in Section 2.4.3.

This chapter describes the algorithms used to solve the CCRP. The applicability of these algorithms will be illustrated by computational results. The rostering problem is solved in two steps: (*i*) the allocation of the duties among the different groups of crew members, and (*ii*) the actual construction of the rosters for each group. The focus in this chapter is on the second step. However, in Section 4.2, we briefly discuss a straightforward solution approach for the allocation of the duties to the groups. In Section 4.3.1, we provide a mathematical formulation for the cyclic crew rostering problem. This formulation is solved with a standard commercial solver. Computational results that show the suitability of this approach are provided in this section as well.

To test the approach in practice, in 2005 we did an experiment for conductors in the crew depot Utrecht of NS Reizigers. The examples and labor rules presented in this chapter reflect the 2005 situation. The results of this experiment are described in Section 4.4. We end this chapter with some concluding remarks.

4.1   Literature Overview and Related Work

Only a few papers have studied the CCRP. For example, Caprara et al. (1998) developed a heuristic, based on a mixed integer programming formulation, to determine a roster with a minimum number of weeks, such that each duty is done once every day.

Sodhi and Norris (2004) deal with the rostering problem at the London Underground, which is a complex problem considering all kinds of hard practical constraints. The authors decompose the problem into two stages, where in the first stage a pattern of rest-days and duty types is created for each depot, and afterward the individual duties are assigned to this pattern. The first phase is the most complex part, which is further decomposed into three steps. The most complicated step is to find the "optimal" rest-day pattern for each depot. This problem is solved as a mixed integer program. The second phase can be formulated straightforwardly and solved as an assignment problem with side constraints.

As explained before, the rostering problem at NS is solved in two steps: (i) the allocation of the duties among the different groups of crew members, and (ii) the actual construction of the rosters for each group. In our solution approach, we have developed support for both steps. Splitting the problem into smaller problems can decrease the quality of the solution. However, the aim of developing support is not to generate more efficient rosters, but to speed up the solution process and get attractive rosters. We will show that our approach both leads to a shorter lead time and to attractive rosters.

4.2   Allocating the duties to groups

The first step is to allocate the duties to the different groups as fairly as possible. This can be seen as an assignment problem where duties should be assigned to different groups. The basic requirements are that each duty should be assigned to exactly one group, and that each group should not have more duties than the number that could be assigned to that group on each individual day. In addition, some restrictions should be taken into account such that each group has a large probability of being able to construct a feasible roster afterward (see Section 4.3).

To allocate the work to the groups as fairly as possible, one could apply two strategies. The first one is that constraints are added which deal with the minimum (maximum) amount of (un)popular work. Alternatively, one could take as objective

that the deviation from the average of popular and unpopular work should be as small as possible for each group. We choose for a combination of these two strategies. Several aspects of the duties are considered to be (un)popular. For instance if a task is performed on a route where there is, in general, a higher amount of aggressive behavior of passengers, this is considered to be an unattractive aspect.

The duty assignment problem above can be formulated as an assignment problem, with additional constraints. In the following, I denotes the set of duties, G denotes the set of roster groups, and A denotes the set of defined aspects of the work. For example, these aspects A can include the amount of popular work in a duty, but it can also be the duration of a duty. The set of possible combinations of different types of roster days (i.e. early, late and night duties), and different weekdays (i.e., Monday, Tuesday, etc.), is denoted as M. Before allocating the duties to the groups, the maximum number of duties per combination m and per roster group g is given, and denoted by N_{mg}. Also, I_m denotes the subset of duties corresponding to weekday m.

For every duty i, we introduce the score function $s_a(i)$, which provides the score on aspect a. Furthermore, we introduce l_{ag} and u_{ag} as the lower and upper bound for the amount of work on every aspect a for every group g. Because we sum the scores for all aspects individually, a high score on one aspect could be compensated by a low score on another aspect. To prevent this, we have added the weight w_a, which enables us to use a weighted sum. We define two types of decision variables. First, the assignment variable x_{ig} is equal to 1 if duty i is assigned to group g and 0 otherwise. Second, variables Z_a and V_a are values per attribute smaller respectively larger than the sum of the function $s_a(i)$ over the duties $(i \in I)$ assigned to group g.

The mathematical formulation of problem reads as follows:

$$\min \sum_{a \in A} w_a \Big(Z_a - V_a \Big) \tag{4.1}$$

$$s.t. \sum_{g \in G} x_{ig} = 1 \qquad \forall i \in I, \tag{4.2}$$

$$\sum_{i \in I_m} x_{ig} \leq N_{mg} \qquad \forall m \in M, g \in G, \tag{4.3}$$

$$V_a \leq \sum_{i \in I} s_a(i) x_{ig} \leq Z_a \qquad \forall a \in A, g \in G, \tag{4.4}$$

$$l_{ag} \leq \sum_{i \in I} s_a(i) x_{ig} \leq u_{ag} \qquad \forall a \in A, g \in G, \tag{4.5}$$

$$x_{ig} \in \{0, 1\} \qquad \forall i \in I, g \in G. \tag{4.6}$$

Constraints (4.2) ensure that every duty is assigned to exactly one group. Constraints (4.3) ensure that the number of duties assigned is less or equal to the maximum number of duties N_{mg}, for every weekday and group. Constraints (4.4) ensure that variables Z_a and V_a are smaller respectively larger than the sum of the score function $s_a(i)$ over the duties ($i \in I$) assigned to group g. Because the values are on the minimized objective function, they will be equal to the lowest and highest score over the groups. For every aspect, the difference between the highest and the lowest score, over the groups, is minimized. Also, for every group, we introduce lower bound (l_{ag}) and upper bound (u_{ag}) constraints (4.5) on the aspects. Next to that, it is important to use the same scale for the score function s_a, for the different aspects. Together with the introduced upper and lower bounds on the aspects, we obtain an allocation of the duties to the rosters that is as fair as possible.

We solve this MIP with the commercial solver CPLEX 9.1, where we put emphasis on finding good and feasible solutions quickly. Moreover, we use CPLEX's local branching heuristic in every fifth node of the branch-and-bound tree. In this way, we could easily find a fair division in several minutes. Because this problem is solved only once a year, and manually it can take days, solving the problem in minutes is considered fast enough.

As an example, we consider the allocation problem for the conductors of depot Utrecht, the largest depot of the Netherlands. The problem at hand consisted of 14 groups with 3 general, 3 late, 3 early and 5 part time rosters. By using the model, we can allocate the duties in such a way that almost all the popular and unpopular work is allocated more or less fairly. The solution time to solve the optimization problem is about 20 minutes, which is much less than the 3 days needed to solve the problem by hand, based on a complex negotiation process.

4.3   Constructing the rosters

When the duties have been allocated to the different groups, each group has to construct its own feasible cyclic crew roster. Since the CCRP is a hard problem to solve, we split it into two phases. In the first phase, we create a pattern where we assign to each day in the roster an early, late or night duty, a day off, a WTV day, a CO day, or a RES day (see also Section 2.4.3. In other words, we assign a type of duty to each day in the roster and not the specific duty. In the second phase, we assign the specific duties to the places in the roster. Note that this split is rather similar to the split that is described in Sodhi and Norris (2004).

Both phases can be formulated as an assignment problem with additional constraints. The objective is to minimize the total sum of the penalties, which are determined by undesirable combinations of duties, days off, etc. The details are provided in the following sections.

4.3.1 Mathematical formulation CCRP

We will first describe the mathematical formulation for the first phase of the CCRP. In the following, T denotes the set of days in the roster, where $|T| = 7k$ for a roster of size k. Since the roster is cyclic, day $|T| + 1$ is the same day as day 1. We define I as the set of roster days. Note that in the first phase the specific duties are not considered. Therefore, only the following type of roster days exist: Early, late and night duties, days off, and CO, WTV and RES days.

The number of duties for each type per day of the week is given and based on the number of specific duties of that type to be assigned to that day in the second phase. All possible combinations of different types of roster days and different weekdays (i.e., Monday, Tuesday, etc.) are denoted by the set M, where for each combination m the number of roster days is denoted as b_m. For some type of roster days (e.g., WTV days), there is no assignment to the different weekdays a priori, in other words it still has to be decided in the CCRP. In that case, b_m gives the total number of these roster days and only one constraint for this type of roster days is added.

Finally, we define two types of decision variables. The assignment variable w_{it} is equal to 1 if roster day i is assigned to day t and 0 otherwise. Note that we only define these variables for feasible assignments, i.e., a roster day representing one of the duties on a Monday can only be assigned to a Monday. Therefore, we define the set TP as all possible combinations of assignments of roster days to days. TP_m is the subset of TP for combination m of a type of roster day and weekday, e.g., all early roster days on Mondays.

The second type of decision variables deals with undesirable patterns of adjacent roster days. These patterns (denoted by the set P) are penalized in the objective function. For each possible pattern p that starts on day t, we define a binary variable y_{pt}, which is 1 if this pattern starts there and 0 otherwise. Its corresponding penalty is denoted by c_{pt}. Note that the value of this penalty can be different on different days of the week.

The mathematical formulation reads as follows:

$$\min \sum_{p \in P} \sum_{t \in T} c_{pt} y_{pt} \tag{4.7}$$

$$s.t. \sum_{i:(i,t)\in TP} w_{it} = 1, \qquad \forall t \in T, \tag{4.8}$$

$$\sum_{(i,t)\in TP_m} w_{it} = b_m, \qquad \forall m \in M, \tag{4.9}$$

$$\sum_{\substack{i:\text{early duty,}\\(i,t)\in TP}} w_{it} + \sum_{\substack{i:\text{night duty,}\\(j,t+1)\in TP}} w_{j,t+1} - 1 \leq y_{1t}, \qquad \forall t \in T, \tag{4.10}$$

$$w_{it} \in \{0,1\}, \qquad \forall (i,t) \in TP, \tag{4.11}$$

$$y_{pt} \in \{0,1\}, \qquad \forall p \in P, t \in T. \tag{4.12}$$

The objective (4.7) is to minimize the total sum of the penalties assigned to the different undesirable patterns. Constraints (4.8) and (4.9) assure that each day has exactly one roster day assigned to it and that each roster day occurs on a particular weekday exactly the right number of times, respectively.

As an example, we included one set of constraints (4.10) in the model. This states that, if there is an early duty on day t and a night duty on the next day $t+1$, then the corresponding y-variable to this pattern y_{1t} should be equal to 1. In that case, we count the corresponding penalty in the objective. Since the rosters are cyclic, the first day of the roster group is the successor of the last day of the roster group. In a similar way, all other undesirable patterns can be defined and added to the model. This is not included in the presented formulation. In our computations, the set P contained 48 undesirable patterns. Moreover, all labor rules can be added in a similar way. We discuss some examples in the next section.

For the second phase the mathematical formulation is almost identical. The main difference is that the roster days are replaced by specific duties. This implies that $b_m = 1, \forall m \in M$, a specific duty can only be assigned to the corresponding specific day of the week.

4.3.2   Labor rules constraints

In Section 2.4.3, we gave a few examples of labor rules. We will now show for these examples how we model them into additional constraints. For the additional constraints, the set of days T is split in sets $T_1, \ldots, T_7$, where T_1 represents the set of Mondays, T_2 represents the set of Tuesdays, etc.

The first example of a labor rule deals with the rest time between two duties. In the first phase we do not take the specific duties into account, therefore, we cannot handle this rule directly . However, we know that, if we do not take it into account at all, it is likely that we cannot find a feasible solution in the second phase. Therefore, we do not allow some combinations in the first phase that give a large probability of a too short rest time. Alternatively, we could allow them but with a high penalty. In other words, we could translate this rule in both phases into a rule, which says that roster day i could not be followed by roster day j on the next day, or in mathematical terms:

$$w_{it} + w_{j,t+1} \leq 1. \tag{4.13}$$

The maximum working time per calendar week is the second example. This rule is taken into account in the second phase. By defining d_i as the working time of duty i, D as the maximum working time, and s as the number of days in this period, we can formulate this as:

$$\sum_{n=t}^{t+s-1} \sum_{i:(i,n)\in TP} d_i w_{in} \quad \leq \quad D \quad \forall t \in T_1 \tag{4.14}$$

The next example, also taken into account in the second phase, deals with the fact that a day off should cover at least 30 hours. In other words, it tells something about the time between two duties, on days t and $t + 2$, with a day off in between. The following constraints are added to the second phase:

$$w_{ht} + \sum_{\substack{\text{day off } i:\\(i,t+1)\in TP}} w_{i,t+1} + w_{j,t+2} \quad \leq \quad 2 \quad \forall t \in T, \tag{4.15}$$

where the start time of duty j is less than 30 hours after the end time of duty h and $(h,t), (j,t+2) \in TP$.

The so-called "Red Weekend" is an example, which results in constraints in the first and second phase. In both phases, we add the constraints:

$$\sum_{p\in RW} (y_{p,t} + y_{p,t+7} + y_{p,t+14}) \quad \geq \quad 1 \quad \forall t \in T_5 \tag{4.16}$$

where the set RW defines all patterns containing a Red Weekend. The difference between the first and second phase is in the definition of RW. In the second phase, we take the end time of the duty on Friday and the start time of the duty on Monday into account as well.

Our last example deals with the duties which cover a part of the period between midnight and 6:00 AM[1]. These duties form the set ZN. We define s as the period length in days in which at most Z of these duties are allowed. This can be modeled as:

$$\sum_{n=t}^{t+s-1} \sum_{\substack{i \in ZN: \\ (i,n) \in TP}} w_{in} \quad \leq \quad Z \quad \forall t \in T_1 \tag{4.17}$$

This constraint is taken into account in the second phase only.

4.3.3   Computational results

In 2005, we performed some experiments for the crew depot Utrecht of NS Reizigers, which contains 12 groups of train drivers. Some characteristics of the groups are given in Table 4.1. These data correspond to the year of the experiments.

group	size	average working time	specifics
A	12	36 hours	no night duties
B	20	36 hours	no night duties
C	20	36 hours	no night duties
D	20	36 hours	all types
E	20	36 hours	all types
O1	20	36 hours	only early duties
O2	10	36 hours	only early duties
L1	20	36 hours	no early duties
L2	12	36 hours	no early duties
DO	10	32 hours	only early duties
DR	12	32 hours	all types
DL	12	32 hours	no early duties

Table 4.1. Driver groups of the crew depot Utrecht

We have developed and tested our model on 7 of these groups. The computational results for the first phase are given in Table 4.2. In this table, the objective values, lower bounds and computation times are reported every time that a significantly better feasible solution is found. In addition, we report these values after a reasonably long computation time.

---

[1]All night duties cover a part of this period. However, some early and late duties cover a part of this period as well. Due to a change in the official Driving Time Law, this period was changed, after the experiment in 2005

group	computation time	objective value	lower bound
A	1 min	37.3	0
	10 min	34	0.43
	1 hour	34	2.6
B	23 min	80.8	6.2
	45 min	58.9	6.6
	2 hours	58.9	7.5
D	48 min	99.9	6.2
	6 hours	82.6	6.9
	11 hours	68.9	7.5
	16 hours	68.9	7.5
O2	1 min	42.6	0
	16 min	27.4	3.7
	1 hour	27.4	6.6
L2	3 min	58	0
	30 min	37.1	0.5
	1 hour	37.1	1.3
DO	1 min	12.4	4.1
	17,5 hours	12.4 (optimal)	–
DR	1 min	37.3	2.3
	20 min	16.5	4.9
	1 hour	16.5	7

Table 4.2. Results first phase

For group DO, we can see that we find after 1 minute a solution with value 12.4. At that moment, the lower bound is 4.1 and the gap is very large. However, for this instance we could prove after 17.5 hours of computation time that the solution with value 12.4 is indeed the optimal one. For the larger instances, we could not prove this. However, we noticed that the solution is never improving after 30 minutes of computation time, while the lower bound only slightly increases. We believe that at the end one could prove that the solution we found is indeed the optimal one (or very close to it). So we conclude that we find a "good" solution quickly.

To help CPLEX find good feasible solutions early in the process and, therefore, to reduce computing time, we have added a redundant constraint:

$$\sum_{p \in P} \sum_{t \in T} c_{pt} y_{pt} \leq \alpha k \tag{4.18}$$

for an appropriate choice of α. This gives an upper bound on the objective function, which depends on the size of the roster as well as on the penalties for the undesirable

patterns. The parameter α should be selected with care in order to assure that the feasible region is non-empty.

For the second phase, the results are given in Table 4.3. For almost all groups, an optimal solution can be found within a reasonable amount of time. An exception is group B, where no optimal solution is found after more than 20 hours of computing time. Adding constraint (4.18) gives much better results for group B. After 7 minutes, a solution with objective 231.6 and lower bound 67.8 is found and after two hours for the same solution the lower bound has increased to 100. This shows that adding the constraint can lead to finding better results in shorter computing time.

group	computation time	objective value	lower bound
A	2.5 min	209	68
	1.7 hours	93.6	90.5
	8.8 hours	93.6 (optimal)	–
B	30 min	752.3	76.2
	4 hours	583.7	84.3
	20 hours	583.7	90
D	8 min	871.1	149
	10 min	245.5	170
	3.2 hours	217.5 (optimal)	–
O2	2 min	56	39
	3 min	48	40
	7 hours	48 (optimal)	–
L2	2 min	33	12.8
	17 min	27 (optimal)	–
DO	2 min	150	32.9
	3.6 hours	73	68
	11 hours	73 (optimal)	–
DR	1.5 min	129.3 (optimal)	–

Table 4.3. Results second phase

As an example of the solutions found, we present the obtained first [phase solution for group D in Table 4.4. This roster is a full-time roster for 20 weeks (crew members) and contains all types of duties. Early, late and night duties are denoted by E, L and N, respectively. A day off is denoted by R. The solution for the first phase is found after 11 hours with an objective value of 68.9 and a lower bound of 7.5. For the second phase, the optimal solution is found after 3 hours with an objective value of 217.5. Only the results of the first phase are presented in the table.

Wk	Mo	Tu	We	Th	Fr	Sa	Su
1	RES	E	R	L	WTV	R	R
2	E	E	E	E	E	E	R
3	R	L	WTV	R	R	E	E
4	E	E	E	E	E	R	R
5	L	L	WTV	R	L	N	N
6	N	R	R	L	L	L	L
7	WTV	R	L	L	RES	R	R
8	E	E	E	E	E	R	R
9	L	WTV	R	R	N	N	R
10	R	L	L	RES	CO	E	E
11	WTV	R	E	E	E	R	R
12	L	N	N	R	L	L	L
13	R	R	RES	L	L	L	L
14	CO	L	L	L	WTV	R	R
15	L	L	L	L	L	R	R
16	E	WTV	R	R	L	L	L
17	R	RES	E	E	E	R	R
18	L	L	L	WTV	R	E	L
19	L	N	N	N	CO	R	R
20	L	L	L	WTV	R	E	E

Table 4.4. Example roster group D

First, we give some remarks about the results of the first phase. Within the roster, the WTV days are nicely spread over the weekdays and the weeks, and are all followed by a rest day. Ten times there is a single day off, but in most of the cases days off are clustered. Three times, a night duty is scheduled after a late duty, and once a night duty is scheduled after an early duty. These are undesirable combinations. However, in general the sequencing of the duties is quite attractive.

The next remarks are related to the results of the second phase, for the same roster group D, not included in the table. The generated roster contains only two weeks with an average working time above 36 hours. The roster contains two duties with a post time of 13 hours, seven with a post time of 14 hours, 15 with a post time of 15 hours and 57 duties with a post time of more than 16 hours. Most of the routes occur in every week in less than 3 duties (which spreads the routes equally over the weeks). Also, the nice trains have been spread over the weeks fairly.

4.3.4   Portability

Cyclic Crew Rosters are commonly used in many public transport companies. We present a generic approach of modeling the problem, which can be applied to these other companies as well. We have modeled the regular labor rules as well as the variation rules in a manner that can easily be extended. Due to the decomposition of the problem, we are able to solve quite large rostering instances. The sub-problems themselves are solved with a general purpose solver which can be bought off the shelf. The main contribution of this work is the way of modeling the desired roster patterns which can be used in other companies as well.

4.4   Experiments

The conductors in the crew depot Utrecht had some problems with the creation of the rosters in 2005, and we were asked by the management to support the planning of the rosters for the year 2006. The management was aware of our research that we performed in 2005. The idea was to generate rosters parallel to the manual planning process and to evaluate the differences.

As mentioned in Section 4.2, the first step in the rostering process is to allocate the duties to the different roster groups. The problem at hand in Utrecht consisted of 14 groups with 3 general, 3 late, 3 early and 5 part time rosters, all for train conductors. Representatives of the different groups took 3 days to allocate the duties over the different groups. The popular and unpopular work was allocated reasonably fairly to the different groups. However, there was one group with a lot of unpopular work. By using the approach described in Section 4.2, we could allocate the duties in such a way that almost all the popular and unpopular work was allocated more or less fairly. The solution time to solve the optimization problem was about 20 minutes, which is much less than the 3 days needed by the representatives. The representatives did not use the generated solution, but slightly modified their allocation based on our solution, such that the group with a lot of unpopular work got some other duties in order to increase the overall fairness of the allocation.

In the next step, the construction of the cyclic rosters, the main practical problem was related to the assignment of the duties to the pattern rosters, the second phase in our approach. The first phase, where the patterns are created, were less important in our experiment. The groups used the same patterns for the rosters as in the previous year. Based on the patterns, an attractive solution for the second phase was found within one hour for all groups, except for the C group. In this case, it was not

possible to find a feasible solution based on the given pattern.

After generating a new pattern, we were able to find a good roster for group C. For the smaller or part time rosters we found an optimal solution almost instantly. The larger cases took some hours to be solved to optimality. Manually, the planners were not able to find a feasible solution for three rosters within the available time.

In these cases, we presented both the manually created roster and the roster generated by the model, where the manually created rosters did not obey all labor rules. The violated rules were related to the official national regulations. However, these violations are not commonly known and hard to detect. The personnel did not know which rosters were generated by the model, and which ones were generated manually. In all three cases, the personnel preferred the rosters generated by the model. As a result of this, three of the rosters generated by the model have been implemented in practice. This shows that our model is not only successful in creating valid rosters but also creates attractive rosters.

4.5 Conclusions

In this chapter, we described a model for solving the CCRP. The model was tested on several instances involving groups of train drivers or conductors of NS, the largest operator of passenger trains in the Netherlands. The results were promising: Rosters for the crew depot Utrecht were generated in a fraction of the time that was required by the manual rostering process. In addition, the rosters generated by the model satisfied all the relevant rules, which was not the case for the manually generated rosters. Moreover, the rosters generated by the model were preferred over the manually generated ones.

In Section 4.1, we answered research question 3: *"What are commonly used methods for supporting the Crew Management process?"*, by elaborating on the commonly used techniques for solving the CCRP. We presented a formulation to describe the desired roster patterns and have shown that solving the problem using this formulation with a general purpose solver (CPLEX) gives good results. This answers research question 4: *"What is a good way of applying these methods to the research problem?"*

As stated in the introduction of this chapter, next to cyclic rosters, rostering can be done in several ways. NS is also interested in creating a roster for individual crew members where individual preferences (e.g., their vacations) can also be taken into account. NS has performed some experiments with this way of creating rosters and wants to implement this in the future rostering system. Therefore, the introduction

of decision support for the current method was halted. A project on the development of a new roster system was halted shortly after initiating it. This because there was no real need to change working methods and the current system was stable enough to be used for several more years. Since then, no improvements have been made on the development of a decision support system for the rostering problem.

Summarizing, we have shown once more that the application of Operations Research techniques for solving a practical problem may lead to a shorter throughput time in the planning process, as well as to a higher quality of the obtained results. In particular, since the results are appreciated by the crew, this can lead to more satisfied crew. It can be expected that the more satisfied crew will lead to a higher productivity of the crew and to a better punctuality of the railway services. Due to circumstances outside the scope of this research, the results were not implemented in practice.

Chapter 5

Real-time Crew Rescheduling

Railway operations are based on an extensive planning process. As illustrated in the previous chapters, crew scheduling and rostering are complex processes. After the planning process, the daily plans are carried out on the day of operation. Preferably, the plans are carried out as scheduled. However, at the day of operation, plans have to be updated frequently to deal with delays of trains and disruptions of the railway system.

For example, a disruption may be due to an incident, or a breakdown of infrastructure or rolling stock. On the Dutch rail network (with more than 5,000 trains per day), on average 3 larger disruptions occur per day, where a number of train services need to be canceled. Each day, about 2,300 crew duties are carried out. Furthermore, at any moment in time, the number of active crew duties at that moment is about 700. Several trains may have to be canceled, due to a disruption. As a result, the planned rolling stock circulation and crew duties are not feasible anymore and must be rescheduled. For an overview of the disruption management process, we refer to Jespersen-Groth et al. (2009). This chapter presents a method, based on multi-agent techniques, to solve the crew rescheduling problem in case of a large disruption with multiple train services canceled.

The presented research system is structured according to the Actor–Agent paradigm: Here agents assist in rescheduling tasks of train drivers or conductors. Coordination between agents is based on a team formation process in which possible rescheduling alternatives can be evaluated, based on constraints and preferences of involved human train crew and dispatchers. The main objectives of the train-driver rescheduling system described in this chapter are to explore the effectiveness and

suitability of a decentralized, actor-agent based approach to crew rescheduling and to determine whether multi-agent technology is sufficiently mature to be used in a real-world decision support system.

Important aspects are the ability to generate adequate solutions in a reasonable time, and to be able to communicate them before changed duties need to be carried out. Next to that, reality is changing constantly, and solutions might not be valid anymore if the solution process takes too much time.

We compare this method with an alternative approach called Column Generation with Dynamic Duty Selection (CGDDS), which was developed by Potthoff et al. (2010) (see also Potthoff (2010)).

5.1   Literature Overview and Related Work

Traditionally, crew scheduling problems are approached using Operations Research techniques. Real-time crew rescheduling, however, is a relatively new area of research. NS decided to evaluate two techniques on their suitability to solve the Real-time Crew Rescheduling Problem.

First, Potthoff et al. (2010) present an algorithm to reschedule NS crew when a disruption occurs. They study the same problem as presented in this chapter. Their CGDDS algorithm is based on column generation techniques combined with Lagrangian heuristics from Operations Research. To handle the very large number of duties in practical instances, they first define a core problem, with a limited number of duties. If some tasks remain uncovered in the solution of the core problem, they perform a neighborhood exploration to improve the solution by adding additional duties to the core problem. Computational experiments with real-life instances show that their method is capable of producing good solutions within a couple of minutes of computation time.

This chapter describes a second approach, agent-based crew-rescheduling, which is a relatively new area of research. We will compare the results of CGDDS with the solutions of the method presented in this chapter (see Section 5.4.3). To our knowledge, no research has been published on agent-based crew rescheduling applications in the railway domain. We will first discuss some literature, related to our agent-based approach to solve the problem.

Shibghatullah et al. (2006) propose an agent-based framework for bus crew scheduling including crew-reassignment. The paper provides an overview of the potential advantages of agent-based approaches (e.g., modeling individual preferences, more

suited for partial, on-demand rescheduling), but lacks further details of the proposed framework.

In Castro and Oliveira (2011), an agent-based approach to airline operations recovery is described, part of which concerns crew recovery. The architecture consists of specialized agents representing parts of the traditional Airline Operations Control Center organization. Similar to our approach, costs are assigned to various factors such as hotels and extra crew tasks between the different operational depots. One of the agents handles the crew rescheduling problem integrally, using a hill climb algorithm from artificial intelligence. In our method, we propose a more decentralized approach, using multiple agents to solve the crew rescheduling problem.

Tranvouez and Ferrarini (2006) present a multi-agent based approach to disruption management in the supply chain domain. A distinction is made between partial and complete rescheduling, as well as periodic and event driven rescheduling. Partial rescheduling involves only the affected operations, not the complete schedule; it is denoted as repair scheduling when it refers to local repair of the current schedule. Event driven rescheduling means that the rescheduling process is started when events occur that make the current schedule infeasible. It is argued that partial, event driven rescheduling appears to increase schedule stability. Schedule stability is an important and desirable characteristic in the railway crew rescheduling domain, given the fact that initial crew schedules have already been optimized for efficiency, and changes should be minimized. Tranvouez and Ferrarini (2006) further describe a BDI architecture, for its potential to design complex decision making processes and to represent rich domain expert knowledge.

Mao et al. (2007) recognize the need for short-term operational planning and scheduling methods in the domain of airport resource scheduling, and present an agent-based approach based on two coordination mechanisms: Decommitment penalties and a Vickrey auction mechanism. In this chapter, we present a coordination approach, based on a combination of similar mechanisms: A driver-agent interaction protocol with auction-like properties (agents report costs (i.e., bid) for taking over tasks), and decommitment penalties are determined based on increasing commitment levels.

An important factor to consider when dealing with rescheduling in this domain is the dynamic environment and the related uncertainty. Estimates of the duration of a disruption can be changed as more information is obtained; the timetable can be adjusted during the disruption due to knock-on effects, etc. De Weerdt et al. (2005) state in their overview of multi-agent planning, that although most researchers

recognize the importance of dealing with changing environments, most planning approaches still assume fairly stable worlds. The authors mention contingency planning (plans for all contingencies that might occur) as a traditional approach of handling changes in the environment. As in many situations planning for all possible contingencies is not feasible, the authors argue that so-called plan repair approaches are more realistic: Detecting deviations from the original plan through monitoring, and adjusting the plan as needed.

DesJardins et al. (2000) present an overview of approaches in the field of distributed planning. In the paper, approaches are classified according to the properties they share with cooperative distributed planning (emphasis on forming a global (optimal) plan and negotiated distributed planning (emphasis on satisfying local goals). The authors argue that only recently research in this field has been concerned with coping with dynamic, realistic environments. To cover this emerging work, the authors introduce the distributed, continual planning paradigm. This paradigm considers planning to be a dynamic ongoing process combining both planning and execution. The work presented in this chapter, fits this paradigm, as the crew rescheduling process is performed in real-time and disruptions continuously require agents to revise their duties to cope with new circumstances.

5.2   Solution Approach

In this section, the main principles underlying the actor-agent based crew rescheduling process are introduced. First, the applied design paradigm is introduced. After this, the two main elements of the rescheduling system are described.

5.2.1   Actor-Agent Paradigm

Wijngaards et al. (2006) define that the actor-agent paradigm explicitly recognizes both human actors and artificial agents as equivalent team members, each fulfilling their respective roles in order to reach the team objectives. The involved agents have quasi cognitive capabilities that are complementary to (as opposed to mimicking) human cognition. Actor-agent teams and communities are hybrid collectives of human experts ("actors") and agents with complementary cognitive capabilities which are focused on a particular problem and reach a structural and functional complexity that matches the size and nature of the problem as well as possible.

The actor-agent based design process provides the system with several useful global system characteristics. First, the decentralized approach in which agents use

local knowledge, world views, and interactions, contributes to an open system design. This openness facilitates easy reconfiguration and/or adaptation to changing system requirements. Second, combining humans and agents within the system design allows for integrating them at their appropriate abstraction levels.

The main objective of the crew rescheduling application is to realize a decentralized (multi-agent based) application which provides solutions faster than human dispatchers. In addition, the solutions should preferably minimize changes to original duties.

5.2.2 System Architecture
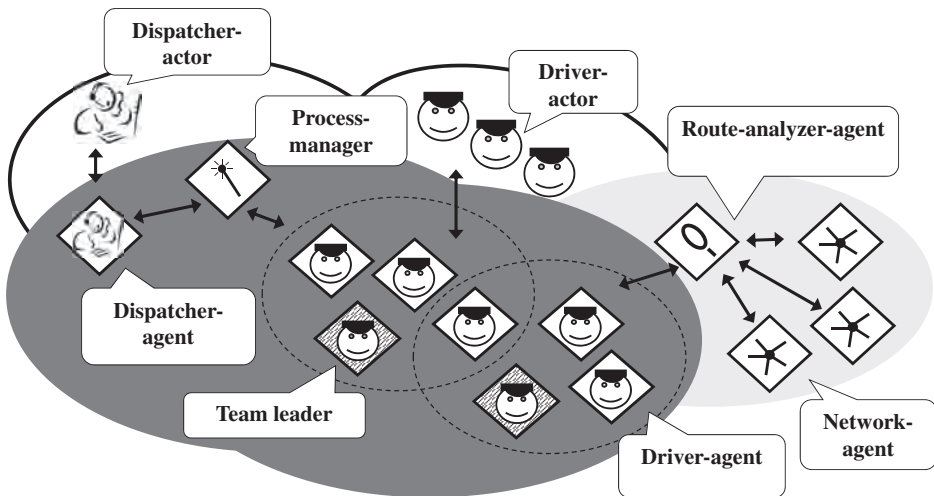
Figure 5.1. Current system architecture.

The following actors and agents are recognized in the system architecture (see Figure 5.1):

- Dispatcher-actor: A (human) dispatcher, who interacts with the rescheduling system via one dispatcher-agent.

- Driver–actor: A (human) train driver[1], who imposes constraints on the rescheduling process based on the preferences he/she may have with respect to

---

[1]In this research we addressed the train driver rescheduling problem. This approach can also be used for train conductors.

performing his/her duties. These constraints can be hard (e.g., familiarity with rolling stock types) or soft (preferences for certain lines). Each driver–actor is associated with one driver–agent.

- Dispatcher–agent: Presents a monitoring and control interface on the rescheduling process to the dispatcher–actor.

- Process–manager–agent: The process–manager–agent (PMA) manages the rescheduling process, and provides a contact point for the rescheduling subsystem to dispatcher–agents. The PMA coordinates the rescheduling process by communicating the disruption information and parameters to the driver-agent population. The PMA maintains status information of the negotiation process and informs the dispatcher–agents.

- Driver–agent: Responsible for resolving conflicts arising in duties due to disruptions. Each driver–agent (DA) represents a specific driver–actor[2] in the rescheduling process. Driver–agents are capable of forming task exchange teams, resolving a scheduling problem, and participating in multiples of such teams.

- Route–analyzer–agent: To determine the impact of changes to their existing duties, driver–agents interact with a route–analyzer–agent (RAA). The RAA interacts with network–agents.

- Network–agent: A network–agent (NA) maintains an up–to–date view of the railway network, reflecting any changes in the timetable and rolling stock schedules due to disruptions. The NA processes queries from the route–analyzer–agent.

In the implemented system, we developed versions of the (software) agents as presented in the system architecture. During the research, we acted as Dispatcher–actors, using the system interface provided by the developed Dispatcher–agent. This interface enabled us to start and stop the solution process, monitor progress and evaluate the results. There was no specific interface developed between the Driver–actor and the Driver–agent. This was outside the scope of our research.

5.2.3   Crew Rescheduling

The main principle for the crew rescheduling application is to model the solution based on "levels of responsibility" in the dispatch and rescheduling process:

---

[2]In case of a train conductor, the corresponding actor is a conductor–actor.

1. Human dispatchers at the management level;

2. Human train crew at the level of defining and guarding their personal interests;

3. Their respective agents at the level of implementing the management decisions and resolving actual schedule conflicts.

In our approach driver-agents represent driver-actors in the rescheduling process. Schedule conflicts can be divided into two categories (see Jespersen-Groth et al. (2009): Time-based and location-based. A location-conflict can occur when one or more tasks are removed from a schedule due to a disruption. A time conflict occurs when due to a delay the arrival time of one task in a duty is later in time than the departure time of the following task. Other disruptive circumstances (such as a train driver becoming unavailable to perform the remainder of his duty) are translated into these two types of conflicts. For instance, the unavailable train driver can be considered to have a location conflict because he is not at the location at the time he should perform the remainder of his duty. As an alternative, he could have a time conflict, with the introduction of a task of type "absent" into his duty which overlaps in time with the other tasks.

The basic principle underlying the solution process is called task exchange. Each driver's schedule consists of a number of tasks. If, in the event of a disruption, a driver can no longer perform one or more tasks due to one of the two types of schedule conflicts, these tasks are taken over by another driver. In turn, this driver may have to hand over tasks which conflict with the newly accepted tasks to another driver.

The process of finding task exchanges which are feasible and desirable is performed by the driver-agent population. In the event of a disruption, all driver-agents are informed of the disruption details (i.e., removed/delayed train services). The driver-agent(s) directly affected by a disruption (i.e., the disrupted train service is associated with a task in the driver's schedule) assume the role of team leader. Each team leader starts a team-configuration process in order to resolve their respective schedule conflicts. Using cost functions, the most favorable solution is selected. In the next section, this process is described in more detail.

5.2.4 Task Exchange Teams

A team leader starts a recursive team extension process by announcing its set of conflicting tasks to the driver–agent population. The main principle underlying the actor-agent based rescheduling process is that of task exchange teams: Each team is

extended with additional team members able to take over tasks from agents already participating in the team. A driver-agent may be able to take over tasks without affecting other tasks already present in its schedule, for example replacing a positioning task with a task on the same task (unconditional takeover). However, in many cases, a driver-agent will only be able to take over tasks by replacing existing tasks in its schedule (conditional takeover). This will then lead to a new set of conflicting tasks to be taken over by another driver-agent.
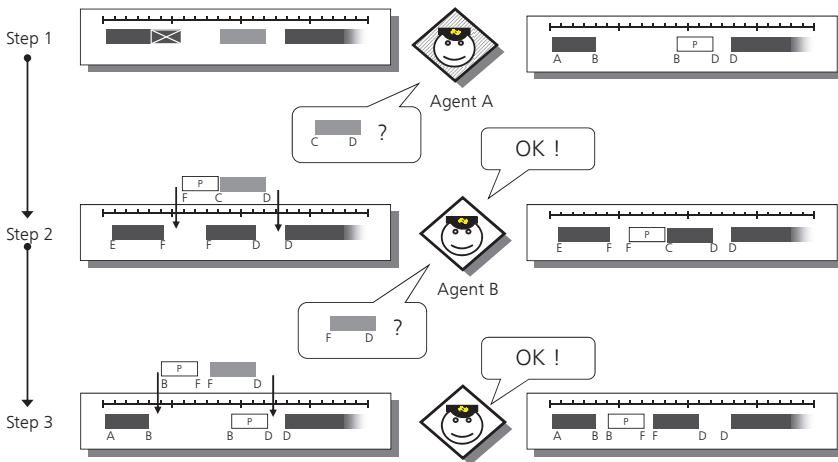
Figure 5.2. The task takeover process

In Figure 5.2, an example of this process is shown: Driver-agent 2 replaces task F-D with task C-D from driver-agent 1. Driver-agent 1 now takes over task F-D from driver-agent 2 unconditionally. In a task exchange, driver–agents assume the role of team leader or team member. In both cases, the aim of the agent is to resolve a conflict: In case of the team leader role, the conflict is due to modified tasks (e.g., as a result of a disruption, tasks may have been removed or delayed). In case of the team member role, the conflict is due to additional tasks originating from another driver-agent.

It is possible for driver-agents to participate multiple times in task exchanges within the same team (and in other teams). This allows teams to discover configurations in which driver-agents "trade" tasks. Figure 5.3 shows an example of three team configurations: A–B, A–C–D and A–C–A.
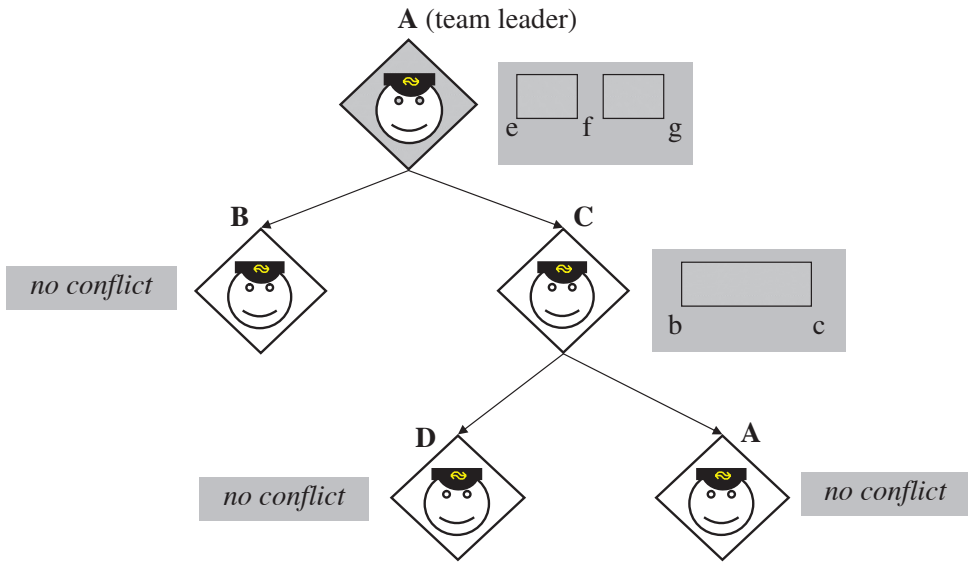
Figure 5.3. Team configuration example

In case a driver-agent has determined that tasks can be taken over, a cost-function is applied to determine the costs associated with the takeover: Costs are assigned to various aspects such as the amount of overtime introduced, replacement of meal breaks, etc. Subsequently the set of new conflicting tasks of this agent (if any) is announced to other driver-agents. This leads to a recursive addition of layers of team members to the team, resulting in a team consisting of multiple task exchange configurations, originating at the team-leader.

In Figure 5.4, two team configurations are shown, consisting of three driver-agents (A, B and C): In the left team, driver-agent A is a team leader and starts the task exchange process. The same driver-agent participates as a team member in its own team, as well as (in two different configurations) in the team led by driver-agent B: This feature allows for any driver-agent to participate in possible task exchange configurations and thus to facilitate the solution process to find the best task exchange configurations for each team.

Driver-agents can withdraw themselves from teams and team configurations based on commitment levels in the task exchange protocol, ensuring local and global consistency when final team-configurations are determined. This is described in more detail in the next section.
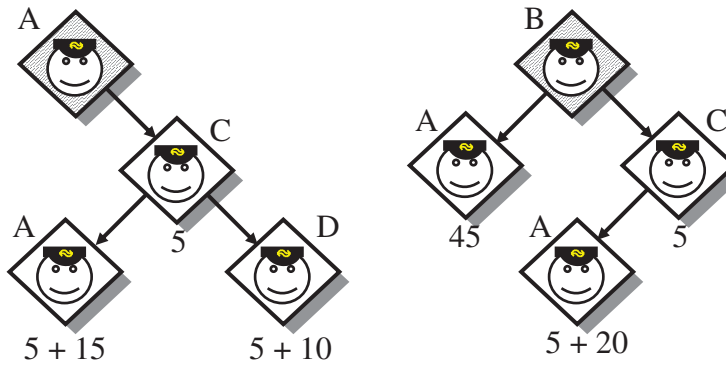
Figure 5.4. Two example task exchange teams consisting of three agents.

The team extension process is considered complete when a configuration of task exchanges is determined in which all conflicts are resolved, or any remaining conflicts are sufficiently shifted forward in time to be resolved at a later point in time (re-introduced as new conflicts later). At this point, the recursive team formation process is reversed: Each layer within a team selects the task exchange associated with the lowest cost, starting at the lowest layer.

Once all team leaders have determined a final team configuration, the entire solution is presented to the dispatcher. During the team formation process, a number of protocols are used, aimed at limiting team extension to promising additional team members. These formation protocols are described in Section 5.2.5.

Task Exchange Protocol

When a driver-agent is a team leader or left with a new conflict as a result of accepting a task exchange, an attempt is made to hand over the conflict to another driver-agent using the task exchange protocol. In Figure 5.5, the message exchange is shown for a number of scenarios, discussed below.

The protocol is initiated by a driver–agent in the role of team leader (TL): A call is issued by TL announcing the conflicting tasks to all driver–agents. Driver–agent DA1 has determined that it cannot participate in the task exchange (i.e., either the exchange is infeasible or the associated costs are too high), and informs the TL by responding with a "not-interested" message. Driver–agent DA2, on the other hand, is able to participate in the exchange, and indicates this by responding with an "interested" message. DA2 indicates that its interest is "conditional", meaning that
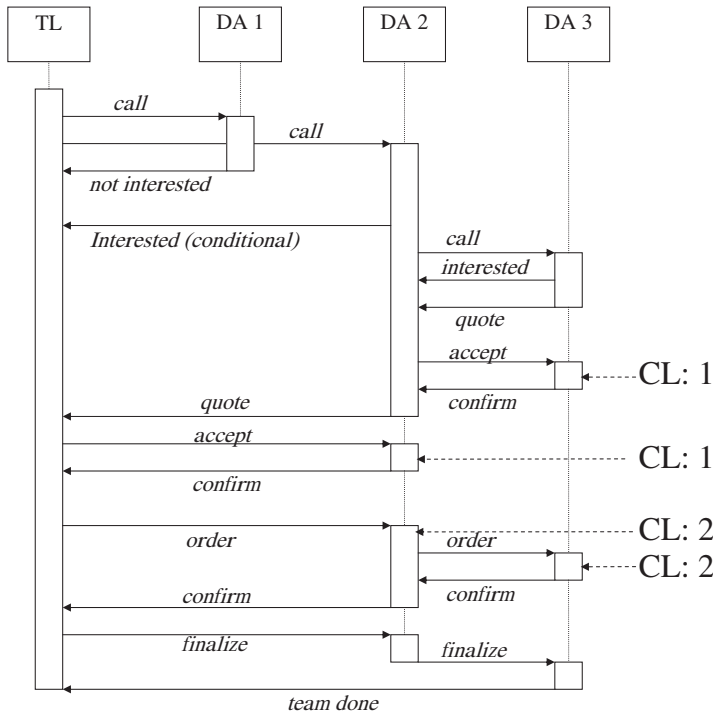
Figure 5.5. Task exchange protocol

DA2 first has to exchange a new conflict to another driver-agent to solve the conflict from TL. DA2 now initiates its own task exchange protocol, to which DA3 responds with an "interested" message. In this case, DA3 does not generate a new conflict. Instead, the costs for the task exchange are calculated by DA3 and communicated to DA2 using a "quote" message.

When DA2 has received sufficient quotes, the most favorable quote is selected. In this example, DA3's quote is selected, and DA3 is informed using an "accept" message. DA3 confirms the acceptance using a "confirm" message. Based on the confirmed quote received from DA3, and the costs calculated for its own task exchange with TL, DA2 can now report the total costs for the takeover to TL by sending a quote to TL. Similar to DA2, TL collects quotes and confirms the quote received from DA2. TL can finally decide on finishing the task–takeover process by issuing an order to DA2 (which is passed on to DA3): The returned "confirm" messages indicate that the team is ready to be finalized, which is the final message sent by TL. The protocol ends by DA3 returning a "team done" message to TL.

Commitment Management

The design of the task exchange process explicitly supports agents participating multiple times in task exchange teams. To prevent driver-agents from committing in conflicting task exchanges, a commitment management mechanism is included, based on commitment-levels, see Sandholm and Lesser (2002).

The task exchange protocol is divided into sections. Each section is assigned a commitment level. At specific points in the protocol (i.e., between these sections), a commitment level change occurs. In Figure 5.5 the commitment level changes are shown (CL). At these points, a driver-agent applies a commitment strategy, which specifies whether ongoing task exchanges are allowed to proceed to the next commitment level:

- CL 1: Upon receiving an "accept message", a driver-agent needs to decide whether the task exchange is still favorable, before confirming the acceptance.

- CL 2: Upon receiving an "order" message, a driver-agent must ensure that only non-overlapping task exchanges are allowed to proceed, before confirming the order.

Every commitment level increase makes it more difficult for that driver-agent to decommit from this specific task exchange. In the final commitment level (i.e., CL:2), a driver-agent must ensure that any ongoing task exchanges, that overlap with the fully committed task exchange, is decommitted.

Determining Duty Impact

Upon receiving a request for a task–takeover, a driver-agent sends its (up-to-date) duty and the received conflict (i.e., tasks to be taken over) to the route-analyzer-agent (RAA). The RAA processes the request. In case the RAA has determined that the conflict can be integrated into the current driver's duty, the RAA's reply to the driver-agent contains two parts: A delta on the original duty (Δ) describing the required modifications to the duty, and a new conflict (NC) resulting from the modifications (i.e., tasks replaced with tasks from the conflict). The second part may contain zero or more tasks: In the first case Δ did not lead to any tasks being replaced (i.e., the conflict is resolved without introducing another conflict, e.g., because the new tasks are replacing a break in the driver's duty). For more information on the process of determining routes, see Section 5.2.6.

5.2.5 Team Formation Protocols

Due to the large number of driver-agents, engaged in various roles in the team formation process, and the fact that a short calculation time is an important success factor (as explained at the beginning of this chapter), protocols are used to constrain the extension of teams with additional team members. The main goal behind the applied protocols is to only let driver-agents participate that have a high probability of improving the solution found. In this section, the applied protocols are described: Interest determination, cost calculation, evaluating team score and decommitment determination. Below, these will be discussed in more detail.

Interest Determination

Based upon the schedule impact determined earlier, a driver-agent decides whether it is interested in joining a team. Several domain-specific criteria are evaluated to determine a takeover desirability:

- If the impact of taking over a conflict results in a new conflict which is larger than the original one, the agent is not interested.

- Taking over tasks from a driver-agent may only result in introducing a new conflict if the tasks in the new conflict are situated later in time.

- If the new conflict introduced by taking over tasks from a driver-agent consists of the same set (or superset) of tasks introduced by this agent elsewhere in the team, the agent will not be interested. This prevents repeating evaluation of sequences of task exchanges that have already been evaluated.

Cost Calculation

A driver-agent interested in taking over tasks must determine the costs associated with this takeover. The cost function is non-decreasing and assigns costs to the following elements:

- The amount of overtime introduced by extending a schedule past the original end time;

- An affected meal break;

- The use of spare drivers. Spare drivers are preferred in case of larger disruptions, and should not be used for resolving small/simple conflicts;

- The size of the team (e.g., preferring recurring team members);

- The individual train driver preferences (i.e., modifiers to the above elements).

The relative weights of the cost elements can be varied to prioritize different aspects of the solutions that are found (e.g., prevent the use of spare drivers by increasing their relative cost). The costs are subsequently compared to costs of other exchanges using a scoreboard mechanism, described below.

Team-Configuration Scoreboard

Upon starting a task exchange process, each team leader publishes a scoreboard, which can be used by team members to inform other driver-agents of the progress within a task exchange team. The scoreboard is based on the principle that every driver-agent in a configuration of a team has knowledge of the "cost" of the exchange configuration (the task exchanges leading up to the task exchange this driver-agent is currently examining).

The moment a driver-agent has determined that a conflict can be resolved without generating a new conflict, it publishes the value of the current solution (i.e., the cost of one complete (possible) configuration) on the scoreboard. All driver-agents can access the scoreboard and examine the costs of the solutions which have already been found. Driver-agents can use the values on the scoreboard to decide whether to continue a task exchange or not. If the cost of the current partial solution is already the same or higher than the scoreboard value, the driver-agent will terminate its involvement in this branch of the task exchange.

In Figure 5.6, an example is shown: First, driver-agent B determines that it can solve the conflict of driver-agent A(TL). It finds that the scoreboard is still empty, and updates it with the calculated costs (arrow 1). Subsequently, driver-agent C finds that in order to solve the TL's conflict, it has to introduce a new conflict. Before continuing, it checks the scoreboard to determine if the current costs still are an improvement.

In this case, driver-agent C continues. Driver-agent D determines that it can solve the conflict of driver-agent B without introducing a new conflict. It also determines that the cost of the solution improves the current score on the scoreboard. Driver-agent D updates the scoreboard (arrow 2). Finally, driver-agent A(DA) aborts, as it has determined that its (partial) solution does not improve the team's current score.

The scoreboard mechanism ensures that only solution alternatives are evaluated which improve the currently found solution(s). Note that this mechanism depends on
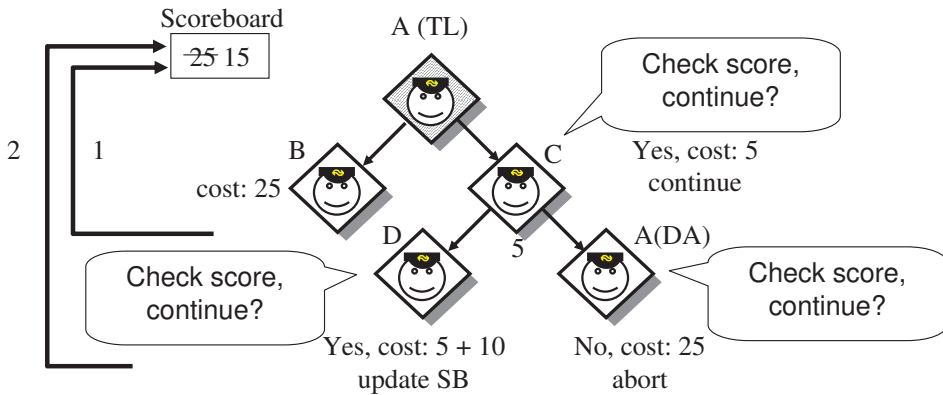
Figure 5.6. The task takeover process

the cost function to accurately discriminate between "better", "similar" and "worse" solutions.

5.2.6 Analyzing Routes

In this section, the agents involved in determining the feasibility and impact of schedule changes on driver schedules are discussed. When a driver-agent intends to take over a task, or set of tasks, he needs to determine how to get from his current location to the station where the task(s) starts and he needs to determine how to get back from the station where the task(s) ends. This means that he needs to determine a route or path through the available tasks, preferably including as much as possible the original tasks of his current duty. He has to make sure that the route is also valid in time, he needs to arrive before the start time of the task(s), and he can only return after the task(s) ends.

This rather computationally expensive functionality is deliberately separated from the driver-agents task exchange capabilities to enhance performance and ensure cleaner division of responsibilities. The route-analyzer-agent (RAA) is the central point of contact for the driver-agents which can request to determine routes; distributed network-agents (NA) support the RAA in actually computing the routes.

Route-Analyzer-Agent

Requests for route calculations from the driver-agents are handled by a RAA. A request consists of a duty and a conflict. The duty is the current duty of the requesting

driver-agent. The conflict consists of one or more tasks to take over from another driver-agent. The answer returned by the RAA can either be feasible, feasible conditional or not feasible. "*Feasible*" indicates that it is possible to add the conflict to the duty of the driver-agent without introducing a new conflict. In addition to the conflict, one or more passenger tasks (tasks to/from the conflict) may be added to the duty. If it is possible to add the conflict to the duty of the driver-agent, a new conflict is introduced in the process, and the answer is "*conditional feasible.*" In this case, the new conflict may not contain any tasks contained in the old conflict. When it is not possible to add the conflict to the duty of the driver-agent at all, the answer will be "*not feasible,*" meaning that this agent cannot help in solving the problem.

The RAA attempts to determine the correct answer to a request without having to take into account the detailed current state of the rail network. To this end, the RAA performs three steps. First, a check is performed to determine if a negative answer can be given quickly. After this, the request is compared to a history of previously received requests. If no answer is found, the requests are distributed over the available NAs for detailed examination. Below, these steps are described in more detail.

Step 1: Sanity Check: In a large number of cases ($\tilde{5}0\%$) it can be easily determined that a request is not feasible. Consider, for example, a request where a driver-agent's current location is Rotterdam and tasks to take over are in the vicinity of Groningen, and take place within 15 minutes. The distance between Rotterdam and Groningen is more than 200 kilometers. This clearly is an impossible takeover. The RAA uses an origin-destination matrix with lower bounds on the travel time between all stations. These lower bounds are static and calculated beforehand by taking into account only the track kilometers in the network and maximum driving speeds of the available train-units. If the lower bound is larger than the available time we know it is not possible to find any route.

Step 2: Request History: The RAA retains all calculated routes in memory. If the same request is received more than once (possibly from different driver-agents), the answer is retrieved from this history (about 5% of the remaining requests). Furthermore, if a previous request with a wider time-interval for the same route resulted in "not feasible", the route-analyzer-agent can conclude the current request also results in a "not feasible" answer. This history is reset when changes on the rail network or in the actual timetable occur.

Step 3: Send to network-agent: If no relevant requests are found in the request history, the RAA sends the request to one of the NAs and returns the thus obtained

answer to the driver-agent.

Departure	Destination	Departure time	Arrival time
Rotterdam	Utrecht	7:45	8:24
Utrecht	Woerden	9:08	9:22
Woerden	Rotterdam	9:32	10:08
Rotterdam	Maassluis	10:58	11:21
Maassluis	Rotterdam	11:29	11:52
Rotterdam	Eindhoven	12:17	13:30
Eindhoven	Rotterdam	14:02	15:15

Table 5.1. Example duty for a driver-agent

As an illustration, consider a driver-agent with a duty as shown in Table 5.1. This driver-agent wants to know if he can take over a task from another agent. This task departs at 10:51 from Den Haag (The Hague) and arrives at 11:10 in Gouda (not shown in Table 5.1; this driver-agent is not a team leader, i.e., not directly affected). The driver-agent sends this request to the route-analyzer-agent when the actual time is 9:00. The RAA determines that the driver-agent's current location is Utrecht. The minimal time to get from Utrecht to Den Haag is 35 minutes. In this case, the driver-agent has 111 minutes available (i.e., from 9:00 to 10:51) so a route might exist. Similarly, the RAA concludes that a route from Gouda back to Rotterdam could also exist. The RAA concludes that further examination of the request is necessary.

When all NAs are unavailable the RAA maintains a priority queue of requests to send to a network-agent. For each request, the RAA assigns a prediction-value and keeps the queue sorted according to this value. When a NA becomes available the RAA sends the request with the best prediction-value. The prediction-value represents an expectation by the RAA of how well the conflict can be fit into the duty. This is the weighted sum of:

- Lengths of the train driver duty tasks in the conflict time interval.

- Lower bounds on the travel time from the current location to the start of the conflict, and lower bounds on the travel time from the end of the conflict to the base.

These items are determined by an initial analysis of the outcome of a couple of thousand requests, where the weights were determined by regression analysis. For example, it is straightforward to understand that if a driver-agent has a lot of work

within the time interval of the conflict he will surely have to send out a new conflict
if he takes over the current conflict in his request. Similarly, agents that are located
closer to a conflict are more likely to take over the conflict in an efficient way. An
important factor contributing to the success of this scoreboard mechanism is a first
value being published as soon as possible. Sorting the requests this way helps to find
good solutions more quickly. Once a good solution is found in a team, the scoreboard
is seeded with this first score, activating the score-board mechanism.

Network-Agents

The NA maintains knowledge of the current timetable, including all disruptions and
delays. In the system, at least one NA has to exist, but possibly more if needed
to improve computational performance. NAs can easily be distributed over multiple
platforms (no cooperation is necessary and the data can be replicated). Based on
this timetable the NA can determine, by using a shortest-path algorithm, if a route
between two stations exists, and if so, which route.

The objectives in this process are to (i) maintain as much of the original duty of
the driver-agent as possible, and (ii) to arrive at the destination (i.e., the conflict's
location) as soon as possible. To illustrate this, consider again the example of Table
5.1, continued in Table 5.2.

Departure	Destination	Departure time	Arrival time
Rotterdam	Utrecht	7:45h	8:24h
Utrecht	Woerden	9:08h	9:22h
Woerden	Rotterdam	9:32h	10:08h
Rotterdam	**Den Haag**	**10:14h**	**10:40h**
Den Haag	*Gouda*	*10:51h*	*11:10h*
Gouda	**Rotterdam**	**11:16h**	**11:38h**
Rotterdam	Eindhoven	12:17h	13:30h
Eindhoven	Rotterdam	14:02h	15:15h

Table 5.2. New duty for driver–agent

The NA attempts to find a route from Utrecht to Den Haag between 9:00h and
10:51h. The fastest way to get to Den Haag is a direct connection which departs
from Utrecht at 9:03h and arrives in Den Haag at 9:51h. This means a route actually
does exist. But, there also exists a better alternative for this driver-agent: The NA
also finds a direct connection between Rotterdam and Den Haag departing at 10:14h
and arriving at 10:40h. This way the train driver can still perform the tasks of its
own duty until 10:08h and be in time for the task exchange task.

Finally, the NA is able to find a task from Gouda to Rotterdam, so the driver-agent can also perform the last two tasks of its own duty. This means, if the driver-agent wants to take over the requested task, its new duty will be as shown in Table 5.2. The *italic* task is taken over by this driver-agent, while for the **bold** tasks, the driver takes the train as a passenger. In this case, the driver-actor can no longer perform the task from Rotterdam to Maassluis (10:38h–11:21h) and from Maassluis to Rotterdam (11:29h–11:53h). This is the "conditional feasible" answer the NA will return to the RAA, which forwards this to the driver-agent.

Note that this change to the driver–agent's duty also means that the 50 minute break in Rotterdam from 10:08h until 10:58h no longer exists. The driver–agent will take this into account when examining the schedule changes (Δ); the NA only assesses if a route is possible, not whether the route is desirable.

In case the conflict can be fit into the duty (as specified in the request received by the NA), the returned answer consists of the necessary duty adjustments and a set of tasks which can no longer be performed. This set of tasks can be empty; in that case the conflict can be fit into the duty without introducing a new conflict. When the conflict cannot be fit into the duty ("failure"), the NA returns "no duty adjustments nor a set of tasks."

5.3   Implementation

For the implementation of the research system, the Cougaar agent framework (see Helsinger et al. (2004)) is used. Cougaar has initially been developed for logistics operations and provides a useful Java-based agent platform with emphasis on stability. Cougaar provides an agent model based on plugins, allowing for clean separation of functionality within agents. Furthermore, Cougaar allows for implementing services, which can be made available to agents running on a Cougaar node. Cougaar agents use a distributed blackboard for storing information and communication purposes.

To start the system a bootstrap-agent dynamically instantiates the driver-agent population and other main agents. This allows for a flexible startup process, enabling for example dynamic instantiation of the agents across a multiple-node configuration.

In order to run realistic scenarios, a dataset containing train activities and driver tasks for a full day has been provided by NS. The dataset is distilled from multiple data sources, is stored in a MySQL database, and is made available to agents through a SQL-Service. The database is to be connected to real-time disruption information for more continuous experimentation of the research system.

5.4   Computational Results

This section provides an overview of our findings. The system is able to find solutions for relatively large disruptions, using sizable driver–agent populations. To illustrate this, results of two relatively complex example scenarios are presented.
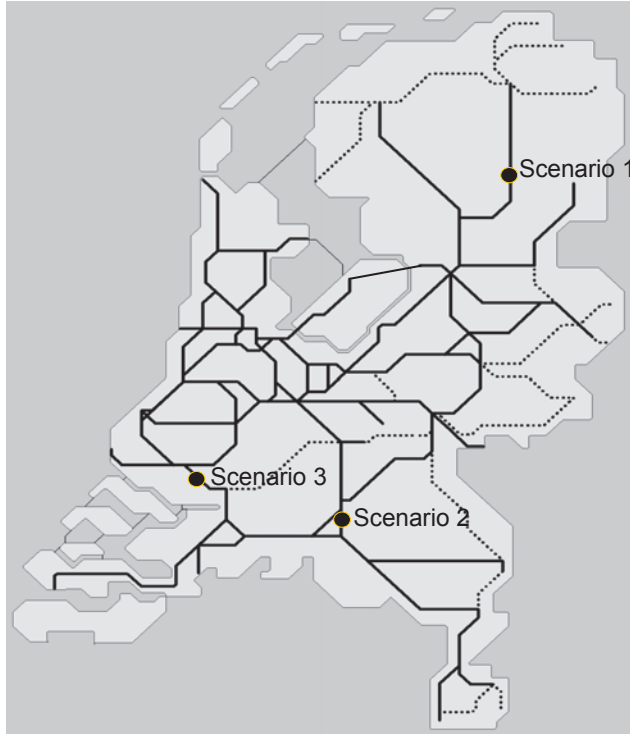
Figure 5.7. Simplified Dutch railway network and location of disruption scenarios.

Figure 5.7 provides a simplified overview of the Dutch railway network, in which the locations of the disruption scenarios are indicated. Solid lines denote the network responsibilities of NS, dotted lines are handled by other operators.

The first scenario consists of a complete blockage between stations Groningen and Zwolle from 16:00h to 17:00h. The second scenario consists of a complete blockage at station Vught from 15:15h to 16:15h. The third scenario consists of a partial blockage at station Rottedam Lombardijen, starting at 18:10h, and lasting the remainder of the evening. This third scenario will be discussed in Section 5.4.3.

The results discussed in this section are evaluated on the number of driver–agents involved in the final solution, as well as the number of spare drivers and additional

overtime introduced. The aim of the analysis of the two scenarios is to assess scalability and behavior of the system.

The number of messages exchanged between agents provides an indication of the performance of the system. The route-analyzer-agent together with the network-agents comprise the computationally most expensive part of the system; the number of routes calculated provides an indication of the effort spent. In addition, calculation times are provided. These times are indicative only (although they do indicate that the system outperforms a single human dispatcher), as the system has not been (re-)engineered as a real-time operational system.

5.4.1 Scenario 1: Blockage Groningen-Zwolle

For scenario 1, the number of canceled train services due to the blockage is 11, which leads to 11 driver–agents to act as team-leaders. Table 5.3 shows the results for the first scenario of various runs with different driver–agent populations.

run	# driver–agents	spare driver–agents	# team members	minutes overtime	# spare driver–agents in solution	avg time per team
1	59	0	21	237	0	32s
2	59	1	14	0	1	06s
3	87	0	14	0	0	08s
4	87	4	14	0	0	07s
5	135	4	14	0	0	10s
6	183	4	14	0	0	18s

Table 5.3. Results Scenario 1

In run 1, all 59 selected driver–agents are located near the disruption at the time of the disruption: In this case, a solution can be found. This solution could be improved by including a spare driver–agent, as is shown in run 2. In run 3, there are 28 additional driver–agents included that are located near the disruption, but after the actual disruption itself has already been solved. This also improves the solution found, as driver–agents now have the possibility to exchange tasks forward in time until later in the evening, when more capacity is available.

In this case, adding extra spare driver–agents does not contribute to a better solution. It does, however, help to find the same solution more quickly, as is shown in run 4. This can be explained by the scoreboard mechanism: Even though the spare driver–agent is not chosen in the final solution, these driver–agents publish a

value on the scoreboard early in the process, eliminating many, more costly solutions at an early stage.

In run 5, 48 driver–agents are added which are located relatively far from the disruption at the time of the disruption. None of these added agents contributes to an actual solution. This is detected quickly by the system: The final solution remains the same and the time needed to find the solution increases only marginally.

In run 6, 49 driver–agents are added that are located relatively close to the disruption. These agents do not contribute to an improvement of the final solution but did participate in the team-formation process. Therefore, the calculation time for this run increased substantially as opposed to the previous runs.

Run	# messages RAA	# routes calculated	# messages sent
1	41.314	13.920	398.000
2	4.826	1.944	52.250
3	6.943	1.952	73.500
4	4.893	2.033	61.500
5	8.465	2.390	108.750
6	14.182	4.171	167.000

Table 5.4. Communication statistics for scenario 1.

As shown in Table 5.4, a large amount of communication between the agents was needed to find the final solution for run 1. Because the final solution is not a very good solution in terms of overtime and number of participating agents, a large number of alternatives have to be considered during the negotiation process in order to conclude that a better solution does not exist. In run 2 where the final solution has no overtime, a large number of alternatives which were explored in run 1 can now be eliminated beforehand due to the score-board mechanism.

This leads to much less communication. In runs 3, 4 and 5 there are more agents in the system, which leads to more communication; in these cases most of the extra requests for routes from the driver–agents can be handled by the route-analyzer-agent without sending them to the network-agents. Because of this, the calculation time increases only marginally. Only in run 6 the network-agent has to process almost twice as many requests, which leads to the increase in calculation time.

5.4.2   Scenario 2: Blockage Vught station

For scenario 2, the number of canceled train services due to the blockage is 14. Table 5.5 shows the results for the first scenario of various runs with different driver–agent

populations.

Run	# driver–agents	spare driver–agents	# Team members	Minutes overtime	# spare-drivers in solution	avg time per team
1	33	0	20	501	0	19s
2	33	2	19	386	2	15s
3	106	2	28	121	2	39s
4	106	10	26	0	4	37s
5	157	10	26	87	3	44s

Table 5.5. Results for scenario 2.

In the first run, only the driver–agents who are located close to the disruption, at the time of the disruption, were included. These agents were able to find a solution for the disruption, but they needed to introduce a lot of overtime. After including two spare driver–agents in run 2, the system was able to find a better solution. In run 3, more agents that are located relatively close to the disruption were included. This resulted in a more acceptable solution, but also in more than twice the calculation time. Including all available spare driver–agents in the neighborhood of the disruption improved this solution further. In this case, 4 spare driver–agents were actually used in the final solution, the others helped to find the solution more quickly.

In run 5, 51 driver–agents are added that are located further away from the disruption at the time of the disruption. In this case, there are 87 minutes of overtime in the final solution where there was no overtime in the solution of run 4. This can be explained by the fact that the solution found in run 5 contains one less spare driver–agent in the final solution.

Run	# messages to RAA	# routes calculated	# messages sent
1	20.190	9.246	190.841
2	20.056	8.260	187.033
3	46.012	10.136	431.528
4	33.741	11.421	341.316
5	37.551	12.561	388.892

Table 5.6. Communication statistics for scenario 2.

Table 5.6 shows that when a relatively good solution can be found early in the negotiation process, the score-board mechanism is able to reduce the total amount of communication needed. In run 4, the total number of requests from driver–agents to the route analyzer-agents decreased with almost 30% and the total number of

messages decreased by more than 20% compared to run 3. The number of requests processed by the network-agent increased slightly, because additionally the routes for the spare driver–agents had to be calculated. In run 5, there was more communication needed than in the fourth run, but still less than in the third run.

As shown in the results discussed above, the score-board mechanism helps to find good solutions quickly, in cases where a good solution exists. In cases where only solutions with high costs exist (i.e., complex solutions with many task exchanges) the scoreboard is not seeded with relatively low-cost values, leading to the evaluation of almost all alternatives in the search-space, before concluding that the best solution has very high costs. A possible approach to solve this problem could be to allow partial solutions of a conflict to be published on the scoreboard. A partial solution means that a conflict is moved sufficiently forward in time to be solved at a later point in time. The reasoning behind this that it is better to find two partial solutions quickly, possibly having little higher costs, than wait too long for a complete solution.

5.4.3   Comparison MAS versus CGDDS

We have made a small study to compare the results of the agent based algorithm Multi Agent System (MAS) presented in this chapter and the work of Potthoff et al. (2010), called CGDDS (Column Generation with Dynamic Duty Selection). We have developed a set of cases and computed solutions for them with both approaches. We have used the cases presented earlier in this chapter (scenario 1 and scenario 2), and added a third scenario.

Scenario 3 represents a blockage of 3 out of 4 tracks at station Rotterdam Lombardijen. This disruption starts at the beginning of the evening (18:10h) and remains for the rest of the evening. Some trains can still be operated, but 58 duties need to be recovered, due to the canceled trains.

There are a few problems in comparing both approaches. First of all, MAS and CGDDS are not tuned on performance issues. We are aware that performance can be improved by making the code more efficient. During the project, the focus was on the quality of the plan and not on the computational performance, which was sufficient for the purpose of the research. Next to that, the cost functions are not the same. For example, CGDDS does not include costs for overtime (at the moment of comparing both algorithms), where MAS has a large penalty for every minute of overtime. MAS, especially the scoreboard algorithm, uses these costs to differentiate solutions during the solution process. These additional costs reduce the computational time a lot, are essential for the system, and we cannot set them to zero for comparison with

the results of the CGDDS algorithm. The other way around, there are easy ways to implement these extra cost function elements into CGDDS, but this has not been implemented yet. Therefore, we cannot compare the value of the cost function of the algorithms, but look at other indicators. This implies that the results of this comparison are indicative, current differences in both approaches make it impossible to compare them objectively.

	Scenario 1		Scenario 2		Scenario 3	
	MAS	CGDDS	MAS	CGDDS	MAS	CGDDS
spare duties	4	2	5	2	6	3
normal duties	1	0	8	0	4	1
overtime	97	97	68	312	357	59
computation time	149	534 (5)	948	405	583	516 (30)

Table 5.7. Comparison MAS versus CGDDS

Table 5.7 shows some differences in the solutions generated. The row "spare duties" shows the number of spare duties that are used. In all three cases, CGDDS used less spare duties. The column "normal duties" shows the number of normal duties that are changed, on top of the duties that were directly affected by the disruption. Again CGDDS provides better solutions on this indicator. The third row, "overtime" shows the amount of overtime in minutes of work. Here, it is not clear which algorithm scores better. As stated before, there was no mechanism in CGDDS to penalize overtime, and it can be expected that results can be improved if a penalty is included. Overall the quality of the solutions of CGDDS are better for the three scenarios.

The final row indicates the computation time. It shows that CGDDS is quite stable in the computation time, and MAS is faster for the smaller scenario 1 and slower for the larger scenario 2. It seems to be more sensitive to the size of the problem. Between brackets, the computation time is shown after adapting the CGDDS algorithm a little, making the neighborhood search somewhat smaller. This leads to the same solution, but in much shorter time. This improvement was studied in more detail in the work of Potthoff et al. (2010), but the results were not available at the moment of performing this benchmark study.

Based on this benchmark, NS decided to implement CGDDS in a production system, and developments of MAS were halted. The fact that CGDDS is based on the techniques described in Chapter 3, which are well known at NS, helped in making this choice. CGDDS became an extension of the CREWS system, provided

by the software vendor SISCOG, which was also used to improve the crew scheduling process.

5.4.4   Discussion

In the real-world domain of train crew rescheduling in the Netherlands, we presented an actor-agent based approach to (a) support human dispatchers and (b) accommodate individual train crews' preferences. The previous sections provided an outline of the task exchange team-configuration process including the formation protocols used to regulate the process and the role of the various rescheduling constraints. Summarizing, two multi-agent sub-systems have been co-developed in the system: The crew rescheduler and the route-analyzer.

The system aims to find a balance between optimizing for performance, quality and clarity of solutions. With respect to performance and quality, good results have been obtained. An important factor contributing to the clarity of the solutions is the fact that the basic principle of task exchange teams is straightforward and resembles in many ways the rescheduling process human dispatchers use.

The presented actor-agent approach is based on cooperative agents, to solve the crew rescheduling problem. This means that they share all information required to solve the problem. However, the problem is solved sequentially for each driver-agent that has a problem, and not integrally as with the CGDDS algorithm. The system shows much similarity to a local search approach, with less focus on the global optimum. This is compensated by the RAA, who has a global perspective on the requests and the scoreboard mechanism, which prevents many unnecessary searches. This, however, mainly leads to shorter computation time and not to better solutions.

The solution is optimal per task exchange team, but not necessarily for the complete problem. A choice made for one team can influence the potential solutions for the remaining teams. This is not taken into account while selecting the solutions. As shown, this leads to sub-optimal solutions for larger cases, where CGDDS gives better results. For smaller cases, this effect is shown to be limited, and the system gives very good solutions, and has a potential advantage over CGDDS. On the other hand, for smaller cases, the human dispatcher is usually able to find solutions and the need for support is not yet apparent.

In a dynamic environment, with a lot of uncertainty on the information that is input for the algorithm, solutions can become infeasible due to changes in reality. In this case, the agent approach could be in favor because it generates partial solutions (for each driver-agent) that can be used. For instance if the disruption time is

reduced, e.g., the blockage is cleared earlier than expected, one could remove the changes for the driver-agents that were expected to have an infeasible duty, but in fact have a feasible duty. In our agent approach, however, driver-agents participate in solving problems of other driver-agents based on the assumption that the trains during the disruption are canceled. Canceled trains create gaps in their duties, which enables these driver-agents to take over tasks from other driver-agents. This means that, although the problem is solved per affected duty, when a train is not canceled as expected, the original duty might be not valid anymore because the driver-agent already accepted work from another driver-agent.

The advantage of having partial solutions is that they can be applied before the complete solution is found, which makes computation time less important. The system first generates solutions for the most urgent problems, the problems later on the day can be solved after that. The CGDDS algorithm has proven to be fast enough to solve the complete problem in a short time, so there is no need to partition the problem in time. Once new information arrives, and the plan becomes infeasible, one can simply apply the algorithm again and solve the new problem. Both algorithms present total solutions before the new information generally arrives. Because CGDDS delivers better solutions, the starting point for a new computation will be better as well. Therefore, we think that CGDDS will at least perform as well as MAS, probably better, in case of dynamic updates on the situation at hand. We did not evaluate this with scenario studies.

5.4.5 Future Research

In this section, a number of possible changes to the system are discussed, aimed at improving the performance of the solution finding process and quality as well as clarity of the solutions found. With respect to performance, the main factor impacting performance is the route finding process. In addition to the mechanisms described in this chapter, the route finding capacity of the system can be scaled up relatively easily, by adding more network-agents and additional supporting hardware. With respect to solution quality, the effectiveness of the protocols influencing the team formation process is the determining factor. Although the protocols described in this chapter could be evaluated and improved, NS decided to halt the research process on this.

One of the key principles in CGDDS is the selection of the duties that should be considered for computing a solution. This seems to work fine for medium to large cases. For smaller cases, like a delayed train causing an inconsistent duty, this

selection mechanism can be improved. Some practical experiments show that MAS is very efficient in these cases, because it does not depend on an initial selection of duties. Future research on CGDDS could focus on integrating techniques from MAS into the duty selection mechanism of CGDDS.

During the development of MAS, we also performed some initial research on extending the system for crew rescheduling with agents representing train stations, thereby introducing a different view on the railway network than that of the driver–agents, who are mainly concerned with their own duties. Station-agents aim to improve the robustness of the resulting crew duties against future disruptions. When facing new disruptions, a robust set of duties gives a larger probability of feasible solutions, or solutions in which all trains can be operated in case of new disruptions. For instance, if spare drivers are used to drive trains, then in case of a new disruption there are fewer drivers available at a station to drive trains for which no other driver is available. The Station-agents monitor the amount of spare-drivers and the amount of free time (time between two consecutive tasks of a driver).

The proposed extension is to measure such robustness by monitoring the "inventory" of train crew at interesting (i.e., large) stations during the day, i.e., the station fitness. When this fitness becomes too small, the Station-agents start a similar solution process to change the duties of the crew, with the aim that this fitness reaches a certain level. A spare driver can be repositioned from one station to another station where there is a higher potential that he is needed. For instance at the end stations of a blockage, it could be wise to have additional spare drivers because there is a reasonable chance that the duration of the blockage will be longer than expected. Future research could study this idea in depth.

5.5   Conclusions

The crew rescheduling system MAS is a real-world application of an actor-agent system. It has shown that it can handle large rescheduling problems. The calculation times indicate that the system easily outperforms a human dispatcher in the quantity and thoroughness of the solutions found. Parallel to the development of this algorithm, an alternative approach (CGDDS) was developed that outperforms this algorithm. Nevertheless, elements of this actor-agent approach can be useful in supporting real-time dispatching in the future.

In Section 5.1, we partly answered research question 3: *"What are commonly used methods for supporting the Crew Management process?"*, by elaborating on an alternative approach based on Column Generation and Lagrangian Relaxation, and discussing related work in applying multi-agent techniques. Both can be used for solving the Crew Rescheduling Problem.

To answer research question 4: *"How to apply these methods to the Crew Management process?"*, we have shown how the actor-agent methodology can be applied to give good solutions for the Crew Rescheduling Problem. However, we have found that another approach gives better results, and this other approach has been introduced in the daily operations of NS.

Chapter 6

Applying Operations Research in Practice

The previous chapters focused on developing models and solutions for Crew Management and computational results. This chapter focuses on the organizational change process, related to the introduction of decision support systems (DSSs). This includes the research and development process, the implementation of new working methods and the impact of using DSSs in practice. First, we will introduce a framework for organizational change, on which we will project the practical work done at NS, illustrated by some examples. We will present the goals and activities of our work and, at the end, we will summarize the lessons learned and relate them to lessons identified by other practitioners. We show how to introduce advanced decision support systems successfully, resulting in a large impact on the performance of the organization.

6.1 Framework for Organizational Change

Cawsey et al. (2011) present an adapted framework for organizational change depicted in Figure 6.1. The framework is based on work of Beckhard and Harris (1987). The first phase of the framework starts with an initial organization analysis to determine a well-grounded sense of what needs to be changed and how, given that there is a problem. Second, based on an evaluation of the need for change, a sound rationale for the change and a compelling vision of a possible future is developed. This to unfreeze organizational members. Third, based on a description of the present state and a definition of the desired (future) state, a gap analysis is performed, identifying nec-
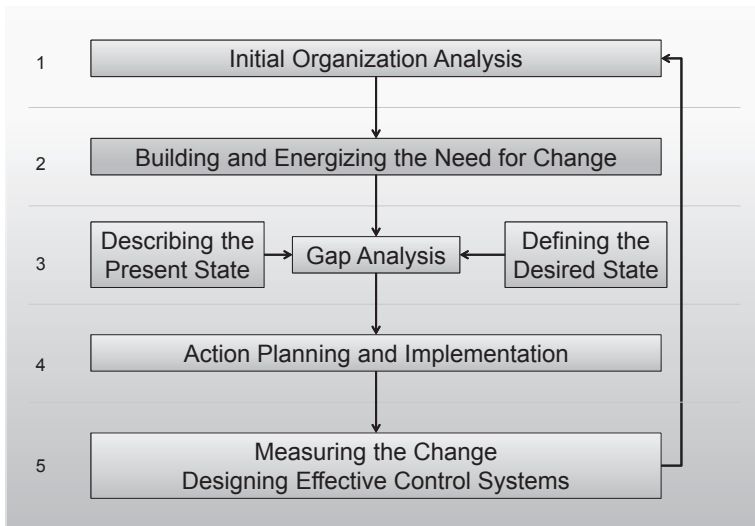
Figure 6.1. Framework Organizational Change

essary formal structures for change, the informal organizational aspects, the change recipients and the change agents. The fourth phase is the action and implementation. Here, an activity plan is developed, including contingency and communication planning. The transition is managed, and at the end there is a planned moment of celebration and after-action review. The fifth and final phase is the measurement of the achieved change over time. After this, the process of change starts again, if necessary.

We will fit the change process NS went through on the presented framework. Figure 6.2 gives a global outline of the initial state and the change process considered in this chapter. This change process ends in 2003, when the first iteration of the change process as presented in Figure 6.1 ends. Successive iterations of the change process are not included in the figure.

External developments

Internal developments

1980

Neoliberal reforms

1990

Initial state

Question subsidies NS — Limits current planning method reached

Appointment new head of Logistics

Effort put on best practices

Crews project started

Start privatizing NS

Lovers Rail — OR-Expert hired

NS split in Business Units — Crews A*-Algorithm Unsuccessful

Expected increase in passengers
=> efficiency gain required

NS get concession main lines — Introduction of Crews system

Regional lines tendered

"Circling-the-Church" rules developed

Change process

2000

Outplacement ProRail[1] — Introduction of TURNI

National Strikes

"Circling-the-Church" rules operated

Replacement NS management — "Sharing-sweet-and-sour" rules developed

"Sharing-sweet-and-sour" operated

2003

[1] At that time there were 3 organisations (Railinfrabeheer, Railned and Railverkeersleiding), which became ProRail in 2003

Figure 6.2. Timeline Change process

   To support the framework, Cawsey et al. (2011) formulated several critical questions. Because this chapter discusses the performed change process in retrospective, these questions are reformulated to:

1. What was the environment telling you prior to, at the beginning of, during, and following the implementation of the change?

2. Why was change needed? Who saw this need?

3. What was your purpose and agenda? How did this purpose project to a worthwhile vision that went to the heart of the matter?

4. How did you implement and manage the change?

5. What have you learned about change and how can you remember it for the future? How can you pass on what you have learned?

6. Once the change is completed, what comes next?

   In the remainder of this chapter, we fit the change process to the described framework and answer the questions. Each of the following sections corresponds to one phase in the framework.

6.2   Initial Organization Analysis

In this section, we answer the first main question, "*What was the environment telling you prior to, at the beginning of, during, and following the implementation of the change?*", and three sub-questions suggested by Cawsey et al. (2011):

1. What were your customers or clients telling you?

2. What were your competitors doing, how were they responding to you, and how were you responding to them?

3. What was the broad environment communicating to you about future economic, social, and technological conditions and trends?

   NS as a railway operator has two types of customers: the passengers who use the train services, and the government who is responsible for public transport. In the early nineties, the government started to question the subsidies given to NS, which was fully state-owned as most railway operators at that time. After the neoliberal reforms of the eighties, the government wanted to reduce the subsidies to companies

in general and also questioned the way the subsidies were spent by NS. The government decided to start privatizing NS. The idea was that rail transport should be economically viable, and that competition could improve the efficiency of the rail industry, and would eventually lead to profitable rail transport, without the need for high subsidies. This is, even currently, not fully completed because the government is still the sole shareholder, but the idea was that NS should be fully privatized and independent of subsidies.

In the airline industry, liberalization of the US markets in the eighties was very successful, leading to additional services for lower costs. This successful liberalization not only influenced national politics, but also the European Union passed a directive. This directive prescribed, among other things, to split the management of the rail infrastructure from train operations, at least in the bookkeeping. The Dutch government chose a more strict division. The plan of the Dutch government entailed that the government would remain responsible for the rail infrastructure, while NS would commercially provide the rail transport. When part of the services would prove to be economically unviable, the government would only subsidize that specific part.

The part that was split from NS and became responsible for the rail infrastructure is currently called ProRail. Within NS, several business units were founded, where NS Reizigers became responsible for operating the trains within the Netherlands. Other large business units were NedTrain (responsible for rolling stock maintenance) and NS Stations (responsible for commercial activities at the stations). The freight division NS Cargo became part of Deutsche Bahn (DB), and is currently known as DB-Schenker Rail.

Large railway companies like DB were prepared to enter the Dutch rail market. In 1996, one of the first initiatives to enter the market, came from Lovers Rail. They operated two railway lines but failed to commercially succeed. NS reacted to this initiative by increasing the frequency of trains, making it difficult for the competition to run trains on "their" lines. After Lovers Rail ended their activities, it was agreed that NS would be the sole operator on the main lines. Later on, a few regional lines were tendered and foreign competition entered the Dutch rail market. For instance Arriva, a full daughter company of Deutsche Bahn, operates lines in the north, east and western part of the Netherlands.

Predicted future economic and social trends indicated a large increase in mobility and, therefore, required NS to prepare for increasing services. This increase in mobility would lead to congestion on the road and an increase of CO_2 emission. Rail was considered to be the preferable alternative to facilitate the growth in mobility.

Together with the required cost reduction, this implied a more efficient usage of the available resources. Technical developments enabled the increased use of advanced IT systems in organizations. Computer hardware increased in processing capacity and software was developed to support planning processes. In similar industries, this became common practice.

For example, the airline industry saw the great potential in using Operations Research techniques, originating from research in the military defense industry. There was already a heavy competition between the airlines, due to the market liberalization in the early eighties, which resulted in focus on improving optimization techniques so they could be practically used.

Especially, techniques for crew pairing were improved due to the business need from the airline industry, resulting in advanced decision support systems (DSS). The main reason to introduce these decision support systems is to improve the quality of the plans, both with respect to customer service (improved frequencies and more robust schedules) and with respect to costs (more efficient usage of resources).

We conclude that there was a strong focus on cost reduction, and that an increase in services was expected from NS. Competition was to be introduced, and NS had to live up to expectations, and had to demonstrate that it could improve its performance. Technology was potentially an enabler in improving the performance.

6.3   Building and energizing the need for change

After the description of the broad environment of NS in the previous section, we will answer the second framework question: "*Why was change needed? Who saw this need?*"

As stated before, it was required from NS that the company should become more cost efficient because the intention was to stop subsidizing NS and it should prepare to compete with other commercial operators. The planning process plays a very important role in the performance of a railway company, which is basically a logistical organization. The planning process determines the quality of the plan which is the basis for the quality of the service delivered, and the related costs of operating it. The expected increase in the number of passengers, combined with limited resources, would result in more complex planning problems. It was clear that this would affect the planning process and the people involved. The current process would reach its limit and needed to be changed.

Cost reduction can be achieved by an efficient planning process that delivers effi-

cient plans. Most of the costs relate to the usage of the resources: rail infrastructure, rolling stock and train crew. Here, rail infrastructure is the most expensive, followed by rolling stock and then train crew. More efficient plans in this context mean that the product is delivered using fewer resources, at the same quality level. The rail industry historically was not focused on creating efficient plans, and it was reasonable to expect that the efficiency of the plans could be improved a lot.

Although less contributing to the costs, the large number of planners that created the schedules was to be reduced as well. A couple of hundreds of planners were, and still are, working at the department of Logistics. With new technical support, like decision support systems, the expectation was that this number could be reduced significantly, together with an increased efficiency of the plans.

Not everyone at NS was aware of the great potential improvement a DSS, in combination with a new planning process, could bring and why this change was needed. Cawsey et al. (2011) present five approaches to create awareness for the need of change: (1) Create a crisis, increase awareness that crisis conditions exist, or communicate that a crisis is on the horizon; (2) Develop a vision that creates dissatisfaction with the status quo in the organization; (3) Find a champion-of-change leader who will build awareness of the need for change and articulate the vision for change; (4) Focus on common or superordinate goals; (5) Create dissatisfaction with the status quo through education, information, and exposure to superior practices and processes of both competitors and non-competitors.

NS spent effort on getting information on best practices of competitors, non-competitors like airlines, and from science. This information helped creating dissatisfaction with the status quo and in raising awareness that NS could really improve. This helped, but was to our opinion not the main contribution to the increase of the need for change.

At NS, the awareness of the need for change was basically increased by a champion-of-change leader (being the head of Logistics), who saw the need for change and envisioned a future in which DSS played an important role. This leader was able to influence the higher management to start this change process and to allocate the needed resources. With his technical background, he understood the potential of Operations Research and was able to connect to the scientific community. Being responsible for the department of Logistics, in combination with the strong relational skills, he was the ideal person to start the change process. With the solid support of the champion-of-change leader, the change process was started in which the best practices in introducing DSS were also to be implemented at NS.

6.4   Gap Analysis

After having described the broad environment of NS and the need for change in the previous section, we will perform a gap analysis for the changes needed. This analysis is based on a description of the initial state of the scheduling process, before NS introduced DSSs for the planning processes, and the desired state. This section answers the third framework question: "*What was your purpose and agenda? How did this purpose project to a worthwhile vision that went to the heart of the matter?*"

We will mainly focus on the crew scheduling process, because this is the first topic of this thesis, where we went through a complete cycle of the change process. We consider this part of the planning process to be representative for the other parts of the planning process, and will discuss the other topics in Section 6.7.

6.4.1   Initial State

In the early nineties, planning the crew duties was mainly a manual task. The construction of the crew duties was the responsibility of the department of "Planning Rijdend Personeel" (planning train personnel), within the so-called "Jaarplan" (year plan) organization of Netherlands Railways. In 1996, 22 full time employees worked at this department. IT-support was limited, and planners had to compose the duties completely by hand. In those years, IT-support was a character based registration system. This system supported a validation of the duties against the timetable, to check whether the train activities were consistent and whether all trains were provided with both a driver and the right number of conductors. These checks were all performed off-line and reported in printed reports. The planners needed to correct all inconsistencies by hand.

For a new timetable year, the first phase of scheduling started with creating a pre-design. A few planners started with the distribution of trains over the different depots. This was done by using colored pencils and large papers containing the graphical presentation (time-space diagrams) of the timetable. The planners literally colored the train tasks on these diagrams, and each color represented a specific depot. The assigned train activities were then manually plotted on duty charts. Then the planner started to puzzle and to optimize the duties. This was done by moving the tasks on the paper using pencils and erasers. One can imagine that this is a time consuming task. This pre-design phase could take several months, up to half a year.

This pre-design was based on an initial version of the timetable. During the duty scheduling process, the timetable and the rolling stock schedules were completed.

Therefore, the crew duties needed to be adapted to the final changes to the timetable, after which the rosters were created. In total, the process of completing the crew schedules took about half a year, on top of the completion of the pre-design phase. In total, the construction of a new crew schedule took more than a year.

In the early nineties, NS started a large IT program called "Transport By Train" (in Dutch: Vervoer Per Trein (VPT)), resulting in a system with the same name, built by the internal IT department. This project delivered support for the planning and dispatching process by registration, validation and distribution of the plan. With the continuous growth of the company, due to an increasing number of passengers, the planning and dispatching process had become very labor intensive, and the amount of communication between planners, on several locations in the country, increased a lot. This made the process time consuming, inflexible, inefficient, and mistakes were easily made. The systems delivered by the VPT project helped in improving the planning process by the described support.

While creating a plan, there is a trade-off between efficiency, expected operational performance and the attractiveness to the crew. In general, the crew members try to influence the planning process in order to get the most attractive set of duties possible. This influence can be informal by discussing possibilities with the planners or can be formalized by labor rules being the result of negotiations between the company and the labor unions. In the early nineties, there was no technical support for performing what-if studies. Requests from the unions were evaluated based on an expert judgment of the (senior) planners. They estimated the consequences of a proposal, in terms of efficiency and expected operational performance. Also, ideas of the management were evaluated by experts. These judgments could not be verified objectively and were, therefore, always debatable. As described in Appendix B, different points of view could eventually lead to serious problems in the relations between management and crew members.

6.4.2 Desired State

Inspired by external developments in IT and in related industries, the head of the department of Logistics envisioned more than a system that supported only in registering and distributing the plan. He wanted to introduce advanced decision support systems (DSS) to support the planners with optimization techniques. There were a number of people in the VPT project that supported him in this ambition.

Ultimately, a DSS generates all schedules fully automatically. This is called the "red button solution", where a user only has to push the button and all schedules are

generated. It was clear that this level of support could not be reached in the coming decade, for the complicated planning problems at NS with expected technological developments, both in hardware and software techniques.

6.4.3   Gap between Initial and Desired State

In the early nineties, the possibilities were not clear. For crew scheduling, NS expected to be able to automatically schedule several depots, which would take hours or days to compute. Current developments (see 3.6.2) enable us to automatically solve instances that are about fifty times as large as initially expected, still taking days to compute. Because the desired state was not realistic, NS focused on achieving a state where (large) parts of the planning problems are supported by DSS. NS chose a step-wise approach in which it addressed parts of the planning, which will be described in more detail in the next section. This step-wise approach included the gradual development of DSS, working methods and training of planners. This step-wise approach was also reflected in the development of the R&D department that on average hired one additional analyst every year. Section 6.6 describes the established changes, which can be interpreted as a gap analysis of the difference between the current and the initial state.

6.5   Action Planning and Implementation

This section will answer the fourth framework question, *"How did you implement and manage the change?"*, and four sub-questions suggested by Cawsey et al. (2011):

- How did you resource the change initiative?

- How did you work with the broader organization?

- How did you select and work with your change team?

- How did you ensure that you acted (and were seen to act) ethically and with integrity?

6.5.1   How did you resource the change initiative?

The work described, originated from a visionary leader that wanted to improve both the planning product as well as the planning process. He gave the research group the freedom to work at this without any explicit business case. The analysts were part

of the internal staff of the department of Logistics. In cooperation with universities, PhD-students were sponsored for performing rail related research. The budget for this was arranged by the leader, who convinced his managers that a small investment would pay-off in the long term.

6.5.2   How did you work with the broader organization?

In the beginning of the change process, NS decided to invest in advanced support for the planning of the crew duties with the introduction of the system called CREWS, see Morgado and Martins (1998), from the software vendor SISCOG[1]. In 1991 NS visited several exhibitions and discovered SISCOG and its initial crew scheduling prototype, originally made for an airline operator. SISCOG was invited to demonstrate its system to all planners of train crew at NS.

Several workshops were held in which the system was demonstrated. Many of the planners were impressed by the prototype and by the promised optimization tool that would be able to generate the crew schedules. A clear advantage of the demonstrated system was the ability to perform what-if analyses by changing parameters and business rules. At the end of the series of workshops, having support from the planners, NS decided to start a project aiming at the implementation of CREWS at NS.

The project team included representatives from the department of Logistics. They were dedicated to work for the project and worked closely with the team members of SISCOG. They met frequently, both in Lisbon and in Utrecht. These meetings were crucial in exchanging knowledge on the planning process in order to develop the required functionality.

It was clear from the start of the project that introducing a new system would also imply new working methods for the planners. These methods were developed together with all planners. They formed working groups in which they discussed how they could distribute the work in the system and which functionality they would need for working in the envisioned way. This way of working was discussed with SISCOG, and the required functionality was built in the system.

The project took several years, since the reality of NS turned out to be much more difficult than expected initially. In 1998, the CREWS system went operational and was initially mainly used as a manual planning support tool, with the first 6 levels of support: Registration of schedules, distribution of schedules, generation of

---

[1]www.SISCOG.pt

scheduling elements, graphical presentation, validation of schedules, and evaluation of schedules (see Chapter 1).

Using CREWS was a big change for the planners. Where the former VPT system was only used to register the duties, created by hand using pencils and paper, the planners had to get used to working with the system. Planners had to get used to the fact that the system showed all conflicts in the plan. At first, not all conflicts were shown correctly, but this was corrected easily in the system. Explicitly formulating the rules lead to the insight that planners were not all using the same rules. They interpreted them differently or had their own reasons to neglect them.

With the introduction of the so-called "Circling-the-Church rules"[2] in 2001 (see Appendix B), also a new set of duty planning rules was determined, which should not be violated while scheduling. The basic idea behind these rules was that buffers in the schedules should be larger, so the duties would become more robust. Later on, when the Sharing-Sweet-and-Sour rules were determined, the larger buffers were kept. The rules for variation in the schedules were changed, but the rules for the individual duties were kept. This straightforward set of duty rules ended the discussions on how to apply the rules.

Having a clear set of rules also helped in introducing algorithmic support. Planners are very creative in interpreting rules. They can sometimes find "better" solutions than the optimal solution found by an algorithm by violating a rule that is strictly obeyed by the system. In order to include this kind of solutions in an algorithm, the violation can be modeled in the system as an exception to the rule, or the rule can be modeled as an objective. In the second case, a violation is penalized in order to only allow violations when it gives a significantly better solution. At the end, modeling all these details makes the solutions much harder to explain and this raises the question whether the overall gain is worth the complexity of the model.

For example, a transfer time of 19 minutes, where a minimum of 20 minutes is the rule, can result in a solution with one duty less. However, the buffer for absorbing a train delay is reduced with 1 minute. Too many violations would cause the schedule to become less robust. It became a company rule that these rules should not be violated. The efficiency loss caused by these stricter rules was compensated by applying the optimization algorithm.

Key in the successful research and development is that the projects were carried out in direct contact with the department of Logistics and the planners who had the knowledge on the problems. They explained the problem, the attributes, the

---

[2]The official term for these rules were "Destination: Customer" rules, however this term is only used inside NS

constraints and the goals. Important is that they could also give feedback on the generated solutions. In all cases, there were many unwritten constraints and goals that were revealed when a solution was presented. The planner then addresses and explains the problems in the solution. It is important to directly show that the researchers are able to incorporate these additional constraints and goals in their models, and show that these lead to better solutions than the ones created by hand.

While introducing the system, not all planners were used to work with computers. Some of them never worked with a computer and did not know how to move a mouse-pointer or to click on objects. The management even discussed what to do with the planners who could not be taught how to use the system. Special training was given in using computers, and at the end everybody was able to use the system. A nice proof of this is one planner who was having troubles using the new system, and initially showed clear resistance to work with the system, once had to work at another department where they still used the old system. He returned, telling that he could not understand why this department was still working in this old-fashioned way. This was clearly a sign that everybody accepted the new system.

When the planners received the system and started using it, they found out that it was giving them more support than they expected. Advanced drag-and-drop functionality, search and filter options, and conflict detection enabled them to handle a much larger part of the scheduling problem. Where, in their envisioned working method, they split the national scheduling problem into regions, it turned out that a single planner could handle all duties of a certain day for the whole of the Netherlands. Key in getting the planners to adapt new working methods was their involvement with the development of these methods, from the start. This enabled them to adjust their working methods to the functionality provided by the system.

A planner does not need to know a lot of details by heart anymore, due to the support the system gives. The system will validate the schedules a planner is creating and visually show the tasks that need to be scheduled. Before the introduction of the system, a planner needed to know the route and rolling stock per task based on the train number, and needed to know which personnel depots had which knowledge. Also, he needed to know all constraints by heart. Next to that, he had to perform repetitive tasks, for instance computing a lot of time intervals to check the transfer time constraints. With the system, it became much easier to train a new planner, given the new functionality. Also, NS became less dependent on individual planners skills.

6.5.3   How did you select and work with your change team?

From the early nineties, NS started a close cooperation with universities to develop advanced decision support systems, after an unsuccessful project with a commercial software vendor to develop a timetabling system. Initially the research was performed at universities, sponsored by NS. After a first successful development of DONS (see Schrijver and Steenbeek (1993)), NS decided to hire an internal Operations Research specialist and to invest in in-house expertise. He became responsible for the research and development on rail related topics. At that same period, NS hired a specialist with both a background in IT and Operations Research. He was responsible to intermediate between the end users of the system and the external software vendors of DSS. During the years, the number of people with Operations Research expertise was gradually increased, and a Research & Development (R&D) department was founded.

The goal of having this department was twofold: first, the railway domain is very knowledge intensive and, therefore, the head of Logistics wanted to establish a research center with its own staff. Being NS employees, the researchers have easy access to their colleagues with a high level of domain knowledge. Second, NS wanted to have a strong influence on the research agenda, based on the key areas where it needed to introduce decision support.

The head of Logistics was well aware that NS should not isolate the research department from the scientific community. This was achieved by facilitating internships for Master and PhD students and stimulating staff to work part-time at a university. Having an affiliation with a university makes it easy to get Master and PhD students to write their thesis on a rail related optimization problem. This way, the department stayed aligned with the developments in the scientific community.

Cooperation with the university requires providing openness to the research results and publishing them. A possible drawback of this is that the competition can also benefit from the research performed. NS considered this to be less relevant, as long as it could maintain its strategic advantage to be ahead of its potential competitors by being able to implement the results faster because it had the required knowledge in its own organization.

After the mentioned successful implementation of the DONS system, several research projects were started on improving timetabling, rolling stock planning and crew scheduling. These projects were carried out with external partners from both the scientific community and from commercial software vendors. We will describe the research performed on the crew scheduling problem.

Before the R&D department was started, an intensive effort for eliciting the optimization goals and constraints for the crew scheduling problem was carried out. This effort was part of the project described in the previous section, where we introduced the CREWS system. First, the manual scheduling method was studied to be used for an algorithm that mimics this manual scheduling method. This algorithm was based on the so-called A^*-heuristic from artificial intelligence, as described in Morgado and Martins (1998), and implemented in the system.

The results generated by the algorithm showed that, although the system was able to generate solutions for small sub-problems, it could not generate adequate solutions for real-life instances. Analysis showed that the planners were able to provide better results because they made exceptions to the described method, based on a more global perspective on the problem to be solved. Because the implemented heuristic is basically a local search heuristic, with insufficient possibilities for adjusting the method to provide it with a more global perspective, NS decided to search for an alternative method to solve the problem.

A benchmark case was provided to several software vendors which could present their solutions. The best results were provided by the system called TURNI[3], developed by an Operations Research expert from Italy. NS purchased a license for this software to use it together with the CREWS system. CREWS was operationally used to support the planners, providing the first 5 levels of support (described in Chapter 1). The optimization part of the software, level 6 support, was not used and was replaced by the TURNI algorithm. Data interfaces were created with the TURNI system to export tasks and import generated duties in order to use both systems together in the easiest way possible.

As described in Appendix B, TURNI was successfully used in ending a conflict between NS and its drivers and conductors, in 2001. After ending this conflict, NS needed to use an advanced planning algorithm for producing the crew schedules due to the very complex set of labor-rules. This complex set of labor-rules made it impossible to schedule the duties by hand, as the planners were used to do. TURNI became the algorithm that was not only used to perform studies, but was also used to generate the yearly schedules.

---

[3]www.doubleclick.it

6.5.4   How did you ensure that you acted (and were seen to act) ethically and with integrity?

We supported the planners in the daily use of the system. We were responsible for solving problems quickly and for defining functional changes to the system. To really understand the needs of the planners, we worked physically at the same location as the planners and could therefore socially integrate with them and understand their needs. Organizational issues were handled by the management of the department. This enabled us to focus on improving the quality of the solutions and the systems support. Being consistent in this focus ensured that we acted ethically and with integrity. We never felt any personal mistrust from the planners involved.

At first, there was a lot of distrust from the crew to the schedules provided by the system. At the start of introducing the system, there was a heavy conflict between management and crew. During the conflict, the works council was allowed to hire an external consultancy firm to help them develop the new set of rules for the crew schedules. This consultancy firm was specialized in applying Operations Research techniques. They tried to create a mathematical model to study different scenarios. It turned out that this was too complicated, and when they learned about the TURNI solution, they knew that it would be much better to use this system for performing the analysis than creating a new model. We decided to help them in their task and generated the duty schedules, based on the specifications provided by the consultancy firm. This consultancy firm was responsible for formulating the requirements and for analyzing the schedules and advising the works council in selecting the right set of rules. At the end, the Sharing-Sweet-and-Sour rules were chosen. Appendix B describes this process and the achieved success in detail.

After the conflict had been ended, it was clear that the TURNI system could produce duties with the new set of rules. During the conflict, we already produced these kind of duties with the system. Still there was some distrust from the works council. They were worried that the system would only focus on making efficient duties and would prevent attractive duties from being created. To overcome this situation, the works council was allowed to continue to hire an external specialist from the consultancy firm that helped them during the conflict. His role was to validate that the system was used in the proper way, fitting the requirements of the council and to build trust on the schedules produced by the system.

6.6 Measuring the Change and Defining Effective Control Systems

In this section, we describe the effect of applying decision support techniques on the CSP. We will explain the improvements in the internal process, both in lead-time and in flexibility. We describe the effect on the planning process and on the result of the process, the plan. This section will give the answer to an additional sub-question for the fourth framework question: "*How did you monitor progress?*"

As stated, there was no explicit business case, and there even was no described final state. Therefore, there was no identified need to make detailed project plans and measure progress, and no explicit control system was in place.

6.6.1 Realized changes in the planning process

The first expected change was the improvement of the planning process. Improvement of the planning process can be measured by several performance indicators, where the most important indicators are lead-time and number of planners involved.

Using TURNI, several alternative production models were studied, and several different sets of duty schedules were generated. Performing studies at this level of detail was not foreseen with the introduction of the system. By hand, it simply took too much time to create a single plan. The system made it possible to develop these alternative schedules in a few weeks, with only a few persons using the system.

Before the introduction of the system, it would take months to create an initial plan. This would then be finalized during several additional months. Altogether, it would take about a year in lead time to create a new plan. After the introduction of TURNI as an add-on to CREWS, creating the initial plan could be done in a few weeks. The planners finalized this initial plan in a few additional weeks, with the help of the CREWS system. Altogether, the plan could be created in a few months, a large reduction of lead-time compared to the year it took initially.

6.6.2 Realized changes in the crew schedules

The second expected benefit, of the change initiative, was that the quality of the plan could be improved. In the following paragraphs, we discuss the three main quality aspects of a crew plan: (1) the impact on expected operational performance; (2) its attractiveness to the crew; and (3) the efficiency of the duties.

The first aspect, the *expected operational performance* of a plan, can be improved

in several ways. One can add more buffer time, between two consecutive train tasks, to prevent that a train has to wait for a crew member that has a delay on a preceding train. Another option is keeping a crew member on a single train as much as possible. This way there are fewer changes from crew during a duty which results in a reduction of potential disturbances. Also keeping crew members on a single line during the day is considered to be better, because disruptions will not be distributed to other lines by crew members that need to change trains.

All these elements were incorporated in the Circling-the-Church rules. The buffers were increased to 20 minutes, changing trains was limited, and crew of each depot was allocated to a given set of railway lines. In the year NS operated this plan, it resulted in the worst performance in history. It turned out that the motivation of the crew members, which were very dissatisfied with the monotonous duties, is crucial in providing the operational performance needed. This aspect will be addressed in the next section. After the replacement of the Circling-the-Church rules, NS kept the larger buffer time between two consecutive tasks, and the other elements were replaced by more variation in the duties.

The system enabled the introduction of duties based on the Sharing-Sweet-and-Sour rules, which had a strong influence on the second aspect of the quality of a crew schedule: *attractiveness* to the crew members themselves. TURNI was used successfully in ending a conflict between NS and its drivers and conductors, clearly indicating that the related set of duties were considered far from being attractive.

Finally, for the third aspect, *the efficiency* of the schedules created with CREWS and TURNI, we did not measure an improvement over the years. About the same percentage of work per duty was measured. This means that the benefits, of introducing the DSS, was allocated to improving the robustness and the attractiveness of the schedules.

6.7   Iterations of the change process

In this section, we answer the sixth framework question: "*Once the change is completed, what comes next? The completion of one change simply serves as the starting point for the next.*" We will describe how we developed and implemented several improvements in the support of the crew scheduling process. After the initial implementation of the CREWS system and the TURNI system, we did not stop improving the support for the crew scheduling problem. In this section, we describe this additional improvement in algorithmic support for the crew scheduling process. We

describe the development steps and the results that were achieved.

After NS had adopted TURNI, it was used in several projects to study the effects of different sets of rules to be applied in its crew scheduling process. For the regular labor-rule negotiations, TURNI was used for studying the effect of e.g., changing the minimum length of the meal breaks. NS also used TURNI in a bidding process, which helped NS to win a long term concession to operate trains in the Liverpool area (United Kingdom).

In the 2007 timetable, effective from December 10, 2006, per day there were about 200 trains more than in the previous timetable. All these trains require a driver and several conductors (depending on the length of the train). Since it was expected a year ahead that the crew capacity was insufficient, to operate the new timetable, further optimization of the crew schedules was necessary.

This resulted in the research described in Section 3.5, where we presented a method that improved the usage of TURNI. We showed that applying some basic partitioning techniques can have a significant added value when combined with an advanced crew scheduling algorithm. This method was automated which not only enabled creating efficient production plans, but also enabled using it for what-if scenario analysis. In the past, the scenarios were only studied for a single weekday. With this method, the analysis became more accurate because the complete week is taken into account.

Based on this improved usage of TURNI, the overall efficiency was improved with almost 2%, on top of the improvements due to the introduction of TURNI. In this way, the expected crew capacity shortage for the 2007 timetable could be reduced significantly such that the timetable went smoothly into operation. For the effect of the introduction of this timetable, we refer to Appendix C.

In the years that NS used the TURNI software in combination with the CREWS system, SISCOG started to develop its own OR based scheduling algorithms. They adopted the work described in Huisman (2007) and implemented an algorithm for crew rescheduling. This was successfully introduced to the short term planning process at NS. SISCOG and NS decided to extend their partnership and started a joint project aiming at developing an optimization model capable of solving from scratch large-scale duty scheduling problems, with results that would be comparable to the existing TURNI solution.

The project was done in two steps. The first challenge was to solve the duty scheduling problem for train drivers with a reduced number of constraints. The results of the first prototype were very promising, and a second project was started

as a continuation of the first. The second project addressed the challenge to efficiently solve in a single run for a single weekday the entire duty scheduling problem both for train drivers and conductors with all real-world constraints.

The results for the benchmark cases designed for the challenge were compared with solutions of TURNI on these cases. All new solutions were feasible, which was not the case for the reference solutions. They were between 0.12% and 2.84% more efficient. Most of the solutions were obtained in less than 12 hours of running time. Parallel computing techniques highly contributed to short computational times. The algorithm was called LUCIA (Lisbon Utrecht Crew schedulIng Algorithm).

For the timetable starting in December 2008, LUCIA was used as the tool to generate the production schedules. Initially it was a stand-alone tool, but in April 2009, SISCOG delivered a version of CREWS in which LUCIA was fully integrated. This was a major improvement for the planners, because it became much easier to generate duties with this integrated toolset.

With the development of the rescheduling algorithms (see Chapter 5), NS is able to compare the expected operational performance of different sets of crew scheduling rules. An internal NS study, comparing the Circling-the-Church rules and the Sharing-Sweet-and-Sour rules, showed that these two models can technically provide the same operational performance. Small delays are absorbed by the buffers and disruptions can be managed effectively in both cases. However, with duties based on the Sharing-Sweet-and-Sour rules, on average more duties are affected by a blockage of a railway line and the solution space is also larger due to the broader knowledge the crew members have by operating on more lines. This means that more crew members can be used while solving a disruption. The set of Circling-the-Church rules, being subject of discussion based on debatable arguments, were evaluated, and proved to be not having the expected effect. This ended the long-lasting discussion about the effectiveness of this set of rules.

Although the implementation of support for the crew scheduling problem was the first success, the research department also worked on the development of DSSs for other topics. For instance on the rolling stock assignment problem. For this problem, there was no existing solution on the market. Initially, NS tried to team-up with an external software vendor specialized in constraint programming. An advanced demonstration system was developed, but in the end, the conclusion was that the tool was not suitable to solve the complex rolling stock assignment problems of NS. Therefore, NS decided to start its own research program on this topic.

Several students studied sub-problems where gradually the complexity of the sub-

problems increased over time. These students worked in close cooperation with the internal OR-specialists. NS also provided information on this topic to scientists, who found this problem very interesting and performed their own research. This contributed to finding the right model and technique for solving the rolling stock assignment problem. For publications on this topic we refer to Abbink et al. (2004), Fioole et al. (2006), Maróti (2006), and Kroon et al. (2009).

6.8   Overall improvement

As we described in the previous section, the support for crew scheduling improved gradually over the years. There was a need for continuously improving the algorithm, giving better results, decreased computation times and making the systems much easier to use by integrating all levels of support with a single system.

The introduced DSS has been used for performing what-if-scenario analysis during labor-agreement negotiations. All recent proposals, having an effect on the duty schedules, have been evaluated with the system. The results of this evaluation are accepted to be valid, by both management and the labor unions representing the crew members. This lead to a very stable organizational relationship and a more open negotiation process. Also a clear set of criteria has been developed to ensure an attractive set of duties. As the generated plans satisfy these criteria, there is less discussion on the contents of the schedules. This also contributes to this stable relationship.

During the years additional rules like the shorter duration of duties that start early, and a minimum amount of unique track kilometers per driver, were introduced. Without the changed planning process and support, it would not be possible to create duties obeying these rules.

With these rules and the duties created, the relation between management and train crew stabilized and the operational performance increased, reaching an all-time high punctuality in 2007. Only part of this performance increase can be allocated to the increased motivation of the crew members, but still this is an important factor. Also the sickness rate was very high during the conflict with management and decreased to normal proportions after the conflict ended in 2002.

The improved working methods and support also implied that fewer planners were needed. Several of the experienced planners were close to retirement and management decided not to replace them by others. This ensured that there was no agitation on the reduction in the number of planners. In 2006, 10 full time employees worked at

this department, which is a reduction of 55% compared to the number of employees in 1998, before the introduction of the system. This reduction can only be explained by the introduction of the system.

Currently, NS can theoretically create a schedule in a couple of days. For the crew schedules of the timetable 2014, there is an available time window of 4 weeks to create the schedules. In this period, there is still some slack for potential delays in the delivery of the timetable and the rolling stock schedules.

During the research and development process, several benchmark cases were made where results were compared to manually created plans and also compared to previous versions of algorithmic support. These cases showed that, with each improvement, NS was able to create more efficient plans with the systems. In 2008, a study was conducted with the then available algorithm and the planning data and planning rules of 1999, the last crew schedule that was completely created by hand. The study showed that NS could save about 9% on the amount of crew needed to perform the same amount of tasks.

In reality, the efficiency of the duties has not increased this much over the years. All the benefits of the system have been allocated to increasing the satisfaction of the Crew members by giving them attractive duties and to increasing the buffers in the schedules to reduce the impact of delays in operations. The system has enabled the management of NS to make this choice.

The developments in crew scheduling described in this thesis were part of a larger effort to introduce DSS in the planning process of NS. In Appendix C, we describe the impact of applying Operations Research techniques to support the construction of a completely new timetable, introduced by NS in December 2006. We show that close cooperation with the scientific community led to very successful applications of Operations Research techniques.

This conclusion is based on several quantitative indicators, checked by the internal accounting department of NS. These indicators, discussed in more detail in Appendix C, show that NS had an all-time high number of passengers in 2007 and train punctuality reached a record high. Since then, punctuality has been stable at or above the level reached in 2007. Also, the total annual additional profit of the development of the new timetable is approximately € 70 million. This result was largely based on the introduction of advanced decision support systems.

After this success NS improved the rolling stock schedules, using statistical models for predicting the number of passengers and using an optimization module. This improvement resulted in a further estimated cost reduction of € 30 million, so in total

these developments have contributed to a yearly gain in profit of more than € 100 million. This partly answers research question 5: "*what is the impact of decision support on the quality of the plan?*"

For extreme weather conditions, the concept of an adapted timetable has been developed, where the number of train services is reduced to be able to handle expected problems with rail infrastructure. When such an adapted timetable is in place, on the day before operations, the crew schedules are rescheduled with the developed algorithm. About two-thirds of the crew duties are affected and need to be rescheduled. NS is able to perform this task within a couple of hours (see Snijders et al. (2013)). In the winter of 2012-2013, this was operationally applied on twelve days. NS stayed in control on all of these days, even in circumstances for which expert judgments indicated afterward that NS would not have been able to stay in control without operating an adapted timetable. At the beginning of this change, adapting the timetable on such short notice was simply not possible.

The availability of decision support enabled NS to create several alternative plans providing the management with the opportunity to make a thorough decision. The tools enabled NS to quickly create new schedules, needed because of unforeseen events, shortly before the date of operation. This shows that we can answer research question 6: "*what is the effect of decision support on the Crew Management process? This includes the new working methods, improvements in lead-time, reaction time and flexibility, and support for management decisions.*"

6.9   Lessons learned and recommendations

In this chapter, we first described the setting at the start of the change process. After that, we discussed the change process and gave examples of the development of advanced algorithms to support the initially manual planning process. Key elements in success and failure were addressed.

In this section, we answer the two remaining (fifth and sixth) framework questions:

- What have you learned about change and how can you remember it for the future? How can you pass on what you have learned?

- Once the change is completed, what comes next? The completion of one change simply serves as the starting point for the next.

6.9.1   Literature on applying OR in practice

This section starts with a summary of lessons learned on applying Operations Research in Practice in other domains than NS. These lessons were found in literature, where three papers are related to practices that were rewarded by the INFORMS Franz Edelman Award, thus demonstrating that these practices are internationally recognized as the best in business. We will relate these lessons learned to the best practices we identified at NS.

1. Dutta (2000) presents some observations that influence the success of OR/MS in Indian and US Steel plants

2. Nigam (2008) presents the approach of Merrill Lynch on delivering business results by excellence in Management Science. Gains of several hundred million of dollars are reported in 20 years of applying OR/MS;

3. Benoist et al. (2012) summarize their lessons learned from 15 years of practicing OR at Bouygues e-lab, which led to a reported increase in revenue of € 20 million per year;

4. Fleuren et al. (2013) present the success story of supply chain–wide optimization and the Global Optimization (GO) Academy at TNT Express, leading to € 207 million of cost savings over the period 2008–2011.

6.9.2   Required Skills

OR skills   Benoist et al. (2012) tell that their team has a broad knowledge on different optimization techniques so they can select the right technique for the problem to be solved. Dutta (2000) states that it is less relevant to have the right up-to-date knowledge on OR/MS techniques, and that basic knowledge is enough to start with applying OR/MS successfully.

The R&D department at NS has analysts who are OR-professionals, with knowledge on a variety of techniques. To keep that up-to-date, they work on different topics, that require different techniques to solve them.

IT-Skills   Next to OR/MS skills, IT skills are essential, as reported by Nigam (2008) and Benoist et al. (2012). Fleuren et al. (2013) do not explicitly mention IT skills, but have developed a GO data management tool to get the needed data from several internal IT-systems, which they consider being crucial for the development of user-friendly and fast models.

We stress that IT-skills are essential. First of all, the data preparation is crucial for obtaining the right results. A human planner is able to handle inconsistent or incorrect data much better than an algorithm. IT Functionality needs to be developed so that data integrity is checked before the system can be used. It should support the user in correcting data in an easy way. Much effort is invested in understanding the problem domain and distilling useful parts of real-world data (including incompleteness, inaccuracy and sometimes unavailability). When creating optimization models for a real-time system it is even more difficult to collect the required data and restoring solutions. Interfering with operational systems needs to be done carefully, without risking failure of those systems.

Second, the system should present the results with an intuitive graphical representation. Analysts should be able to create the right representation of the results, not only to communicate them with the end users, but also for their own interpretation of the results and the development of the models. With graphical representations, the results of the system can be analyzed, and elements of the solution that can be improved are identified.

Finally, the system should support easy adjustment of objectives and constraints. The user must be supported in setting the right parameters in order to achieve the required schedules. This requires both knowledge of the model and of building IT systems. Making the system easier to use facilitates better use of the optimization models.

Social Skills   Nigam (2008) indicates that social skills are essential. The ability to deal with people: Ability to communicate well, negotiate, listen, resolve conflicts, smile, make eye contact, and carry on a conversation.

The analysts at the R&D Department of NS, are expected to be able to build a good understanding and relation with the end users. First of all, this is needed for understanding and modeling the problems. After this, in order to get the system to be used in operations, they need to influence and convince planners to use the system. Next to specific training in this, a success factor is the personal attitude and drive of an analyst.

Even when the functionality is easy to use, the user should be supported in using the system in the best way possible. The user is trained; and the analysts coach him while using the system and even create complete schedules. In order to supply this support, for example, the analysts are trained in social skills like teaching.

Find the right people   Nigam (2008) strongly focuses on the people that are crucial for success. Although technical and social skills are required, a positive attitude and persistence, together with a self-drive to make a difference, personal accountability and respect for all make the difference between success and failure.

We agree that the people in the team make this difference between success and failure. Interns and PhD-students are a good source for recruiting the right people. When they demonstrate the right set of technical and social skills and show the right attitude to make the difference, often a position within the R&D department is created for them. The R&D Department at NS is very careful in hiring people and seldom hires people without having worked with them in person.

6.9.3   Required Knowledge

Knowledge of domain   Dutta (2000) emphasizes the need to have practical knowledge of the domain, acquired by performing operational tasks. For example, manning an operational control center for the allocation of power in the steel plant.

In the R&D department, we agree that, to develop a real-world DSS, knowing the details of the problem is essential. When modeling the problem, first the formal constraints and objectives are formulated. After generating solutions, the planner can give feedback on these and show elements that are "not so good," or even infeasible. Most of the effort is spent on getting to understand the hidden rules that are behind these remarks. There are many details and relations that the analyst should be aware of in building and using the system. It takes years, and a short distance between operations and R&D, before this knowledge elicitation is complete and incorporated in the models.

Acquiring external knowledge   Dutta (2000), Nigam (2008) and Benoist et al. (2012) report that time needs to be reserved for performing activities that are not directly related to the projects performed. This to keep the right knowledge, maintain a network with other professionals and work on projects of personal interest that stimulate the analysts on personal development.

At the R&D Department, we stimulate visiting congresses and seminars. Also having part-time affiliations with a scientific institute, next to their affiliation at NS, is supported. Close cooperation with the R&D department of NS and scientific institutes enables transfer of domain knowledge as well as scientific knowledge. For example in the project described in Chapter 5, the close cooperation between NS and D-CIS Lab enabled the transfer of domain knowledge as well as actor-agent knowledge

to the mutual benefit of both parties and the overall success of the research system.

Joint development is not limited to scientific communities. The work described in Chapter 3 is the result of joint research between a commercial software vendor and a large railway operator. It is interesting to see that the combination of the common Operations Research knowledge, the specific IT knowledge of SISCOG and the specific domain knowledge of NS lead to benefits for both parties. For SISCOG, the research resulted in a new optimizer that was integrated in their software suite. Their competitive strength was enlarged, and SISCOG is now able to support the most complex crew scheduling problems. For NS, this research resulted in a ready to use system, improving both the planning process and the resulting schedules.

6.9.4   The R&D Process

Selecting projects   What these lessons from different companies have in common is that they all focus on selecting the projects with a large potential for success. How they organize this is different. Benoist et al. (2012) use internal payment for the services to focus on real business needs, while Dutta (2000) selects the right project by its own internal knowledge of the domain. They focus on problem solving, where they want to find solutions for urgent operational problems. It is crucial to positioning the OR/MS team close to the operations and not as a separate business unit.

Nigam (2008), suggests solving complex or ill-structured problems to demonstrate the added value of OR/MS. Make sure to keep a healthy workload for the employees involved by prioritizing projects, by focusing on business impact and implementation from the beginning.

For introducing OR in business, Fleuren et al. (2013) suggest starting simple and following business maturity in applying more advanced methods. These basic OR solutions can already result in a large improvement. Many projects were started based on Master projects in their GO academy, and are implemented in the daily operations of TNT and provide direct business value.

At NS, projects are started that are easy to scope, and for which solutions could be compared with existing ones. The systems are introduced step-by-step and gradually improved. This is due to the fact that the problems are too complex to be solved within the first implementation. Next to that, although there are many problems that can be optimized, the focus is on a few of them. With the limited resources available, we carefully choose the right problem to be solved.

Model development   Benoist et al. (2012) indicate two factors in the development of a model: (1) avoid using hard constraints in a model, only use them when they relate to physical limitations. This is to make it easier to detect causes of problems in finding a solution, almost always due to inconsistency in input data, and (2) present solutions early in the project. This helps in getting correct input data, and in finding additional specifications for the model.

At the R&D department, we do not apply such modeling guidelines yet, but solutions are presented early in the project. Interaction from the start of a project is essential in modeling the right problem, and in correctly modeling the problem. Showing intermediate results also helps understanding the model and the way results are created, by the planner.

Change management   Fleuren et al. (2013) introduced a benefits tracking system to objectively monitor expected, agreed and implemented savings. The main objective of the GO academy is to teach optimization principles to TNT Express employees and to acquaint them with the available optimization tools, without turning them into mathematicians. At the end, the GO academy exceeded expectations, OR has become part of the core values of TNT express and a solid network of people within TNT are ambassadors for introducing optimization models.

Nigam (2008) involves analysts in all stages of a project, this to ensure that the final solution delivered is in line with the intake of the original problem.

We think that implementing an advanced DSS requires changes by nature. Processes and working methods need to be aligned with the new possibilities that these systems bring. Most of the users of the systems do not immediately see the benefits of introducing them. They have been doing their work in a certain way for years, and for them it is not always obvious why they should change that.

Implementing a DSS is potentially threatening the daily work. When a system can solve the problems, there are potentially fewer employees needed that perform the tasks by hand. The system should be introduced in such a way that it supports the user and does not replace them completely. It could be the case that fewer employees are needed in the end.

Next to the gradual introduction of the system, there was an increase of the amount of work that needed to be handled. Taking advantage of the new systems, NS is able to change its plans more often and provide a more flexible service to the customers. Also, the systems enable to perform scenario studies and to evaluate alternative plans. This additional work compensates for the potential loss of the

regular manual work. Finally, NS is a large and social company where there is a low risk of losing your job. These aspects assured that planners were comfortable about the potential threat of the systems.

To overcome resistance to change, it helps that there is a real need for implementing the system. For example, the new Sharing-Sweet-and-Sour rules for creating the crew duties made it practically impossible to create the duties by hand anymore. This, combined with the fact that experienced planners went on retirement, made the users accept that TURNI was used.

For the rolling stock model this was less clear. Initial versions of the model already showed good results. The first time it was operationally used in 2005, there was heavy pressure on reducing costs and with the system for scheduling rolling stock costs could be reduced. After this, pressure on really reducing operational costs diminished, and the system was not used all the time anymore. Only for the most complex part, the so-called "north–east" lines, where trains split and combine in Zwolle and Utrecht, the model was used regularly. For the other, less difficult parts, the planners preferred to create the schedules manually. When limited urgency to use the systems is observed, the focus should be on making the use of the systems more attractive by extensive user support and on improving the ease of use of the systems. NS also tried to create urgency by limiting the available number of planners, but this was no success. This resulted in delivering incomplete plans and not in the request for better support.

6.9.5   Consolidating Lessons Learned

As described in the previous section, the R&D department of NS has gradually increased and currently has a solid base of persons that are well aware of the organizational change aspects of introducing decision support systems. The department has a mixture of experienced and less experienced employees, and they work together closely on projects and exchange these lessons learned. Some of the people focus on the development of the systems, others on the usage and organizational change. This is a solid basis to consolidate the knowledge about developing DSS and the related change management processes.

6.9.6 Future Work

Currently NS focuses on introducing DSS in the real-time operations. Urgency for improving real-time disruption management has become apparent in recent winters where rail operations poorly performed due to the bad weather conditions. Currently NS is not capable of delivering the service needed in these conditions. Introducing DSS for these conditions will also help in regular disruptions, which occur every day.

Current research topics are:

- Real-time passenger flows, especially in case of disruptions.

- Handling uncertainty of disruptions

- Delay management

- Real-time disruption management

Real-time disruption management gives new challenges, related to the activities in the planning domain.

First, it is harder to get the users acquainted with DSS. They are distributed over the country, work in shifts and in 7x24 rosters. A department called "Model Besturing Centrum (MBC)" was introduced with the aim to experiment with the usage of the models and new working methods in daily operations. They have a high educational level, compared to regular dispatchers, so they can better understand the behavior of the systems. They work in shifts and visit the other operational centers at different locations, and form a linking-pin from the R&D department to the daily operations at NS.

Second, the current IT systems have been created in the nineties and are, therefore, not easily extended with algorithmic support. A further complication is that they are to be replaced by a new generation of systems. This is a time consuming process which will take several years to be completed. This makes it hard to really implement advanced support during these years. The upside from this is the chance to prepare these new systems with functionality to connect to generation modules. This will speed up the implementation after the upgrade of the system support.

Third, it is hard to evaluate generated solutions because in practice the situation is continuously changing and it is difficult to determine the effect of applying the solutions in this changing reality. Simulation studies show that the algorithms can generate good solutions, but only real applications can show the real value of the tools.

The final challenge is developing new algorithms that focus on finding solutions fast, not requiring optimal solutions. In real-time operations it is better to have a quick solution, than having the best solution, which has become invalid because the real-time situation changed. NS can build on the available knowledge on planning, but needs to combine this with new approaches to speed up the solution processes.

The first step in introducing these systems in practice, is the implementation of the real-time solver for crew rescheduling in case of disruptions. This solver was developed (Potthoff et al. (2010)) with similar techniques as the crew scheduling algorithm. The time needed to find a feasible solution was reduced by introducing a duty selection mechanism, which basically reduces the problem size. This solver was compared with the solution approach presented in Chapter 5. It was chosen because it gave better solutions for large disruptions, where the solution described in this thesis gave better results for smaller disruptions. Next to that, it was easier to implement in the CREWS system because the supplier has more knowledge on the techniques used.

The people at the MBC have been trained in using this solver. The solver was built into the CREWS system which was connected to the operational dispatching system. It took more than one year to develop this connection. The conditions for the situations in which the tool can be used are well defined, so if things go wrong, the effect on the service to the passengers is limited. This system has been applied in practice by the MBC in several situations. Some problems occurred, but the organization learned from it and gradually gained confidence in using the system. In time, the system will be used more often and will become a real improvement for daily operations.

When this is successful, the aim is to support the rolling stock dispatching process as well as the dispatching of rolling stock at the shunt areas. Finally, based on models predicting real-time passenger flows, advanced support for full delay management and disruption management can be introduced.

6.10 Conclusion

In this chapter we described how we apply Operations Research at NS, we elaborated on change management and on lessons learned. We discussed that our success resulted from close cooperation with scientists, software vendors and with experts in the NS domain. Organizing the R&D department close to the end-users was critical in achieving the result.

In Section 6.6, we discussed the results achieved. Overall an estimated gain in revenue of more than € 100 million was achieved, NS has shortened the lead-time of the Crew Management from years to days, and the relations between management and crew members stabilized with the introduction of a new set of rules and the decision support enabling the computation of related schedules. This answers the two research questions: (5) *"what is the impact of decision support on the quality of the plan?"*, and (6) *"what is the effect of decision support on the Crew Management process?"*

Chapter 7

Summary and Remarks

This thesis describes how the (railway) Crew Management process can be improved with the introduction of advanced decision support systems. The Crew Management process is the process of solving the Crew Management problem. We describe several facets of the (railway) Crew Management problem, and how we introduced decision support systems, based on advanced mathematical models and algorithms, to support solving the problem.

In Chapter 1, we introduced the Crew Management problem, containing the crew scheduling problem, the crew rostering problem and the real-time crew dispatching problem. We introduced the main research question of this thesis:

> "How to improve the Crew Management Process
> in Passenger Rail Transport?"

The research in this thesis was performed over a period of more than fifteen years. In the beginning of this period, no advanced decision support existed in the railway industry. In the airline industry, similar models and algorithms were available for more than ten years, but these methods had not yet been applied in the railway industry fifteen years ago. We showed that the application of these models and algorithms in the railway domain is not trivial, but in the end was successfully performed.

To answer the main research question we formulated and answered the following sub-questions:

1: *"What are the important aspects of Crew Management?"*

In Chapter 2, we have described the Crew Management problem and its important aspects, as well as the related planning and dispatching problems in

practice. These problems are illustrated by NS examples. Based on our knowl-
edge of the problems and processes of other railway operators (most of them
operating in Europe) we consider the situation at NS as representative for many
other rail operators.

2: *"How can we quantify the quality of a plan?"*

In addition to the general description of the logistical problems, we have in-
vestigated the important criteria for assessing the quality of a crew plan: (1)
efficiency, (2) robustness, and (3) quality of work. Efficiency means that the
total crew costs are as low as possible. Robustness of the crew duties, i.e.,
preventing propagation of delays via the crew schedule, is not quantified with
a single unit of measure. In general, several indicators are used to measure
robustness. For example, we measure the available transfer time of the crews
when transferring from one train to another. The quality of work is the per-
ceived quality of duties by the crew members. This is addressed via labor rules
and company agreements, for example, on the amount of variation in the duties.
The indicators of the quality of a crew plan are discussed in Chapter 2.

3: *"What are commonly used methods for supporting the Crew Management
process?"*

We reviewed available OR methods for solving the sub-problems of the Crew
Management problem. These are described in the related Chapters 3, 4, and
5.

4: *"How to apply these methods to the Crew Management process?"*

Based on the understanding of the Crew Management problem and the available
solution techniques, we considered which techniques are appropriate for solving
the sub-problems and we have shown how these techniques can be used in
practice. Chapters 3, 4, and 5 describe how we applied these methods for the
related problems. Recent developments of scientific techniques and technology
provide the railway community with a wide range of tools that can be used
to develop the required support. This thesis has provided several examples
of successful developments and practical results, in the context of Netherlands
Railways.

5: *"What is the impact of decision support on the quality of the plan?"*

In Chapter 6, we have shown the overall effects of applying decision support on
the quality of the plan, in terms of efficiency, robustness and quality of work.

We have shown that we improved the quality of the plan on all three aspects, using the implemented DSS.

6: *"What is the effect of decision support on the Crew Management process?"*

In Chapter 6, we have shown that introducing decision support for the Crew Management Problem in the context of Passenger Rail Transport can give substantial improvements in the overall performance of a railway company. Within NS, the support for the Crew Management process has led to a stable relationship between management and train crew. In addition, the lead–time of the planning process has been shortened from months to hours and NS is now able to perform scenario analyses, e.g., for studying effects of adjusting the labor rules. NS can also adjust its service when severe weather conditions are expected, by creating a specific winter timetable shortly before the day of operation. We also introduced a decision support system for real-time rescheduling of crew duties in operations.

7: *"How to implement decision support in an organizational context?"*

Chapter 6 also describes the way we implemented decision support in the NS organization, how we managed the change process, and the lessons learned. In this chapter, we provide a managerial perspective on the change process, related to the introduction of decision support systems, and give an overview of the lessons learned in the development of these decision support systems. We used a generic change framework, to describe the process we went through, which enables other scientists to compare our process with others that use a similar framework for describing a change process. We compared our lessons learned with lessons of other Operations Research practitioners, which contributes to a common set of lessons learned and best practices.

With the answers to the sub-questions, we have answered the main question of this research. This answer is not the end of the improvement of the Crew Management Process at NS. As one of the framework questions presented in Chapter 6 asks: *"Once the change is completed, what comes next?"*

For the near future, we foresee that support can be developed for handling small disruptions in operations, for a new way of rostering (based on the individual preferences of the crew members), and for scheduling other types of crew, for instance, developing models and algorithms for the scheduling of "Service & Security" teams.[1] We are currently researching algorithms for this.

---

[1] These mobile teams are NS employees, specialized in reducing fare evasion and social aggression.

For the distant future, we foresee an integration of the crew scheduling problem and the crew rostering problem, and possibly an integration with scheduling other resources like rolling stock. Together with developing support for other problems in the railway domain, a lot of challenging research is still to be performed.

Appendix A

Related Optimization Problems and Solution Techniques

In this appendix, we discuss three well-known combinatorial optimization problems, which we relate to the CSP addressed in this thesis. Furthermore, we provide some mathematical formulations of these problems, and we introduce some solution techniques, frequently applied to solve the discussed optimization problems. We assume that the reader is familiar with the basic concepts of and mathematical programming.

A.1 Related Optimization Problems

In this section we discuss three combinatorial optimization problems: (1) The Minimum Cost Flow problem, (2) The Set Partitioning Problem, and (3) Set Covering Problem. These problem formulations are used to describe parts of the CSP in this thesis.

A.1.1 The Minimum Cost Flow Problem

The *minimum cost flow problem* can be defined as follows (see Ahuja et al. (1993)): *Determine a least cost shipment of a commodity through a network that will satisfy the flow demands at certain nodes from available supplies at other nodes.*

More formally, define $G = (N, A)$ as a directed network with N as the set of nodes and A as the set of directed arcs. Each arc $(i, j) \in A$ has cost c_{ij} per unit flow on that arc. The flow cost varies linearly with the amount of flow. Define u_{ij} (l_{ij}) as the maximum (minimum) amount of flow on arc $(i, j) \in A$. Finally, associate integer b_i for each $i \in N$, which can be seen as the demand (supply) of such a node if b_i is negative (positive). If $b_i = 0$, node i is a transshipment node.

The minimum cost flow problem, in which x_{ij} represents the flow on an arc $(i, j) \in A$, can be formulated as follows:

$$\min \sum_{(i,j) \in A} c_{ij} x_{ij} \tag{A.1}$$

$$s.t. \sum_{j:(i,j) \in A} x_{ij} - \sum_{j:(j,i) \in A} x_{ji} = b_i \qquad \forall i \in N, \tag{A.2}$$

$$l_{ij} \leq x_{ij} \leq u_{ij} \qquad \forall (i, j) \in A. \tag{A.3}$$

In the above formulation, the total costs are minimized such that the inflow minus the outflow in each node is equal to its demand/supply, and the flow on each arc is between its lower and upper capacity.

In general, minimum cost flow problems are not solved as linear programs but by specialized algorithms. A lot of well-known problems are special cases of the minimum cost flow problem. We will discuss here one, the *shortest path problem*, which is relevant to this thesis. In the shortest path problem, the shortest path, i.e., the one with the lowest costs, between two nodes s and t has to be found. It can be easily seen that this is a special case of the minimum cost flow problem. Take $b_s = 1$, $b_t = -1$, and $b_i = 0$ for all $i \in N \notin s, t$ and set $l_{ij} = 0$ and $u_{ij} = 1$ for each arc $(i, j) \in A$, then one unit of flow will be sent through the network from s to t along the shortest path.

A.1.2　The Set Partitioning Problem

The *set partitioning problem* can be formally defined as follows (see Lenstra and Rinnooy Kan (1979)): *Given a finite set S and a finite family F of subsets of S, does F include a subfamily F' of pairwise disjoint sets such that $\cup_{S' \in F'} S' = S$?*

The set partitioning problem can be formulated as follows:

$$\min \sum_{f \in F} c_f x_f \tag{A.4}$$

$$s.t. \sum_{f \in F} a_{if} x_f = 1 \qquad \forall i \in S, \tag{A.5}$$

$$x_f \in \{0,1\} \qquad \forall f \in F. \tag{A.6}$$

In this formulation a_{if} is 1, if subset f contains element i for each $f \in F$ and $i \in S$, and 0 otherwise. Furthermore, by using decision variables x_f equal to 1 if subset f is in the chosen solution and 0 otherwise.

A.1.3   Set Covering Problem

The *set covering problem* is a small variation to the set partitioning problem. Each element must be in at least one subset, instead of in exactly one. It is obvious that, each feasible solution of the set partitioning problem is also a feasible solution of the set covering problem. Furthermore, the optimal solution (and each lower bound) of the set covering problem is a lower bound of the set partitioning problem. In Section 3.2, we formulate the CSP as a Set Covering Problem.

A.2   Related Optimization Techniques

In this section we discuss two important combinatorial optimization techniques, (1) Lagrangian relaxation, and (2) column generation, that are used in the algorithms described in this thesis.

A.2.1   Lagrangian Relaxation

Lagrangian relaxation can be used to obtain bounds on the optimal solution value of a combinatorial optimization problem. It is an alternative for using the, well–known, Linear Programming relaxation. The idea of Lagrangian relaxation is to relax some of the difficult constraints, and penalize their violations by a certain weight in the objective function. Without loss of generality, we assume that the optimization problem at hand is a minimization problem. In this case, the Lagrangian relaxation can be used to obtain lower bounds. We explain the technique by applying it to the general integer-programming problem given below. We split the set of constraints into

two sets, the hard constraints and the easy constraints (N^1 and N^2, respectively).

$$(P) = \min \sum_{j \in J} c_j x_j \tag{A.7}$$

$$s.t. \sum_{j \in J} a_{ij} x_j = b_i \qquad \forall i \in N^1, \tag{A.8}$$

$$\sum_{j \in J} d_{kj} x_j = e_k \qquad \forall k \in N^2, \tag{A.9}$$

$$x_j \geq 0 \text{ and integer} \qquad \forall j \in J. \tag{A.10}$$

We introduce a Lagrangian multiplier λ_i, for each hard constraint i. We delete those constraints from the model and include them in the objective function. This way, the Lagrangian sub problem can be formulated as follows:

$$\Phi(\lambda) = \min \sum_{j \in J} c_j x_j + \sum_{i \in N^1} \lambda_i (b_i - \sum_{j \in J} a_{ij} x_j) \tag{A.11}$$

$$s.t. \sum_{j \in J} d_{kj} x_j = e_k \qquad \forall k \in N^2, \tag{A.12}$$

$$x_j \geq 0 \text{ and integer} \qquad \forall j \in J. \tag{A.13}$$

$\Phi(\lambda)$ is a lower bound on the optimal solution value of the original problem for each vector $\lambda = (\lambda_i)_{i \in N^1}$, since each feasible solution in the original problem is also feasible in the Lagrangian sub problem, and $\Phi(\lambda)$ is equal to the objective value of such a feasible solution. Similarly, we can deal with inequality constraints in set N^1. The corresponding Lagrangian multipliers must then be restricted in sign, in case of "$\geq$" constraints, $\lambda_i \geq 0, \forall i \in N^1$, and in case of "$\leq$"' constraints, $\lambda_i \leq 0, \forall i \in N^1$.

Since each vector λ yields a lower bound, we obtain the best lower bound by solving the Lagrangian dual problem $max_\lambda \Phi(\lambda)$. A commonly used iterative procedure, called *subgradient optimization* introduced by Held and Karp (1971), is well suited to solve this problem. For extended surveys on this topics we refer to Geoffrion (1974), Fisher (1981), and Beasley (1993).

We apply the Lagrangian Relaxation technique to the set partitioning problem introduced in Section A.1.3. The Lagrangian sub problem, of the set partitioning problem, where we relax the constraints A.5 and introduce a multiplier λ_i for each constraint i, can be written as:

$$\Phi(\lambda) = \min \sum_{i \in S} \lambda_i + \sum_{f \in F} (c_f - \sum_{i \in S} \lambda_i a_{if}) x_f \tag{A.14}$$

$$s.t. \quad x_f \in \{0,1\} \qquad \forall f \in F. \tag{A.15}$$

From this, the optimal solution follows immediately:

$$x_f = \left\{ \begin{array}{ll} 1, & \text{if } c_f - \sum_{i \in S} \lambda_i a_{if} < 0, \\ 0, & \text{otherwise.} \end{array} \right\} \quad \forall f \in F. \tag{A.16}$$

It can been shown that, since the optimal solution of the Lagrangian sub problem, without integrality constraints, is already integral, it follows that the best Lagrangian lower bound is equal to the value of the LP-relaxation.

A.2.2 Column Generation

Column generation is a technique that is often used for problems with a huge number of variables. The general idea, introduced by Dantzig and Wolfe (1960), is to solve a sequence of reduced problems, where each reduced problem contains only a small portion of the set of variables (in this case "columns"). After a reduced problem is solved, a new set of columns is obtained by using dual information of the solution. The column generation algorithm converges once it has established that the optimal solution, based on the current set of columns, cannot be improved upon by adding more columns. Then the optimal solution of the reduced problem is the optimal solution of the overall problem. We will refer to the problem of generating a new set of columns as the pricing problem.

We will explain the concept in more detail using the following, general linear programming problem (P') with a huge number of variables $|J|$:

$$(P') = \min \sum_{j \in J} c_j x_j \tag{A.17}$$

$$s.t. \sum_{j \in J} a_{ij} x_j = b_i \qquad \forall i \in N, \tag{A.18}$$

$$x_j \geq 0 \qquad \forall j \in J. \tag{A.19}$$

If we apply column generation to solve (P'), we start with an initial set of columns $J^0 \subseteq J$, such that these columns contain a feasible solution of the reduced problem. We solve this problem, and compute the optimal values of the dual variables. These dual variables are needed in the pricing problem to select new columns with negative reduced costs, which will be added to the set J^0. The reduced cost of a column is defined as $c_j - \sum_{i \in N} a_{ij} u_i$, where u_i is the optimal value of the dual variable corresponding to each constraint in A.18.

If there are no columns left with negative reduced costs, it is easily seen that the

optimal solution of the reduced problem is also the optimal solution of (P'). This follows from the rules of the well-known simplex method, to solve linear programs (e.g., see Chvátal (1983)). Notice that if $min_{j \in J}\{c_j - \sum_{i \in N} a_{ij} u_i\} > 0$, all columns have positive reduced costs. Due to this observation, it is not necessary that all columns are explicitly known.

The pricing problem can be seen as an optimization problem by itself, and columns can be generated implicitly by solving this optimization problem. In the CSP, this problem is a shortest path type of problems (simple shortest path or shortest path problems with resource constraints).

If (some of) the decision variables need to be integral, branch-and-bound algorithms are most often used to solve the resulting (mixed) integer programming problem. Column generation can be embedded in the branch-and-bound tree by solving the relaxations in each node with a column generation algorithm. This solution approach is called branch–and–price. For a general discussion and details about the theory on column generation in an integer-programming context, we refer the reader to a few surveys on this topic: Barnhart et al. (1998), Desrosiers and Lübbecke (2005), Desaulniers et al. (2005).

In many papers where dynamic column generation is applied, it is usual that such algorithms evaluate the effectiveness of a feasible duty based on dual information related to the linear programming relaxation of the underlying model. However, it can also be combined with Lagrangian relaxation introduced in the previous paragraph. For an overview of applying the combination of both techniques we refer to Huisman et al. (2005).

Appendix B

Reinventing Crew Scheduling

In this appendix, we describe how we used Operations Research techniques success-
fully to support its crew scheduling process in practice. In 2001, a conflict arose
between the drivers, the conductors, the works council, the unions, and NS' manage-
ment concerning the structure of the drivers' and conductors' duties. Here the works
council is an organization representing the employees that exists in each sufficiently
large Dutch company. The seemingly unsolvable conflict within NS paralyzed the
railway system and led to nationwide strikes.

However, we used the crew scheduling system TURNI to support the development
of an alternative set of scheduling rules called Sharing-Sweet-and-Sour. This set of
rules was acceptable for all parties, since it improved the quality of the drivers' and
conductors' working conditions, and it supported an improvement of the railway
operator's punctuality and efficiency.

The alternative set of rules is so complex that it is nearly impossible to take them
into account efficiently when planning manually. Automated support in the crew
scheduling process is absolutely necessary. Therefore, NS has been using the auto-
mated crew scheduling support since 2000, see Section 3.4. Before NS started to use
automated support, it scheduled its crew mainly manually, using several supporting
information systems. These systems did not automatically schedule the drivers' and
conductors' duties; the crew scheduling process strongly relied on the experience and
craftsmanship of the planners.

B.1   History

On June 10, 2001, NS introduced the set of scheduling rules, popularly called Circling-the-Church (in Dutch: Rondjes-om-de-Kerk), to schedule the duties of its drivers and conductors. The management wanted to improve their trains' punctuality: according to this set of rules, train drivers and conductors would work on less different lines, and in principle they would transfer from one line to another only during their meal breaks. In this way, the management hoped to reduce the usual snowball effect of delays of trains. The management also expected that these rules would improve NS' passenger service, because crews would be more familiar with local conditions.

The unions and the works council had been involved in developing the Circling-the-Church rules and had approved their introduction. However, the drivers and conductors were quite unhappy when the details of the rules became apparent. They went on strike for several days in an attempt to prevent their introduction. Note that, for a country as densely populated as the Netherlands, that is highly dependent on its public transport infrastructure, strikes in the railway system have a dramatic social and economic impact. As a consequence, the Dutch media paid a lot of attention to the problems within NS.

After various mediation attempts had failed, the works council, the unions, and the management finally came to an agreement on April 23, 2001. This agreement described a path towards normalized working relations. The railway operator would introduce the Circling-the-Church rules as planned on June 10, 2001. However, at the same time, the works council would have the opportunity to develop an alternative set of scheduling rules. This opportunity was not open ended: if the management would not accept the council's alternative, then the decision would be subject to binding arbitrage. If the management would accept the council's alternative, then it would go into effect in 2003.

Why were the drivers and conductors so unhappy with the Circling-the-Church rules? Because drivers and conductors would work on less different lines, they complained that they would have almost no variety in their duties: the resulting monotony would decrease the quality of their working conditions.

Moreover, the work on several lines of NS (mainly in the "Randstad" in the western part of the country) is less attractive due to the relatively high level of passenger aggression on these lines. With the Circling-the-Church rules, the personnel from only a few crew depots would be confronted with this passenger aggression. The drivers and conductors from these crew depots considered it as unfair that they would get a relatively large part of this unattractive part of the workload. Besides

these direct issues, the crews also feared that the management wanted to establish the Circling-the-Church rules because it had a secret agenda including privatization of the unprofitable lines. All these reasons accumulated into a strong opposition against the Circling-the-Church rules.

B.2 The development of the alternative rules

On April 23, 2001, agreement gave the works council the opportunity to develop an alternative to the Circling-the-Church rules. Besides that, the agreement specified that the works council could hire external experts to help it to develop this alternative set of rules. The works council finally asked the combination of Basis-en-Beleid (a consulting company with experience in social and political conflicts) and ORTEC (a consulting company with expertise in logistics) for support. The consulting companies' assignment focused on avoiding arbitrage, since in the case of arbitrage all parties would be losers. So, the consultants should look for a compromise that would be acceptable to all parties.

However, there were many conflicting points of view. The differences between personnel and management were obvious, but the personnel also had many opinions, depending on function, age and crew depot. In addition, the various unions held different points of view, and the relationship between the personnel and the works council was weak. Moreover, management positions turned out to be unstable: in the first few days of 2002, the complete board of the company and two top managers left the stage. In other words, success was not guaranteed at all. But all involved parties agreed that they had to create a solid basis for the alternative set of scheduling rules.

B.3 The participative approach

ORTEC and NS' Department of Logistics could provide the needed expertise in logistics. But, on the other hand, it was clear that any plan developed by experts was doomed to fail. Therefore, Basis-en-Beleid and ORTEC chose for a participative approach: they developed the alternative set of scheduling rules in cooperation with the drivers and conductors. They therefore called the project Je-Bent-Er-Bij (You're-Part-of-It). Openness and transparency were the central concepts. Everyone could follow the whole process at the project's web site, and all employees received dedicated newsletters.

In the first months of the project, the consultants conducted discussions with

700 people from all crew depots. One could fill a book by listing all mentioned bottlenecks, but the central theme in all discussions was the demand for more variety in the duties. Thereafter, the consultants randomly selected in total 300 people to support the further development of the alternative set of scheduling rules. Four groups of 75 people met several times in two-day working conferences. As a result, the project came to life, and several alternative solutions saw the light. The working conferences informed the works council about these alternatives.

Finally, the works council chose the alternative Sharing-Sweet-and-Sour, which seemed to have the most solid approval among the drivers and conductors. After several lengthy negotiations about the details of the alternative set of rules, also NS' management accepted it. The Dutch media followed the whole process and the negotiations avidly, because the duties of the drivers and conductors had long been a hot issue.

B.4   The role of Operations Research

Where did the alternative set of scheduling rules come from? Parallel to the participative process, experts from ORTEC and NS' Department of Logistics developed and evaluated several alternatives, which was easier said than done. A timetable consists of thousands of train movements. More than 6,500 crew members have to be assigned to the corresponding tasks. Moreover, in scheduling the crew members, a railway operator must satisfy the many rules prescribed by law and by the collective labor agreements. For example, each crew member has to work on average about eight hours per day and needs a meal break at an appropriate time and location. As a consequence, the crew of a train must be exchanged regularly. The transfer times between successive tasks of the same crew cannot be too short. Moreover, each driver has a license for only a limited number of rolling stock types and a limited number of lines. The alternative set of scheduling rules had to produce efficient crew schedules, improve the punctuality, and increase the variety in the duties. Altogether, it was an enormous challenge.

To generate duties that complied with all these objectives, we used the planning support system TURNI. NS' Department of Logistics has been using TURNI to produce the basic structure of the duties for drivers and conductors since 2000. In particular, it used TURNI also to produce the duties in line with the Circling-the-Church rules. TURNI generates all the duties for drivers or conductors for a single workday in one single run.

We carried out the computations in parallel to the working conferences. The working conferences defined preferences that we translated into parameters for the model. We then informed the participants of the working conferences about the model results. The model results were also the basis for the further choices. Due to the many different potential alternative sets of scheduling rules, we needed hundreds of runs with the model. Especially during the first phase of the process, we studied many scenarios at the same time with TURNI, keeping several PCs running continuously to solve them.

Initially, we used only a relatively small number of key performance indicators to evaluate the many scenarios and to compare them with each other and with the reference scenarios (the Circling-the-Church scenario and a green-field scenario). The indicators that we used were the total number of duties (efficiency), the total number of transfers from one train to another (punctuality), and the total number of lines and rolling stock types per crew depot (variety).

After we eliminated a number of potential alternatives, we evaluated the scenarios in far more depth. In this process, we had to update and to customize TURNI several times to facilitate the evaluation of alternative sets of scheduling rules that were not included yet.

The participative approach and the development and evaluation of the different scenarios using TURNI were not separate activities. The quantitative results streamlined the discussions at the working conferences. They made the discussions more objective, because TURNI made the consequences of various alternative sets of rules transparent, guiding participants to discuss the real choices to be made.

In this interplay between the working conferences and the quantitative analysis carried out by TURNI, the many alternatives gradually fell into a set of five main alternatives. From these five alternatives, the works council chose the alternative Sharing-Sweet-and-Sour on May 22, 2002.

In the subsequent negotiations with NS' management to set the details of several parameters of the rules, we used TURNI to evaluate the impact of different settings. The objectivity of the quantitative results provided by TURNI allowed NS' management to accept the alternative set of scheduling rules in the end. The symbiosis between the participative approach and the quantitative analysis was the basis for this final success.

B.5   Ending the Conflict

At the start of this project, the working relations between the various parties at
NS were so strained that nobody believed that a solution to the conflicts existed.
However, a well-directed combination of a participative approach to build person-
nel involvement and expertise in Operations Research for quantitative support was
successful. Operations Research revealed its power by facilitating the analysis of the
complex scenarios. Moreover, in emotional discussions, quantitative results helped
people to evaluate arguments objectively. These quantitative and objective results
enabled the management of NS to accept the Sharing-Sweet-and-Sour rules in the
end.

The railway operator's experience with the alternative set of rules were quite
positive. After the drivers had expanded their knowledge of the routes, the intro-
duction of the alternative set of scheduling rules on December 15, 2002, took place
quite smoothly. Since then, the average punctuality increased gradually from about
80 percent to about 86 percent. Here a punctuality of 86 percent means that 86
percent of the trains arrive with a delay of three minutes or less. Obviously, this in-
crease in punctuality was caused only partially by the alternative set of rules: several
other developments improved the punctuality as well, for example, the introduction
of new, more reliable rolling stock, and the intensified maintenance of the railway
infrastructure.

The only negative comments about the alternative set of scheduling rules came
from the traffic-control organization. Because the complexity of the structure of the
drivers' and conductors' duties increased by the Sharing-Sweet-and-Sour rules, in
real-time reconstructing the duties in case of severe disruption of the railway services
seems more difficult than it was in the time of the Circling-the-Church rules. However,
these negative comments were outweighed by the fact that the alternative set of rules
improved the motivation of the drivers and conductors. The latter certainly had a
positive effect on the trains' punctuality.

To the best of our knowledge, this was the first time that a European railway com-
pany used Operations Research techniques successfully, to support its crew scheduling
process in practice. Until the late nineties, the size and complexity of railway crew
scheduling problems prohibited the application of such techniques.

B.6   Impact L&L

Although NS operated its trains quite efficiently already before adopting TURNI, it estimates that using TURNI saved up to 2 percent per year on its crew costs. In particular, NS used TURNI to schedule the duties for the drivers and conductors for the 2004 timetable. For this timetable, the total workload increased by about 3.2 percent in comparison with the 2003 timetable. This was the result of the fact that NS increased the number of trains to provide better service. However, supported by TURNI, the planners generated a set of duties that satisfied all the new rules, but required only 1.2 percent more duties than in 2003. In other words, the operator realized initial savings of about 2 percent. For a total of 6,500+ crews, this amounts to savings of over $7 million per year.

In practice, NS estimates the savings to be $4.8 million per year, since the management gave away part of the initial savings in order to further increase the drivers' and conductors' acceptance of the crew schedules. The final savings of 1.2 percent were still well above the target of 1.0 percent that the management initially set. These estimates of the savings are quite conservative, in particular because generating the crew schedules would have been impossible without the support of TURNI.

The target for the crew schedules for 2005 was a further cost reduction of about 2.5 percent per year. NS achieved this by keeping up the same basic efficiency as in 2004, and by relaxing some of the constraints. In particular, punctuality analysis have shown that reducing the time allowed for drivers and conductors to transfer from one train to another from 25 minutes to 20 minutes had only a minor negative effect on the punctuality, but had a positive effect on the efficiency of the crew schedules. With a system like TURNI, changing the transfer time from 25 minutes to 20 minutes is easy, whereas it would be very complex and time consuming to handle it manually.

NS' application of TURNI led to operational savings of about $4.8 million per year. A further advantage was that the railway operator became less dependent on the experience and craftsmanship of the planners. Moreover, it reduced the throughput time of the logistic planning processes, which increased the organization's flexibility and allowed it to react quickly to requests for modifications or to changes in the environment.

Appendix C

The New Dutch Timetable: The OR Revolution

In this appendix, we describe the impact of applying Operations Research techniques to support the construction of a completely new timetable, introduced by NS in December 2006. Its objective was to facilitate the growth of passenger and freight transport on a highly utilized railway network and improve the robustness of the timetable, thus resulting in fewer operational train delays. Modifications to the existing timetable, which was constructed in 1970, were not an option; additional growth would require significant investments in the rail infrastructure.

Constructing a railway timetable from scratch for about 5,500 daily trains was a complex problem. To support this process, NS generated several timetables using sophisticated Operations Research techniques. Furthermore, because rolling-stock and crew costs are principal components of the costs of a passenger railway operator, NS used innovative Operations Research tools to devise efficient schedules for these two resources.

The new resource schedules and the increased number of passengers resulted in an additional annual profit of 40 million Euro; the additional revenues generated approximately 10 million Euros of this profit. NS expects this profit to increase to 70 million Euros annually in the coming years. However, the benefits of the new timetable for the Dutch society as a whole are much greater: more trains are transporting more passengers on the same railway infrastructure, and these trains are arriving and departing on schedule more than they ever have in the past. In addition, the rail transport system will be able to handle future transportation demand growth

and thus allow cities to remain accessible to more people. Therefore, NS expects that many will switch from car transport to rail transport, thus reducing the emission of greenhouse gases.

In a small and densely populated country, such as The Netherlands, public transport plays an important mobility role and is indispensable to the economy and the public welfare. The backbone of the Dutch public transport network is the national passenger railway system. In this section, we will describe how Operations Research techniques supported the development of a new timetable that facilitates operating a higher number of trains and fewer train delays.

In 2002, all parties involved in the Dutch railway sector (ProRail, NS, and the freight operators) wrote a report, Benutten en Bouwen (Utilize and Build), which studied these issues. This report was the catalyst for launching a project to construct a new timetable with the primary goal of managing future growth on an already highly utilized railway infrastructure.

Because NS is the largest of several operators that use the same infrastructure, NS constructed one timetable for all operators; in doing so, it considered the wishes of the other operators. When there was a conflicting wish, ProRail, an independent organization, made the final decision to resolve the conflict. On December 10, 2006, NS began operating this new timetable, marking a new era for all railway passengers in The Netherlands.

In the remainder of this section, we will discuss why a new timetable was necessary, how NS developed it, and the challenges had to be conquered. We will focus on the key role that Operations Research tools played in the construction of this new timetable.

C.1   Background

The last major change in the Dutch timetable occurred in 1970. Since then, the amount of passenger transport on the Dutch railway network has nearly doubled, with 8 billion passenger kilometers in 1970 growing to 15.8 billion in 2006. During the same period, freight transport increased by 285 percent. To facilitate this growth, NS scheduled more and larger trains without changing the basic structure of the timetable. Moreover, since World War II, the infrastructure has been extended only slightly.

This had two important consequences: (1) Further growth based on the existing timetable was impossible without significant investments in the infrastructure (as

mentioned above, this was not an option), and (2) the buffers in the system became smaller and smaller, resulting in train delays; delays are the most frustrating aspect of train travel. Therefore, one of the NS primary objectives was to improve the punctuality of the railway system. The definition of punctuality in The Netherlands is the percentage of trains that arrive at one of its 35 main stations with a delay of less than three minutes.

Thus, NS had to run more trains on the network and improve the punctuality of the railway system. In principle, these two goals conflict; it is easy to add more trains to an existing timetable if NS allows lower punctuality. Conversely, it is also simple to improve the punctuality by reducing the number of trains in an existing timetable. The only way NS could fulfill both goals was to develop a new timetable from scratch.

When NS decided to develop a new timetable, it realized that it could also meet several other objectives. For example, introducing a timetable that is easy to remember on the most important lines in the western part of the country led to new commercial opportunities. In the new timetable, a long-distance train that stops at the main stations only should arrive every 15 minutes, and a regional train that stops at all stations should also arrive every 15 minutes. Moreover, the timetable should improve connections with the neighboring countries of Germany and Belgium.

The only characteristic of the 1970 timetable that remained intact was its cyclic nature. In the Dutch case, this means that each hour a train leaves at the same minute in the same direction. Passengers find this timetable property, which many other European countries also successfully apply, to be very convenient. Therefore, NS considered only cyclic variants of the new timetable. One disadvantage of a cyclic timetable is that during certain periods of the day, there may be many trains serving a relatively low demand. NS alleviates this by allowing some exceptions; for example, not all trains run in the late evenings.

A specific reason for launching the new timetable in December 2006 was that, at the start of the development process in June 2003, everyone had expected that three major infrastructure extensions would be completed by December 2006: (1) A high-speed line between Amsterdam and Belgium, (2) the Betuwe freight line between the port of Rotterdam and Germany, and (3) four parallel tracks between Amsterdam and Utrecht, which would allow fast (long-distance) trains and slow (regional and freight) trains to have their own tracks. Unfortunately, the construction work took longer than anticipated, causing many challenges for the timetabling project. In the next section, we will discuss these challenges and our solutions.

C.2   Implementation and Challenges

Although the project to construct the new timetable started in June 2003, the development of the models and algorithms in the decision-support tools had begun several years earlier.

Implementing the OR Systems

In the 1990s, NS management recognized the great potential of applying Operations Research in the planning process. Initially, it preferred to buy automated timetabling software. However, because no off-the-shelf packages were available, and external IT companies had failed to develop a prototype, NS decided to pursue an innovative approach. Its objective was to stimulate the development of new methods to solve the timetabling problem in close cooperation with the scientific community, in particular with the Centre for Mathematics and Informatics (CWI) and Erasmus University Rotterdam. After several years of research, NS (and following the NS split, NS in partnership with ProRail) implemented the methods developed as part of the DONS system. It has been in use for timetabling studies since the end of the 1990s.

After this successful project, NS management decided to establish an internal OR group within its Logistics Department. NS started several new research projects, some together with the scientific community and others internally. The ROSA system was one such project. For crew scheduling, NS performed a benchmark of commercially available systems; this resulted in the selection of TURNI. However, making it operational required a large amount of joint R&D work between NS and Double-Click sas, the supplier of TURNI. The system has been in operation since 2002.

There were three crucial factors in the success of this approach. The first was that some of us worked in the same department in which the planning problems were solved manually. Thus, it was easy for us to get data and knowledge about the real problems. The second factor was the existence of a central database containing the timetable and rolling-stock schedules. This database is an integrated system for registration and distribution of the manually created plans.

For crew planning, NS already had a manual planning system, CREWS, developed by the company SISCOG Morgado and Martins (1998). We could easily use the central database and CREWS to find the right data for developing and testing our OR methods. Moreover, once the OR methods proved to be successful, we could easily connect them to the central database and CREWS; thus, NS could distribute the generated plans to operations via these systems. The third success factor was

management support in challenging the OR experts to develop sophisticated solution methods and in investing in these projects without any guarantee of final success.

The New Timetable Construction

In the process of constructing the new timetable, NS used DONS, ROSA, and TURNI intensively and, for the first time, together. In developing this timetable, NS considered approximately 10 line systems; each varied radically from every other line system. For all these line systems, NS used DONS to generate a one-hour timetable.

Because the timetable is cyclic, a one-hour timetable can be repeated throughout the day. Therefore, NS can base many evaluations of the complete timetable on the one-hour timetable. For example, NS simulated the 10 one-hour timetables to determine their expected punctuality. Furthermore, the NS marketing department evaluated the consequences of these timetables for the passengers. Relevant criteria were the number of direct connections and the travel times. The outcome was the attractiveness of the timetables in terms of the expected passenger growth or decline. Finally, for each timetable, NS estimated the operating costs related to rolling stock and crew.

In March 2005, the NS board decided to use 2 of the 10 developed timetables and asked for a timetable that combined the best aspects of both. The result was an 11th timetable.

During 2005, it became clear that two of the three earlier mentioned major infrastructure projects (the high-speed line and the Betuwe freight line) would not be finished in December 2006. Because these are separate lines, their lack of completion would be relatively minor: the timetable would require modification in only a few sections of the country. NS performed these changes manually.

At the end of 2005, the NS board made the final decision to introduce the resulting timetable in December 2006. The process of planning the detailed rolling-stock schedules started in January 2006. The crew schedules were subsequently constructed.

Challenges During the Process

During 2006, many unexpected events happened. When the timetable plans were communicated to the public in spring 2006, many people reacted negatively to parts of the timetable. The first challenge was to address this negative reaction. It is inevitable that when one modifies the timetable of a complete country, the connections of some passengers worsen. In particular, politicians in the northern provinces were unhappy because the travel times to and from these provinces increased by a

few minutes, partly because of the opening times of a specific bridge. Therefore, the politicians began to lobby in the Dutch parliament for a faster connection with the western part of the country. However, significant changes were not possible. Therefore, the NS board decided to decrease the travel times to the north by temporarily reducing some time supplements in that area; however, this had negative punctuality consequences.

In June 2007, the opening times of the bridge were changed, thus providing a better solution. The planners directly adjusted these changes in the central database. A second, more difficult challenge arose in August 2006, when it became apparent that the construction of the four tracks between Amsterdam and Utrecht would not be completed until four months after the start of the new timetable. At that time, NS had already completed the rolling-stock schedules, and NS was about to begin the crew scheduling process. This was a serious problem because the line between Amsterdam and Utrecht belongs to the kernel of the Dutch railway network (see Figure 2.1).

Postponing the introduction of the new timetable was not an option because of the limited time remaining and the national debate already underway regarding the timetable. NS had to find another solution. Therefore, NS decided to introduce a temporary timetable between Amsterdam and Utrecht by canceling one regional train per hour and modifying some other regional trains. These modifications had a significant negative impact on the robustness of the timetable for this important part of the network, and also on the punctuality of the trains in the entire country.

To include such significant modifications at such a late time in the planning process was only possible by using TURNI to construct the crew schedules. Previously, when the crew schedules were made manually, the crew scheduling process started at least a year ahead; it now became possible to start the crew scheduling phase just three months before the introduction of the new timetable.

The third major challenge involved the expected shortage of crew capacity. In January 2006, NS foresaw that the available crew capacity would not be sufficient to operate the increased number of trains in the new timetable. The best solution was to construct more efficient crew schedules. We believed that this could be done by modeling some rules in a different way. For example, one rule prescribes that over an entire week the average length of all duties of a crew base should not exceed eight hours. In our previous computations, NS always took this rule into account by limiting the average duty length per crew base for each day of the week to eight hours. Obviously, this is a tighter constraint than the actual rule.

When we conducted a few experiments, we noticed that we could improve the crew schedules by applying the actual rule instead of the tighter rule. Therefore, we developed an extension to TURNI to address week instances. However, such instances are so huge that they cannot be solved in a single run. Therefore, we developed a procedure in which we used TURNI iteratively to solve instances of up to 15,000 tasks (see Abbink et al. (2008b)). This allowed us to reduce the number of duties by an additional 2 percent; this was sufficient to operate the new timetable.

The introduction of the new timetable on December 10, 2006 went smoothly. Initially, NS permanently monitored the operations to detect any initial problems with the timetable and the rolling-stock circulation, and, in particular, to verify that each train had a good match between seat supply and demand.

Furthermore, based on a detailed analysis of the operation, NS applied several relatively minor modifications to the timetable. The final timetable, with the four parallel tracks between Amsterdam and Utrecht, started in April 2007.

C.3 Portability

Other railway companies in Europe face challenges that are similar to those NS faced in The Netherlands. We suggest that countries with highly utilized railway infrastructures consider the construction of a new timetable from scratch. Because many of these countries use a cyclic timetable, they can also use the approach we described in this chapter: the models within DONS are generic models for solving cyclic timetabling problems.

Furthermore, because of the liberalization of the European railway market, many railway companies are interested in tools for optimizing their resource schedules. New rolling-stock scheduling systems based on the ideas described in this chapter are currently under development. Optimizing crew schedules is addressed in the next chapters.

C.4 Impact and Success

As we mentioned above, railway transport is a critical transport mode in The Netherlands. Hundreds of thousands of daily commuters use the train, and millions of people use the railway system regularly. It is obvious that when a modification in the railway timetable changes so many daily-life patterns, it will cause much media attention and many discussions in the Dutch parliament.

Communicating the need for this dramatic change was not easy. Therefore, NS asked Johan Cruijff, the most distinguished Dutch soccer player ever, to discuss the advantages of the new timetable in television commercials. In one commercial, he talked about how frequently the trains in the railway system run: "When you just missed your train, you are always in time for the next one." He also compared the railway timetable with soccer-game tactics.

After more than a year of operating the new timetable, we can measure its success. NS had an all-time high number of passengers in 2007. When NS made a detailed analysis of all individual routes, it discovered that the routes on which NS put more trains into service had a much higher increase (as high as 15 percent) in passengers than the average (which was 2.8 percent). Overall, NS expects that approximately two percentage points of the long-run passenger increase will be because of the new timetable.

Moreover, in 2007, train punctuality reached a record high: 87 percent of the trains arrived within three minutes of their scheduled arrival time; in both 2005 and 2006, this percentage was 84.8, more than two percentage points less. This is even more remarkable because the punctuality over the first four months of 2007 was almost at the same level as in 2006 because of the delayed opening of the four tracks between Amsterdam and Utrecht. If we replace the first four months of 2007 by the same months of 2008, the punctuality is 87.5 percent.

The NS marketing department conservatively estimated that the changes in the new timetable generated an additional annual profit of 10 million Euros in 2007, which it expects will increase to 20 million Euros in 2009 and later years. One factor in both estimations was an expected punctuality increase of 1.5 percentage points. Several years ago, NS made an agreement with consumer organizations that would allow a bonus fare increase of 2 percent if NS could achieve a record annual average punctuality level (86.8 percent). Normally, NS is only allowed to increase fares on a par with the inflation rate. NS achieved this high punctuality record in 2007 and was permitted the additional fare increase as of February 2008. The result was an additional annual profit of about 20 million Euros from 2008 onwards.

Moreover, we estimate the savings of the optimized rolling-stock schedules over manually constructed schedules to be 6 percent. NS achieved these savings by the introduction of ROSA on part of the long-distance network in 2005. At that time, we compared the schedules that ROSA generated with the manually generated schedules. Six percent corresponds to an annual savings of 18 million Euros. NS invested these savings primarily in improving the seat availability for the passengers, resulting in

higher customer satisfaction.

We estimate the benefits of TURNI and its extensions at another 12 million Euros per year. TURNI has been in use since 2002. In its first year of operation, we compared the automatically generated crew schedules with the manual ones. Applying exactly the same rules, we obtained an improvement of 2 percent. By improving the algorithm in the following years, we gained another 2 percent improvement.

Overall, NS has been able to reduce the number of drivers per train-kilometer by approximately 15 percent because NS was able to construct schedules that permitted adjustments to labor standards and regulations. It is clear that without using TURNI NS could not have achieved this effect; however, it is hard to measure which part of the additional 11 (15-2-2) percent is the TURNI contribution.

Adding up all the quantifiable benefits, we find that the total annual additional profit is approximately 70 (20+20+18+12) million Euros.

Finally, the new timetable is having a positive impact on the Dutch society as a whole. The independent adviser of the Dutch government, Centraal Planbureau (CPB), estimates the direct benefits to the Dutch economy to be at 8 million Euros per year for every percentage-point increase in punctuality. More importantly, because of the new timetable, an additional increase in railway transport will be possible without further significant infrastructure extensions, which would require huge investments. For example, doubling the number of tracks on a route of approximately 40 kilometers would cost about 1 billion Euros. The new timetable will be able to accommodate additional railway transport in the future with only limited infrastructure additions; therefore, the railway system will be able to facilitate the growing demand for transportation to the main cities during rush hours. This will help to reduce the pressure on the roads into and inside these cities and to replace car traffic by rail traffic, thereby reducing the pollution from greenhouse gases.

Bibliography

ABBINK, E. (2006). *Intelligent Shunting: Dealing with Constraints (Satisfaction)*, pages 391–413. John Wiley and Sons, Inc. ISBN 9780471781264.

ABBINK, E., ALBINO, L., DOLLEVOET, T., HUISMAN, D., ROUSSADO, J., SALDANHA, R. (2011). *Solving large scale crew scheduling problems in practice*. Public Transport, 3:149–164. ISSN 1866-749X. 10.1007/s12469-011-0045-x.

ABBINK, E., MOBACH, D., FIOOLE, P., KROON, L., HEIJDEN, E. VAN DER, WIJNGAARDS, N. (2009). *Actor-agent application for train driver rescheduling*. In *Proceedings of the Eighth International Conference on Autonomous Agents and Multiagent Systems*, pages 513–520. International Foundation for Autonomous Agents and Multiagent Systems.

ABBINK, E., MOBACH, D., FIOOLE, P., KROON, L., VAN DER HEIJDEN, E., WIJNGAARDS, N. (2010). *Real-time train driver rescheduling by actor-agent techniques*. Public Transport, 2:249–268. ISSN 1866-749X. 10.1007/s12469-010-0033-6.

ABBINK, E., MOBACH, D., FIOOLE, P., KROON, L., WIJNGAARDS, N., VAN DER HEIJDEN, E. (2008a). *Actor-agent based approach to train driver rescheduling*. In NIJHOLT, A., PANTIC, M., POEL, M., HONDORP, H. (editors), *Proceedings of the 20th Belgian-Dutch Conference on Artificial Intelligence (BNAIC 2008)*, pages 1–8. URL www.bnaic.org.

ABBINK, E., VAN 'T WOUT, J., HUISMAN, D. (2007). *Solving large scale crew scheduling problems by using iterative partitioning*. In LIEBCHEN, C., AHUJA, R.K., MESA, J.A. (editors), *ATMOS 2007 - 7th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems*. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI),

Schloss Dagstuhl, Germany, Dagstuhl, Germany. ISBN 978-3-939897-04-0. URL `http://drops.dagstuhl.de/opus/volltexte/2007/1168`.

ABBINK, E., VAN 'T WOUT, J., HUISMAN, D (2008b). *Solving Large Scale Crew Scheduling Problems by using Iterative Partitioning*. Technical Report Report EI2008-03, Econometric Institute, Erasmus University Rotterdam.

ABBINK, E.J.W, FISCHETTI, M., KROON, L.G., TIMMER, G., VROMANS, M.J.C.M. (2005). *Reinventing Crew Scheduling at Netherlands Railways*. Interfaces, 35(5):393–401.

ABBINK, E.J.W, VAN DEN BERG, B.W.V., KROON, L.G., SALOMON, M. (2004). *Allocation of Railway Rolling Stock for Passenger Trains*. Transportation Science, 38(1):33–42.

AHUJA, R. K., MAGNANTI, T. L., ORLIN, J. B. (1993). *Network flows: theory, algorithms and applications*. Prentice Hall.

ANTHONY, ROBERT N. (1965). *Planning and Control Systems: A Framework for Analysis*. Technical report, Harvard University Graduate School of Business Administration, Boston.

BARNHART, C., JOHNSON, E.L., NEMHAUSER, G.L., SAVELSBERGH, M.W.P., VANCE, P.H. (1998). *Branch-and-Price: Column Generation for Solving Huge Integer Programs*. Operations Research, 46:316–329.

BEASLEY, J. E. (1990). *A Lagrangian Heuristic for Set-Covering Problems*. Naval Research Logistics, 37(151–164).

BEASLEY, J.E. (1993). *Lagrangian relaxation*, pages 243–303. John Wiley & Sons, Inc., New York, NY, USA. ISBN 0-470-22079-1.

BECKHARD, R., HARRIS, R.T. (1987). *Organizational transitions: managing complex change*. Addison-Wesley series on organization development. Addison-Wesley Pub. Co. ISBN 9780201108873.

BENOIST, T., GARDI, F., JEANJEAN, A. (2012). *Lessons learned from 15 years of operations research for french tv channel tf1*. Interfaces, 42(6):577–584.

CAPRARA, A., FISCHETTI, M., TOTH, P. (1999). *A Heuristic Algorithm for the Set Covering Problem*. Operations Research, 47:730–743.

CAPRARA, A., FISCHETTI, M., TOTH, P., VIGO, D. (1998). *Modeling and Solving the Crew Rostering Problem*. Operations Research, 46:820–830.

CAPRARA, A., FISCHETTI, M., TOTH, P., VIGO, D., GUIDA, P.L. (1997). *Algorithms for railway crew management.*. Mathematical Programming, 79(1–3):125–141.

CASTRO, A. JM, OLIVEIRA, E. (2011). *Airline operations control: A new concept for operations recovery*. Airline Industry: Strategies, Operations and Safety.

CAWSEY, T.F., INGOLS, C., CAWSEY, T.F., DESZCA, G. (2011). *Organizational Change: An Action-Oriented Toolkit*. SAGE Publications. ISBN 9781412982856.

CHVÁTAL, V. (1983). *Linear Programming*. W.H. Freeman and Company.

DANTZIG, GEORGE B., WOLFE, PHILIP (1960). *Decomposition principle for linear programs*. Operations Research, 8:101–111.

DE WEERDT, M., TER MORS, A., WITTEVEEN, C. (2005). *Multi-agent planning: An introduction to planning and coordination*. In *In: Handouts of the European Agent Summer*. Citeseer.

DESAULNIERS, G., DESROSIERS, J., SOLOMON, M. M. (2005). *Column Generation*. Springer, New York.

DESJARDINS, M.E., DURFEE, E.H., ORTIZ, C.L., WOLVERTON, M.J. (2000). *A survey of research in distributed, continual planning*. AI Magazine, 20(4):13–22.

DESROSIERS, J., DUMAS, Y., SOLOMON, M.M., SOUMIS, F. (1995). *Time Constrained Routing and Scheduling*. In BALL, M.O. (editor), *Handbooks in Operations Research and Management Science*, volume 8: Network Routing, pages 35–139. Elsevier.

DESROSIERS, J., LÜBBECKE, MARCO E. (2005). *A Primer in Column Generation*. In DESAULNIERS, G., DESROSIERS, J., SOLOMON, M. M. (editors), *Column Generation*. Springer, New York.

DOLLEVOET, T. (2013). *Delay Management and Dispatching in Railways*. Ph.D. thesis, Erasmus Research Institute of Management.

DUTTA, G. (2000). *Lessons for success in or/ms practice gained from experiences in indian and us steel plants*. Interfaces, 30(5):23–30.

FIOOLE, P.J., KROON, L., MARÓTI, G., SCHRIJVER, A. (2006). *A Rolling Stock Circulation Model for Combining and Splitting of Passenger Trains*. European Journal of Operational Research, 174:1281–1297.

FISHER, M.L. (1981). *The lagrangian relaxation method for solving integer programming problems*. Management Science, 27(1).

FLEUREN, H., GOOSSENS, C., HENDRIKS, M., LOMBARD, M., MEUFFELS, I., POPPELAARS, J. (2013). *Supply chainwide optimization at tnt express*. Interfaces, 43(1):5–20.

FORES, S., PROLL, L., WREN, A. (2001). *Experiences with a Flexible Driver Scheduler*. In VOSS, S., DADUNA, J.R. (editors), *Computer-Aided Scheduling of Public Transport*, pages 137–152. Springer, Berlin, Germany.

GEOFFRION, A.M (1974). *Lagrangean relaxation for integer programming*. Mathematical Programming Study, 2:82–114.

HARTOG, A. (2005). *Wiskundige modellen voor het maken van dienstroosters voor het treinpersoneel*. Master's thesis, Free University Amsterdam, The Netherlands. In Dutch.

HARTOG, A., HUISMAN, D., ABBINK, E., KROON, L. (2009). *Decision support for crew rostering at ns*. Public Transport, 1:121–133. ISSN 1866-749X. 10.1007/s12469-009-0009-6.

HELD, M., KARP, R.M. (1971). *The traveling-salesman problem and minimum spanning trees: Part ii*. Mathematical Programming, 1(1):6–25. ISSN 0025-5610.

HELSINGER, A., THOME, M., WRIGHT, T. (2004). *Cougaar: a scalable, distributed multi-agent architecture*. In *Systems, Man and Cybernetics, 2004 IEEE International Conference on*, volume 2, pages 1910–1917 vol.2. ISSN 1062-922X.

HOFFMAN, K.L., PADBERG, M. (1993). *Solving Airline Crew Scheduling Problems by Branch-and-Cut.*. Management Science, 39:657–682.

HUISMAN, D. (2004). *Integrated and dynamic vehicle and crew scheduling*. Ph.D. thesis, Tinbergen Institute, Erasmus University Rotterdam.

HUISMAN, D. (2007). *A Column Generation Approach for the Rail Crew Rescheduling Problem*. European Journal of Operational Research, 180(1):163–173.

HUISMAN, D., JANS, R., PETERS, M., WAGELMANS, A. P. M. (2005). *Combining Column Generation and Lagrangean Relaxation*. In DESAULNIERS, G., DESROSIERS, J., SOLOMON, M. M. (editors), *Column Generation*, GERAD 25th Anniversary Series, pages 247–270. Springer, New York. ISBN 978-0-387-25485-2.

JESPERSEN-GROTH, J., POTTHOFF, D., CLAUSEN, J., HUISMAN, D., KROON, L.G., MARÓTI, G., NIELSEN, M.N. (2009). *Disruption management in passenger railway transportation*. In *Robust and online large-scale optimization*, pages 399–421. Springer.

KERNIGHAN, B., LIN, S. (1970). *An Efficient Heuristic Procedure for Partitioning Graphs.*. Bell Systems Technical Journal, 29:291–307.

KOHL, N. (2003). *Solving the World's Largest Crew the Scheduling Problem*. ORbit Extra, pages 8–12. Danish Operations Research Society.

KOHL, N., KARISH, S.E. (2004). *Airline Crew Rostering: Problem Types, Modeling and Optimization*. Annals of Operations Research, 127:223–257.

KROON, L., HUISMAN, D., ABBINK, E., FIOOLE, P., FISCHETTI, M., MARÓTI, G., SCHRIJVER, A., STEENBEEK, A., YBEMA, R. (2009). *The New Dutch Timetable: The OR Revolution*. Interfaces, 39(1):6–17. 10.1287/inte.1080.0409.

KROON, L.G., FISCHETTI, M. (2001). *Crew Scheduling for Netherlands Railways "Destination: Customer"*. In VOSS, S., DADUNA, J.R. (editors), *Computer-Aided Scheduling of Public Transport*, pages 181–201. Springer, Berlin, Germany.

KWAN, A.S.K., KWAN, R.S.K., PARKER, A., M.E.AND WREN (1999). *Producing train driver crew schedules under different operating strategies.*. In WILSON, N.H.M. (editor), *Computer Aided Transit Scheduling*, pages 129–154. Springer-Verslag.

LENSTRA, J.K., RINNOOY KAN, A.H.G. (1979). *Computational Complexity of Discrete Optimization Problems*. Annals of Discrete Mathematics, 4:121–140.

MAO, X., TER MORS, A., ROOS, N., WITTEVEEN, C. (2007). *Coordinating Competitive Agents in Dynamic Airport Resource Scheduling.*. In PETTA, P., MUELLER, J.P., KLUSCH, M., GEORGEFF, M. (editors), *Proc. of the 5th German Conf. on Multiagent System Technologies*, pages 133–144. LNAI, Springer Verslag.

MARÓTI, G. (2006). *Operations research models for railway rolling stock planning*. Ph.D. thesis, Eindhoven University of Technology, Eindhoven, The Netherlands.

MOBACH, D., ABBINK, E., FIOOLE, P., KROON, L., HEIJDEN, E. VAN DER, WIJNGAARDS, N. (2009). *Train driver rescheduling using task-exchange teams*. In *Proceedings of the Second International Workshop on: Optimisation in Multi-Agent Systems (OPTMAS) at AAMAS 2009*.

MORGADO, E.M., MARTINS, J.P. (1998). *CREWS_NS: Scheduling train crews in The Netherlands.*. AI Magazine, 19(1):25–38.

NIELSEN, L.K. (2011). *Rolling stock rescheduling in passenger railways: Applications in short-term planning and in disruption management*. Ph.D. thesis, E.

NIGAM, R. (2008). *Structuring and sustaining excellence in management science at merrill lynch*. Interfaces, 38(3):202–209.

NS WEBPAGE (2013). *Ns jaarverslag 2013*. Webpage. URL `nsjaarverslag.nl/jaarverslag-2013`.

POTTHOFF, D. (2010). *Railway crew rescheduling: Novel approaches and extensions*. Ph.D. thesis, Erasmus University Rotterdam.

POTTHOFF, D., HUISMAN, D., DESAULNIERS, G. (2010). *Column generation with dynamic duty selection for railway crew rescheduling*. TRANSPORTATION SCIENCE, 44(4):493–505.

REZANOVA, N. J., RYAN, D. M. (2010). *The train driver recovery problem - A set partitioning based model*. Computers & Operations Research, 37:845–856.

SANDHOLM, T., LESSER, V. (2002). *Leveled-commitment contracting: a backtracking instrument for multiagent systems*. AI Mag., 23(3):89–100. ISSN 0738-4602.

SCHRIJVER, A., STEENBEEK, A. (1993). *Spoorweg Dienstregelingontwikkeling (Railway timetable development)*. Technical report, Centrum voor Wiskunde en Informatica, Amsterdam, Netherlands. URL `http://homepages.cwi.nl/~lex/files/rapp1a.pdf`. In Dutch.

SHIBGHATULLAH, A.S., ELDABI, T., RZEVSKI, G. (2006). *A Framework for Crew Scheduling Management System Using Multi-Agents System.*. In *28th Int. Conf. on Information Technology Interfaces (ITI 2006)*. Cavtat.

SNIJDERS, H., WALLE, T. VAN DE, HUISMAN, D. (2013). *De Besliskunde achter de Aangepaste Dienstregeling van NS*. STAtOR, 14(2):9–12.

SODHI, M., NORRIS, S. (2004). *A Flexible, Fast, and Optimal Modeling Approach Applied to Crew Rostering at London Underground*. Annals of Operations Research, 127:259–281.

TRANVOUEZ, E., FERRARINI, A. (2006). *MultiAgent Modelling of Cooperative Disruption Management in Supply Chains.*. In *Int. Conf. on Service Systems and Service Management (2006)*, pages 853–858. Troyes.

VAN 'T WOUT, J. (2007). *Crew Scheduling at Netherlands Railways: using TURNI effectively*. Master's thesis, Faculty of Economics, Erasmus University Rotterdam, The Netherlands.

WEDELIN (1995). *An algorithm for large scale 0-1 integer programming with application to airline crew scheduling.*. Annals of Operations Research, 57:283–301.

WIJNGAARDS, N., KEMPEN, M., SMIT, A., NIEUWENHUIS, K. (2006). *Towards Sustained Team Effectiveness.*. In LINDEMANN, G. ET AL. (editor), *Selected revised papers from the workshops on Norms and Institutions for Regulated Multi-Agent Systems (ANIREM) and Organizations and Organization Oriented Programming at AAMAS05*, volume 3913, pages 33–45. LNGS, Springer Verslag.

Nederlandse samenvatting (Summary in Dutch)

In 2001 was er een publieke discussie over de inzet van machinisten en conducteurs (rijdend personeel) van de Nederlandse Spoorwegen. Aanleiding was een gewijzigde manier van het inzetten van het rijdend personeel, in de volksmond het Rondje rond de kerk genoemd. Kern van deze aangepaste inzet was het meer lijngebonden inzetten van rijdend personeel. Hierbij werd een personeelslid in het eerste deel van een dienst op een baanvak ingezet en na de (lunch)pauze op een ander baanvak.

Het management beoogde met deze aanpassing te zorgen voor meer binding met het product op de baanvakken waar de machinisten en conducteurs nu relatief vaker kwamen. Daarnaast was het idee dat een lokale verstoring niet doorwerkt op andere baanvakken omdat het personeel op het baanvak blijft, en alleen wisselt van baanvak na de pauze. Het rijdend personeel was tegen deze nieuwe manier van inzetten, zij vonden het werk eentonig, wat zou leiden tot een verminderde aandacht voor de seinen en potentiele ongevallen tot gevolg. De discussie werd zo heftig dat er uiteindelijk gestaakt werd en de ondernemingsraad van NS gevraagd werd om een alternatief inzetmodel voor het rijdend personeel te ontwikkelen.

Met behulp van geavanceerde wiskundige systemen (beslissingsondersteunende systemen) werden in opdracht van de ondernemingsraad een groot aantal alternatieve sets van inzetregels doorgerekend en voorgelegd aan het rijdend personeel. Uiteindelijk werd voor het Lusten & Lasten delen model gekozen, wat erop gericht is om een goede verdeling van het aantrekkelijke en minder aantrekkelijke werk te hebben. Deze set regels was te complex om nog met de hand op te lossen en heeft mede geleidt tot het structureel invoeren van wiskundige modellen ter ondersteuning van het plannen van het rijdend personeel.

Dit proefschrift richt zich op de vraag hoe het personeelsplanningsproces van een

(spoor)vervoerder verbeterd kan worden, met daarbij de nadruk op het introduceren van geavanceerde systemen ter ondersteuning van dit proces. We kijken niet alleen naar het maken van diensten maar ook naar het maken van roosters en naar het aanpassen van de diensten in het geval van verstoringen op de dag zelf. We onderzoeken de kwaliteitsaspecten van een personeelsplan, verder kijken we welke technieken beschikbaar zijn en hoe we die moeten toepassen bij het maken van een personeelsplanning. Ook geven we een beschouwing over het veranderproces waarmee de invoering van een dergelijk systeem gepaard gaat. Als laatste beschrijven we wat het effect is van het verbeteren van de ondersteuning van het personeels-planningsproces.

Na een introductie in hoofdstuk 1, geven we in hoofdstuk 2 een overzicht van de planningsvraagstukken van een (spoor)vervoerder, in het bijzonder van NS. Hierbij geven we inzicht in de bijzonderheden van deze vraagstukken. Dit hoofdstuk is bedoeld als een gedegen introductie tot de planningsvraagstukken die binnen NS worden bestudeerd, gemodelleerd en voorzien van geavanceerde wiskundige algoritmen. Ze geven daarmee de context waarin het personeelsplanningsproces binnen NS uitgevoerd wordt.

Hoofdstuk 3, 4 en 5 zijn hoofdstukken die gaan over het modeleren en oplossen van onderdelen van het personeelsplanningsvraagstuk. In ieder hoofdstuk geven we een overzicht van de voor dat vraagstuk relevante theorie, beschrijven we hoe wij het probleem gemodelleerd en opgelost hebben. Hierbij beschrijven we hoe de specifieke en complexe kenmerken van de problemen effectief zijn opgenomen in de modellen en algoritmen. We laten zien dat de vraagstukken goed op te lossen zijn met geavanceerde wiskundige technieken. Dit wordt aangetoond met praktijkvoorbeelden die efficiënt en effectief opgelost worden met deze technieken.

Hoofdstuk 3 gaat over het plannen van personeelsdiensten. Hiervoor is er veel literatuur te vinden in het domein van de luchtvaart. Daar is het gebruikelijk om de diensten van piloten en cabinepersoneel geautomatiseerd te plannen. Het grote verschil met het plannen van diensten voor de spoorwegen, is dat er gemiddeld meer taken in een dienst zitten en dat het aantal combinatie mogelijkheden een orde groter is. Dit geldt in het bijzonder voor de Nederlandse situatie, waar veel treinen over een beperkte afstand rijden en waar er veel tussenliggende stations zijn waar van personeel gewisseld kan worden. Dit maakt dat de technieken uit de luchtvaart lange tijd niet toegepast konden worden in het domein van het spoorvervoer. Ontwikkelingen in de techniek hebben ervoor gezorgd dat het heden ten dage wel mogelijk is. We beschrijven hoe deze technieken (bijvoorbeeld kolomgeneratie en Lagrange Relaxatie) succesvol kunnen worden aangepast zodat ze toepasbaar zijn voor het maken van de

personeelsdiensten.

Hoofdstuk 4 gaat over het maken van personeelsroosters, het toewijzen van de gemaakte diensten aan de individuele personeelsleden. Voor het maken van de roosters binnen NS wordt gebruikt gemaakt van het concept van roulerende roosters. Voor het oplossen van wiskundige modellen voor dit type roosters is er nog niet veel literatuur beschikbaar. We beschrijven hoe we de randvoorwaarden en voorkeuren kunnen modeleren in een algemeen wiskundig (Mixed Integer Programming) model, welke we vervolgens met een commercieel beschikbaar softwarepakket kunnen oplossen. Dit modeleren en oplossen vindt in drie stappen plaats en daarmee krijgen we een goed oplosbaar probleem. De resultaten van dit onderzoek zijn als experiment voorgelegd aan een drietal personeels(rooster)groepen die de keuze hadden uit een handmatig geconstrueerd rooster en een die door het systeem gegenereerd was. In alle gevallen werd het door het systeem gegenereerde rooster beoordeeld als het beste alternatief. Verdere invoering van het systeem is stilgelegd vanwege de wens om van de methodiek met roulerende roosters naar een nieuwe methodiek met roosters per individu over te gaan. De bedoeling is om deze overstap in de komende jaren te maken.

Hoofdstuk 5 gaat over het aanpassen van personeelsdiensten in het geval van een calamiteit op de dag van uitvoering. Als bijvoorbeeld treinen uitvallen, is het vaak zo dat de machinist en/of conducteur niet meer op tijd zal zijn om de volgende trein te halen die ingepland is in de dienst. In het algemeen wordt dit gezien als een van de meest complexe puzzels op de dag van uitvoering. Voor het oplossen van dit probleem hebben we gekeken of we het actor-agent paradigma kunnen toepassen. In de literatuur wordt dit gebruikt voor diverse vraagstukken, maar is het nog nooit gebruikt het vraagstuk wat wij bestudeerd hebben. Naast de hier gepresenteerde aanpak heeft er apart een promotieonderzoek plaatsgevonden waarbij gekeken is naar de toepassing van een alternatieve methode, kolomgeneratie en Lagrange Relaxatie, zoals ook beschreven in hoofdstuk 3. Wij vergelijken beide aanpakken en constateren dat ze beide goede oplossingen leveren. Voor grote verstoringen werkt de alternatieve aanpak net iets beter en daarom is besloten dit in de praktijk toe te gaan passen.

Hoofdstuk 6 is van een bedrijfskundige aard. Aan de hand van een algemeen raamwerk uit de veranderkunde wordt in retrospectief beschreven hoe de eerste succesvolle invoering van beslissingsondersteuning voor het plannen van diensten voor rijdend personeel voltooid is. Gekeken wordt waarom verandering gewenst was, hoe de beweging in gang gezet is, en wat het uiteindelijke resultaat is. Zoals het raamwerk ook beschrijft, is verandering een herhaling van meerdere verander cycli, welke

binnen NS de vorm hebben gehad van verbeteringen van de ondersteuning voor het plannen van diensten, en daarnaast ook voor het plannen van roosters en voor de bijsturing van diensten bij calamiteiten op de dag van uitvoering. Aan het einde van hoofdstuk 6 worden, na de veranderingen te hebben beschreven, lessen getrokken uit de aanpak en de resultaten van de ontwikkeling en invoering van beslissingsondersteunende systemen bij NS. Deze lessen worden vergeleken met enkele publicaties over soortgelijke lessen over de succesvolle implementatie van besliskundige modellen en systemen bij andere bedrijven. Deze lessen en beschouwingen kunnen gebruikt worden door managers en besliskundigen die zich bezig houden met de ontwikkeling en de invoering van beslissingsondersteunende systemen in de praktijk.

Samenvattend presenteren we in dit proefschrift diverse modellen en algoritmen ter ondersteuning van het personeelsplanningsproces. Ook geven we een meer bedrijfskundige beschouwing van het daaraan gerelateerde verandertraject en de lessen die geleerd zijn bij het invoeren van deze systemen in de praktijk.

De introductie van de ondersteuning voor het plannen van de diensten van rijdend personeel heeft geleid tot een stabiele arbeidsrelatie binnen de NS. Daarnaast is de doorlooptijd aanzienlijk verkort en zijn we in staat om in enkele uren een plan te maken in plaats van de maanden die we voor de introductie van deze ondersteuning nodig hadden. Dit maakt het mogelijk om scenario analyses te doen, bijvoorbeeld voor het aanpassen van arbeidsregels. Ook zijn we in staat om in korte tijd de diensten aan te passen, bijvoorbeeld als NS een aangepaste dienstregeling gaat rijden vanwege verwachte winterse omstandigheden.

Tenslotte is het sinds kort mogelijk op de dag zelf ook het aanpassen van de personeelsdiensten te ondersteunen met geavanceerde beslissings-ondersteunende systemen. Hiermee zullen in het geval van stremmingen minder treinen uitvallen en ook minder treinen vertraagd worden en zijn we in staat om in zeer korte tijd de actuele personeelsplanning aan te passen.

About the author

In 1995, Erwin Abbink[1] obtained a Master of Science degree in Econometrics, at the University of Groningen (Rijksuniversiteit Groningen). In 2000, he obtained a Master of Science degree in Information and Knowledge Technology at Middlesex University in London.

Currently, he is a managing consultant innovation at Netherlands Railways. His main focus is on Decision Support Systems and Advanced Analytics. He has a broad experience in rail logistics and on disruption management for railway systems.

Based on his work, he has published papers in journals like Transportation Science, Interfaces, and Public Transport. He published on rolling stock assignment , shunting , crew scheduling , crew rostering and on crew dispatching.

In 2004 he was a finalist in the Daniel H. Wagner Prize for Excellence in Operations Research Practice with the paper: (Abbink et al. (2005)) *Reinventing Crew Scheduling at Netherlands Railways*. In 2008 he was winner of the Franz Edelman Excellence in Practice award with the paper: (Kroon et al. (2009)) *The New Dutch Timetable: The O.R. Revolution*. In 2009 he was awarded with the Best Applied Paper Award at the BNAIC, for the paper: (Abbink et al. (2009)): *Actor-agent based approach to train driver rescheduling*.

His main research interest during his work at Netherlands Railways is in applying Operations Research in practice.

---

[1]Process quality & Innovation, Netherlands Railways, P.O.Box 2025, NL-3500 HA Utrecht, The Netherlands, erwin.abbink@ns.nl

Erasmus Research Institute of Management (ERIM)

ERIM Ph.D. Series Research in Management

The ERIM PhD Series contains PhD dissertations in the field of Research in Management defended at Erasmus University Rotterdam and supervised by senior researchers affiliated to the *Erasmus Research Institute of Management (ERIM)*. All dissertations in the ERIM PhD Series are available in full text through the ERIM Electronic Series Portal: http://hdl.handle.net/1765/1

ERIM is the joint research institute of the Rotterdam School of Management (RSM) and the Erasmus School of Economics at the Erasmus University Rotterdam (EUR).

DISSERTATIONS LAST FIVE YEARS

Acciaro, M., *Bundling Strategies in Global Supply Chains,* Promotor(s): Prof.dr. H.E. Haralambides,
EPS-2010-197-LIS, http://hdl.handle.net/1765/19742

Akpinar, E., *Consumer Information Sharing; Understanding Psychological Drivers of Social Transmission,* Promotor(s): Prof.dr.ir. A. Smidts,
EPS-2013-297-MKT, http://hdl.handle.net/1765/50140

Alexiev, A., *Exploratory Innovation: The Role of Organizational and Top Management Team Social Capital,* Promotor(s): Prof.dr. F.A.J. van den Bosch & Prof.dr. H.W. Volberda,
EPS-2010-208-STR, http://hdl.handle.net/1765/20632

Akin Ates, M., *Purchasing and Supply Management at the Purchase Category Level: Strategy, Structure, and Performance,* Promotor(s): Prof.dr. J.Y.F. Wynstra, EPS-2014-300-LIS, `http://hdl.handle.net/1765/1`

Almeida, R.J.de, *Conditional Density Models Integrating Fuzzy and Probabilistic Representations of Uncertainty,* Promotor Prof.dr.ir. Uzay Kaymak, EPS-2014-310-LIS, `http://hdl.net/1765/1`

Bannouh, K., *Measuring and Forecasting Financial Market Volatility using High-Frequency Data,* Promotor(s): Prof.dr. D.J.C. van Dijk, EPS-2013-273-F&A, `http://hdl.handle.net/1765/38240`

Benning, T.M., *A Consumer Perspective on Flexibility in Health Care: Priority Access Pricing and Customized Care,* Promotor(s): Prof.dr.ir. B.G.C. Dellaert, EPS-2011-241-MKT, `http://hdl.handle.net/1765/23670`

Ben-Menahem, S.M., *Strategic Timing and Proactiveness of Organizations,* Promotor(s): Prof.dr. H.W. Volberda & Prof.dr.ing. F.A.J. van den Bosch, EPS-2013-278-S&E, `http://hdl.handle.net/1765/39128`

Berg, W.E. van den, *Understanding Salesforce Behavior Using Genetic Association Studies,* Promotor(s): Prof.dr. W.J.M.I. Verbeke, EPS-2014-311-MKT, `http://hdl.handle.net/1765/51440`

Betancourt, N.E., *Typical Atypicality: Formal and Informal Institutional Conformity, Deviance, and Dynamics,* Promotor(s): Prof.dr. B. Krug, EPS-2012-262-ORG, `http://hdl.handle.net/1765/32345`

Binken, J.L.G., *System Markets: Indirect Network Effects in Action, or Inaction,* Promotor(s): Prof.dr. S. Stremersch, EPS-2010-213-MKT, `http://hdl.handle.net/1765/21186`

Blitz, D.C., *Benchmarking Benchmarks,* Promotor(s): Prof.dr. A.G.Z. Kemna & Prof.dr. W.F.C. Verschoor, EPS-2011-225-F&A, `http://hdl.handle.net/1765/226244`

Boons, M., *Working Together Alone in the Online Crowd: The Effects of Social Motivations and Individual Knowledge Backgrounds on the Participation and Performance of Members of Online Crowdsourcing Platforms,* Promotor(s): Prof.dr. H.G. Barkema, EPS-2014-306-S&E, `http://hdl.net/1765/50711`

Borst, W.A.M., *Understanding Crowdsourcing: Effects of Motivation and Rewards on Participation and Performance in Voluntary Online Activities,* Promotor(s): Prof.dr.ir. J.C.M. van den Ende & Prof.dr.ir. H.W.G.M. van Heck, EPS-2010-221-LIS, `http://hdl.handle.net/1765/21914`

Budiono, D.P., *The Analysis of Mutual Fund Performance: Evidence from U.S. Equity Mutual Funds,* Promotor(s): Prof.dr. M.J.C.M. Verbeek, EPS-2010-185-F&A, `http://hdl.handle.net/1765/18126`

Burger, M.J., *Structure and Cooptition in Urban Networks,* Promotor(s): Prof.dr. G.A. van der Knaap & Prof.dr. H.R. Commandeur,
EPS-2011-243-ORG, `http://hdl.handle.net/1765/26178`

Byington, E., *Exploring Coworker Relationships: Antecedents and Dimensions of Interpersonal Fit, Coworker Satisfaction, and Relational Models,* Promotor(s): Prof.dr. D.L. van Knippenberg,
EPS-2013-292-ORG, `http://hdl.handle.net/1765/41508`

Camacho, N.M., *Health and Marketing; Essays on Physician and Patient Decision-making,* Promotor(s): Prof.dr. S. Stremersch,
EPS-2011-237-MKT, `http://hdl.handle.net/1765/23604`

Cankurtaran, P., *Essays On Accelerated Product Development,* Promotor(s): Prof.dr.ir. G.H. van Bruggen,
EPS-2014-317-MKT, `http://hdl.handle.net/1765/1`

Caron, E.A.M., *Explanation of Exceptional Values in Multi-dimensional Business Databases,* Promotor(s): Prof.dr.ir. H.A.M. Daniels & Prof.dr. G.W.J. Hendrikse,
EPS-2013-296-LIS, `http://hdl.handle.net/1765/50005`

Carvalho, L., *Knowledge Locations in Cities; Emergence and Development Dynamics,* Promotor(s): Prof.dr. L. van den Berg,
EPS-2013-274-S&E, `http://hdl.handle.net/1765/38449`

Carvalho de Mesquita Ferreira, L., *Attention Mosaics: Studies of Organizational Attention,* Promotor(s): Prof.dr. P.M.A.R. Heugens & Prof.dr. J. van Oosterhout,
EPS-2010-205-ORG, `http://hdl.handle.net/1765/19882`

Cox, R.H.G.M., *To Own, To Finance, and to Insure; Residential Real Estate Revealed,* Promotor(s): Prof.dr. D. Brounen,
EPS-2013-290-F&A, `http://hdl.handle.net/1765/40964`

Defilippi Angeldonis, E.F., *Access Regulation for Naturally Monopolistic Port Terminals: Lessons from Regulated Network Industries,* Promotor(s): Prof.dr. H.E. Haralambides,
EPS-2010-204-LIS, `http://hdl.handle.net/1765/19881`

Deichmann, D., *Idea Management: Perspectives from Leadership, Learning, and Network Theory,* Promotor(s): Prof.dr.ir. J.C.M. van den Ende,
EPS-2012-255-ORG, `http://hdl.handle.net/1765/31174`

Desmet, P.T.M., *In Money we Trust? Trust Repair and the Psychology of Financial Compensations,* Promotor(s): Prof.dr. D. De Cremer & Prof.dr. E. van Dijk,
EPS-2011-232-ORG, `http://hdl.handle.net/1765/23268`

Dietvorst, R.C., *Neural Mechanisms Underlying Social Intelligence and Their Relationship with the Performance of Sales Managers,* Promotor(s): Prof.dr. W.J.M.I. Verbeke,
EPS-2010-215-MKT, `http://hdl.handle.net/1765/21188`

Dollevoet, T.A.B., *Delay Management and Dispatching in Railways,* Promotor(s): Prof.dr. A.P.M. Wagelmans,
EPS-2013-272-LIS, `http://hdl.handle.net/1765/38241`

Doorn, S. van, *Managing Entrepreneurial Orientation,* Promotor(s): Prof.dr. J.J.P. Jansen, Prof.dr.ing. F.A.J. van den Bosch & Prof.dr. H.W. Volberda,
EPS-2012-258-STR, `http://hdl.handle.net/1765/32166`

Douwens-Zonneveld, M.G., *Animal Spirits and Extreme Confidence: No Guts, No Glory,* Promotor(s): Prof.dr. W.F.C. Verschoor,
EPS-2012-257-F&A, `http://hdl.handle.net/1765/31914`

Duca, E., *The Impact of Investor Demand on Security Offerings,* Promotor(s): Prof.dr. A. de Jong,
EPS-2011-240-F&A, `http://hdl.handle.net/1765/26041`

Duursema, H., *Strategic Leadership; Moving Beyond the Leader-follower Dyad,* Promotor(s): Prof.dr. R.J.M. van Tulder,
EPS-2013-279-ORG, `http://hdl.handle.net/1765/39129`

Eck, N.J. van, *Methodological Advances in Bibliometric Mapping of Science,* Promotor(s): Prof.dr.ir. R. Dekker,
EPS-2011-247-LIS, `http://hdl.handle.net/1765/26509`

Essen, M. van, *An Institution-Based View of Ownership,* Promotor(s): Prof.dr. J. van Oosterhout & Prof.dr. G.M.H. Mertens,
EPS-2011-226-ORG, `http://hdl.handle.net/1765/22643`

Feng, L., *Motivation, Coordination and Cognition in Cooperatives,* Promotor(s): Prof.dr. G.W.J. Hendrikse,
EPS-2010-220-ORG, `http://hdl.handle.net/1765/21680`

Fourn, S. P. L., *Managing Organizational Tensions: A Multi-level Perspective on Exploration, Exploitation, and Ambidexterity,* Promotor(s): Prof.dr. J.J.P. Jansen, Prof.dr. S.J. Magala & dr. T.J.M.Mom,
EPS-2014-318-S&E, `http://hdl.handle.net/1765/21680`

Gharehgozli, A.H., *Developing New Methods for Efficient Container Stacking Operations,* Promotor(s): Prof.dr.ir. M.B.M. de Koster,
EPS-2012-269-LIS, `http://hdl.handle.net/1765/37779`

Gils, S. van, *Morality in Interactions: On the Display of Moral Behavior by Leaders and Employees,* Promotor(s): Prof.dr. D.L. van Knippenberg,
EPS-2012-270-ORG, `http://hdl.handle.net/1765/38028`

Ginkel-Bieshaar, M.N.G. van, *The Impact of Abstract versus Concrete Product Communications on Consumer Decision-making Processes,* Promotor(s): Prof.dr.ir. B.G.C. Dellaert,
EPS-2012-256-MKT, `http://hdl.handle.net/1765/31913`

Gkougkousi, X., *Empirical Studies in Financial Accounting,* Promotor(s): Prof.dr. G.M.H. Mertens & Prof.dr. E. Peek,
EPS-2012-264-F&A, `http://hdl.handle.net/1765/37170`

Hakimi, N.A., *Leader Empowering Behaviour: The Leaders Perspective: Understanding the Motivation behind Leader Empowering Behaviour,* Promotor(s): Prof.dr. D.L. van Knippenberg,
EPS-2010-184-ORG, `http://hdl.handle.net/1765/17701`

Hensmans, M., *A Republican Settlement Theory of the Firm: Applied to Retail Banks in England and the Netherlands (1830-2007),* Promotor(s): Prof.dr. A. Jolink & Prof.dr. S.J. Magala,
EPS-2010-193-ORG, `http://hdl.handle.net/1765/19494`

Hernandez Mireles, C., *Marketing Modeling for New Products,* Promotor(s): Prof.dr. P.H. Franses,
EPS-2010-202-MKT, `http://hdl.handle.net/1765/19878`

Heyde Fernandes, D. von der, *The Functions and Dysfunctions of Reminders,* Promotor(s): Prof.dr. S.M.J. van Osselaer,
EPS-2013-295-MKT, `http://hdl.handle.net/1765/41514`

Heyden, M.L.M., *Essays on Upper Echelons & Strategic Renewal: A Multilevel Contingency Approach,* Promotor(s): Prof.dr. F.A.J. van den Bosch & Prof.dr. H.W. Volberda,
EPS-2012-259-STR, `http://hdl.handle.net/1765/32167`

Hoever, I.J., *Diversity and Creativity: In Search of Synergy,* Promotor(s): Prof.dr. D.L. van Knippenberg,
EPS-2012-267-ORG, `http://hdl.handle.net/1765/37392`

Hoogendoorn, B., *Social Entrepreneurship in the Modern Economy: Warm Glow, Cold Feet,* Promotor(s): Prof.dr. H.P.G. Pennings & Prof.dr. A.R. Thurik,
EPS-2011-246-STR, `http://hdl.handle.net/1765/26447`

Hoogervorst, N., *On The Psychology of Displaying Ethical Leadership: A Behavioral Ethics Approach,* Promotor(s): Prof.dr. D. De Cremer & Dr. M. van Dijke,
EPS-2011-244-ORG, `http://hdl.handle.net/1765/26228`

Huang, X., *An Analysis of Occupational Pension Provision: From Evaluation to Redesign,* Promotor(s): Prof.dr. M.J.C.M. Verbeek & Prof.dr. R.J. Mahieu,
EPS-2010-196-F&A, `http://hdl.handle.net/1765/19674`

Hytnen, K.A., *Context Effects in Valuation, Judgment and Choice,* Promotor(s): Prof.dr.ir. A. Smidts,
EPS-2011-252-MKT, `http://hdl.handle.net/1765/30668`

Jaarsveld, W.L. van, *Maintenance Centered Service Parts Inventory Control,* Promotor(s): Prof.dr.ir. R. Dekker,
EPS-2013-288-LIS, `http://hdl.handle.net/1765/39933`

Jalil, M.N., *Customer Information Driven After Sales Service Management: Lessons from Spare Parts Logistics,* Promotor(s): Prof.dr. L.G. Kroon,
EPS-2011-222-LIS, `http://hdl.handle.net/1765/22156`

Kagie, M., *Advances in Online Shopping Interfaces: Product Catalog Maps and Recommender Systems,* Promotor(s): Prof.dr. P.J.F. Groenen,
EPS-2010-195-MKT, `http://hdl.handle.net/1765/19532`

Kappe, E.R., *The Effectiveness of Pharmaceutical Marketing,* Promotor(s): Prof.dr. S. Stremersch,
EPS-2011-239-MKT, `http://hdl.handle.net/1765/23610`

Karreman, B., *Financial Services and Emerging Markets,* Promotor(s): Prof.dr. G.A. van der Knaap & Prof.dr. H.P.G. Pennings,
EPS-2011-223-ORG, `http://hdl.handle.net/1765/22280`

Kil, J.C.M., *Acquisitions Through a Behavioral and Real Options Lens,* Promotor(s): Prof.dr. H.T.J. Smit,
EPS-2013-298-F&A, `http://hdl.handle.net/1765/50142`

Klooster, E. vant, *Travel to Learn: The Influence of Cultural Distance on Competence Development in Educational Travel,* Promotors: Prof.dr. F.M. Go & Prof.dr. P.J. van Baalen,
EPS-2014-312-MKT, `http://hdl.handle.net/1765/151460`

Koendjbiharie, S.R., *The Information-Based View on Business Network Performance Revealing the Performance of Interorganizational Networks,* Promotors: Prof.dr.ir. H.W.G.M. van Heck & Prof.mr.dr. P.H.M. Vervest,
EPS-2014-315-LIS, `http://hdl.handle.net/1765/1`

Konter, D.J., *Crossing borders with HRM: An inquiry of the influence of contextual differences in the adaption and effectiveness of HRM,* Promotor(s): Prof.dr. J. Paauwe,
EPS-2014-305-ORG, `http://hdl.handle.net/1765/1`

Korkmaz, E., *Understanding Heterogeneity in Hidden Drivers of Customer Purchase Behavior,* Promotors: Prof.dr. S.L. van de Velde & dr. R.Kuik,
EPS-2014-316-LIS, `http://hdl.handle.net/1765/1`

Lam, K.Y., *Reliability and Rankings,* Promotor(s): Prof.dr. P.H.B.F. Franses,
EPS-2011-230-MKT, `http://hdl.handle.net/1765/22977`

Lander, M.W., *Profits or Professionalism? On Designing Professional Service Firms,* Promotor(s): Prof.dr. J. van Oosterhout & Prof.dr. P.P.M.A.R. Heugens,
EPS-2012-253-ORG, `http://hdl.handle.net/1765/30682`

Langhe, B. de, *Contingencies: Learning Numerical and Emotional Associations in an Uncertain World,* Promotor(s): Prof.dr.ir. B. Wierenga & Prof.dr. S.M.J. van Osselaer, EPS-2011-236-MKT, `http://hdl.handle.net/1765/23504`

Larco Martinelli, J.A., *Incorporating Worker-Specific Factors in Operations Management Models,* Promotor(s): Prof.dr.ir. J. Dul & Prof.dr. M.B.M. de Koster, EPS-2010-217-LIS, `http://hdl.handle.net/1765/21527`

Leunissen, J.M., *All Apologies: On the Willingness of Perpetrators to Apoligize,* Promotor(s): Prof.dr. D. De Cremer, EPS-2014-301-ORG, `http://hdl.handle.net/1765/1`

Liang, Q., *Governance, CEO Indentity, and Quality Provision of Farmer Cooperatives,* Promotor(s): Prof.dr. G.W.J. Hendrikse, EPS-2013-281-ORG, `http://hdl.handle.net/1765/1`

Liket, K.C., *Why Doing Good is not Good Enough: Essays on Social Impact Measurement,* Promotor(s): Prof.dr. H.R. Commandeur, EPS-2014-307-S&E, `http://hdl.handle.net/1765/51130`

Loos, M.J.H.M. van der, *Molecular Genetics and Hormones; New Frontiers in Entrepreneurship Research,* Promotor(s): Prof.dr. A.R. Thurik, Prof.dr. P.J.F. Groenen & Prof.dr. A. Hofman, EPS-2013-287-S&E, `http://hdl.handle.net/1765/40081`

Lovric, M., *Behavioral Finance and Agent-Based Artificial Markets,* Promotor(s): Prof.dr. J. Spronk & Prof.dr.ir. U. Kaymak, EPS-2011-229-F&A, `http://hdl.handle.net/1765/22814`

Lu, Y., *Data-Driven Decision Making in Auction Markets,* Promotors: Prof.dr.ir. H.W.G.M. van Heck & Prof.dr. W.Ketter, EPS-2014-314-LIS, `http://hdl.handle.net/1765/1`

Markwat, T.D., *Extreme Dependence in Asset Markets Around the Globe,* Promotor(s): Prof.dr. D.J.C. van Dijk, EPS-2011-227-F&A, `http://hdl.handle.net/1765/22744`

Mees, H., *Changing Fortunes: How Chinas Boom Caused the Financial Crisis,* Promotor(s): Prof.dr. Ph.H.B.F. Franses, EPS-2012-266-MKT, `http://hdl.handle.net/1765/34930`

Meuer, J., *Configurations of Inter-Firm Relations in Management Innovation: A Study in Chinas Biopharmaceutical Industry,* Promotor(s): Prof.dr. B. Krug, EPS-2011-228-ORG, `http://hdl.handle.net/1765/22745`

Mihalache, O.R., *Stimulating Firm Innovativeness: Probing the Interrelations between Managerial and Organizational Determinants,* Promotor(s): Prof.dr. J.J.P. Jansen, Prof.dr.ing. F.A.J. van den Bosch & Prof.dr. H.W. Volberda, EPS-2012-260-S&E, `http://hdl.handle.net/1765/32343`

Milea, V., *New Analytics for Financial Decision Support,* Promotor(s): Prof.dr.ir. U. Kaymak,
EPS-2013-275-LIS, `http://hdl.handle.net/1765/38673`

Naumovska, I., *Socially Situated Financial Markets:a Neo-Behavioral Perspective on Firms, Investors and Practices,* Promotor(s) Prof.dr. P.P.M.A.R. Heugens & Prof.dr. A.de Jong,
EPS-2014-319-S&E, `http://hdl.handle.net/1765/1`

Nielsen, L.K., *Rolling Stock Rescheduling in Passenger Railways: Applications in Short-term Planning and in Disruption Management,* Promotor(s): Prof.dr. L.G. Kroon,
EPS-2011-224-LIS, `http://hdl.handle.net/1765/22444`

Nijdam, M.H., *Leader Firms: The Value of Companies for the Competitiveness of the Rotterdam Seaport Cluster,* Promotor(s): Prof.dr. R.J.M. van Tulder,
EPS-2010-216-ORG, `http://hdl.handle.net/1765/21405`

Noordegraaf-Eelens, L.H.J., *Contested Communication: A Critical Analysis of Central Bank Speech,* Promotor(s): Prof.dr. Ph.H.B.F. Franses,
EPS-2010-209-MKT, `http://hdl.handle.net/1765/21061`

Nuijten, A.L.P., *Deaf Effect for Risk Warnings: A Causal Examination applied to Information Systems Projects,* Promotor(s): Prof.dr. G. van der Pijl & Prof.dr. H. Commandeur & Prof.dr. M. Keil,
EPS-2012-263-S&E, `http://hdl.handle.net/1765/34928`

Oosterhout, M., *van, Business Agility and Information Technology in Service Organizations,* Promotor(s): Prof,dr.ir. H.W.G.M. van Heck,
EPS-2010-198-LIS, `http://hdl.handle.net/1765/19805`

Osadchiy, S.E., *The Dynamics of Formal Organization: Essays on Bureaucracy and Formal Rules,* Promotor(s): Prof.dr. P.P.M.A.R. Heugens,
EPS-2011-231-ORG, `http://hdl.handle.net/1765/23250`

Otgaar, A.H.J., *Industrial Tourism: Where the Public Meets the Private,* Promotor(s): Prof.dr. L. van den Berg,
EPS-2010-219-ORG, `http://hdl.handle.net/1765/21585`

Ozdemir, M.N., *Project-level Governance, Monetary Incentives and Performance in Strategic R&D Alliances,* Promotor(s): Prof.dr.ir. J.C.M. van den Ende,
EPS-2011-235-LIS, `http://hdl.handle.net/1765/23550`

Peers, Y., *Econometric Advances in Diffusion Models,* Promotor(s): Prof.dr. Ph.H.B.F. Franses,
EPS-2011-251-MKT, `http://hdl.handle.net/1765/30586`

Pince, C., *Advances in Inventory Management: Dynamic Models,* Promotor(s): Prof.dr.ir. R. Dekker,
EPS-2010-199-LIS, `http://hdl.handle.net/1765/19867`

Porck, J.P., *No Team is an Island,* Promotor(s): Prof.dr. P.J.F. Groenen & Prof.dr. D.L. van Knippenberg,
EPS-2013-299-ORG, `http://hdl.handle.net/1765/50141`

Porras Prado, M., *The Long and Short Side of Real Estate, Real Estate Stocks, and Equity,* Promotor(s): Prof.dr. M.J.C.M. Verbeek,
EPS-2012-254-F&A, `http://hdl.handle.net/1765/30848`

Potthoff, D., *Railway Crew Rescheduling: Novel Approaches and Extensions,* Promotor(s): Prof.dr. A.P.M. Wagelmans & Prof.dr. L.G. Kroon,
EPS-2010-210-LIS, `http://hdl.handle.net/1765/21084`

Poruthiyil, P.V., *Steering Through: How Organizations Negotiate Permanent Uncertainty and Unresolvable Choices,* Promotor(s): Prof.dr. P.P.M.A.R. Heugens & Prof.dr. S. Magala,
EPS-2011-245-ORG, `http://hdl.handle.net/1765/26392`

Pourakbar, M., *End-of-Life Inventory Decisions of Service Parts,* Promotor(s): Prof.dr.ir. R. Dekker,
EPS-2011-249-LIS, `http://hdl.handle.net/1765/30584`

Pronker, E.S., *Innovation Paradox in Vaccine Target Selection,* Promotor(s): Prof.dr. H.R. Commandeur & Prof.dr. H.J.H.M. Claassen,
EPS-2013-282-S&E, `http://hdl.handle.net/1765/39654`

Retel Helmrich, M.J., *Green Lot-Sizing,* Promotor(s): Prof.dr. A.P.M. Wagelmans,
EPS-2013-291-LIS, `http://hdl.handle.net/1765/41330`

Rietveld, C.A., *Essays on the Intersection of Economics and Biology,* Promotor(s): Prof.dr. P.J.F. Groenen, Prof.dr. A. Hofman & Prof.dr. A.R. Thurik,
EPS-2014-320-S&E, `http://hdl.handle.net/1765/1`

Rijsenbilt, J.A., *CEO Narcissism; Measurement and Impact,* Promotor(s): Prof.dr. A.G.Z. Kemna & Prof.dr. H.R. Commandeur,
EPS-2011-238-STR, `http://hdl.handle.net/1765/23554`

Roelofsen, E.M., *The Role of Analyst Conference Calls in Capital Markets,* Promotor(s): Prof.dr. G.M.H. Mertens & Prof.dr. L.G. van der Tas RA,
EPS-2010-190-F&A, `http://hdl.handle.net/1765/18013`

Roza, M.W., *The Relationship between Offshoring Strategies and Firm Performance: Impact of Innovation, Absorptive Capacity and Firm Size,* Promotor(s): Prof.dr. H.W. Volberda & Prof.dr.ing. F.A.J. van den Bosch,
EPS-2011-214-STR, `http://hdl.handle.net/1765/22155`

Rubbaniy, G., *Investment Behavior of Institutional Investors,* Promotor(s): Prof.dr. W.F.C. Verschoor,
EPS-2013-284-F&A, `http://hdl.handle.net/1765/40068`

Schellekens, G.A.C., *Language Abstraction in Word of Mouth,* Promotor(s): Prof.dr.ir. A. Smidts,
EPS-2010-218-MKT, `http://hdl.handle.net/1765/21580`

Shahzad, K., *Credit Rating Agencies, Financial Regulations and the Capital Markets,* Promotor(s): Prof.dr. G.M.H. Mertens,
EPS-2013-283-F&A, `http://hdl.handle.net/1765/39655`

Sotgiu, F., *Not All Promotions are Made Equal: From the Effects of a Price War to Cross-chain Cannibalization,* Promotor(s): Prof.dr. M.G. Dekimpe & Prof.dr.ir. B. Wierenga,
EPS-2010-203-MKT, `http://hdl.handle.net/1765/19714`

Sousa, M., *Servant Leadership to the Test: New Perspectives and Insight,* Promotors: Prof.dr. D. van Knippenberg & Dr. D. van Dierendonck,
EPS-2014-313-ORG, `http://hdl.net/1765/1`

Spliet, R., *Vehicle Routing with Uncertain Demand,* Promotor(s): Prof.dr.ir. R. Dekker,
EPS-2013-293-LIS, `http://hdl.handle.net/1765/41513`

Srour, F.J., *Dissecting Drayage: An Examination of Structure, Information, and Control in Drayage Operations,* Promotor(s): Prof.dr. S.L. van de Velde,
EPS-2010-186-LIS, `http://hdl.handle.net/1765/18231`

Staadt, J.L., *Leading Public Housing Organisation in a Problematic Situation: A Critical Soft Systems Methodology Approach,* Promotor(s): Prof.dr. S.J. Magala,
EPS-2014-308-ORG, `http://hdl.handle.net/1765/50712`

Stallen, M., *Social Context Effects on Decision-Making; A Neurobiological Approach,* Promotor(s): Prof.dr.ir. A. Smidts,
EPS-2013-285-MKT, `http://hdl.handle.net/1765/39931`

Tarakci, M., *Behavioral Strategy; Strategic Consensus, Power and Networks,* Promotor(s): Prof.dr. P.J.F. Groenen & Prof.dr. D.L. van Knippenberg,
EPS-2013-280-ORG, `http://hdl.handle.net/1765/39130`

Teixeira de Vasconcelos, M., *Agency Costs, Firm Value, and Corporate Investment,* Promotor(s): Prof.dr. P.G.J. Roosenboom,
EPS-2012-265-F&A, `http://hdl.handle.net/1765/37265`

Tempelaar, M.P., *Organizing for Ambidexterity: Studies on the Pursuit of Exploration and Exploitation through Differentiation, Integration, Contextual and Individual Attributes,* Promotor(s): Prof.dr.ing. F.A.J. van den Bosch & Prof.dr. H.W. Volberda,
EPS-2010-191-STR, `http://hdl.handle.net/1765/18457`

Tiwari, V., *Transition Process and Performance in IT Outsourcing: Evidence from a Field Study and Laboratory Experiments,* Promotor(s): Prof.dr.ir. H.W.G.M. van Heck & Prof.dr. P.H.M. Vervest,
EPS-2010-201-LIS, `http://hdl.handle.net/1765/19868`

Trster, C., *Nationality Heterogeneity and Interpersonal Relationships at Work,* Promotor(s): Prof.dr. D.L. van Knippenberg,
EPS-2011-233-ORG, `http://hdl.handle.net/1765/23298`

Tsekouras, D., *No Pain No Gain: The Beneficial Role of Consumer Effort in Decision Making,* Promotor(s): Prof.dr.ir. B.G.C. Dellaert,
EPS-2012-268-MKT, `http://hdl.handle.net/1765/37542`

Tzioti, S., *Let Me Give You a Piece of Advice: Empirical Papers about Advice Taking in Marketing,* Promotor(s): Prof.dr. S.M.J. van Osselaer & Prof.dr.ir. B. Wierenga,
EPS-2010-211-MKT, `http://hdl.handle.net/1765/21149`

Vaccaro, I.G., *Management Innovation: Studies on the Role of Internal Change Agents,* Promotor(s): Prof.dr. F.A.J. van den Bosch & Prof.dr. H.W. Volberda,
EPS-2010-212-STR, `http://hdl.handle.net/1765/21150`

Vagias, D., *Liquidity, Investors and International Capital Markets,* Promotor(s): Prof.dr. M.A. van Dijk,
EPS-2013-294-F&A, `http://hdl.handle.net/1765/41511`

Verheijen, H.J.J., *Vendor-Buyer Coordination in Supply Chains,* Promotor(s): Prof.dr.ir. J.A.E.E. van Nunen,
EPS-2010-194-LIS, `http://hdl.handle.net/1765/19594`

Venus, M., *Demystifying Visionary Leadership; In Search of the Essence of Effective Vision Communication,* Promotor(s): Prof.dr. D.L. van Knippenberg,
EPS-2013-289-ORG, `http://hdl.handle.net/1765/40079`

Visser, V., *Leader Affect and Leader Effectiveness; How Leader Affective Displays Influence Follower Outcomes,* Promotor(s): Prof.dr. D. van Knippenberg,
EPS-2013-286-ORG, `http://hdl.handle.net/1765/40076`

Vlam, A.J., *Customer First? The Relationship between Advisors and Consumers of Financial Products,* Promotor(s): Prof.dr. Ph.H.B.F. Franses,
EPS-2011-250-MKT, `http://hdl.handle.net/1765/30585`

Waard, E.J. de, *Engaging Environmental Turbulence: Organizational Determinants for Repetitive Quick and Adequate Responses,* Promotor(s): Prof.dr. H.W. Volberda & Prof.dr. J. Soeters,
EPS-2010-189-STR, `http://hdl.handle.net/1765/18012`

Waltman, L., *Computational and Game-Theoretic Approaches for Modeling Bounded Rationality,* Promotor(s): Prof.dr.ir. R. Dekker & Prof.dr.ir. U. Kaymak,
EPS-2011-248-LIS, `http://hdl.handle.net/1765/26564`

Wang, Y., *Information Content of Mutual Fund Portfolio Disclosure,* Promotor(s): Prof.dr. M.J.C.M. Verbeek,
EPS-2011-242-F&A, `http://hdl.handle.net/1765/26066`

Wang, Y., *Corporate Reputation Management; Reaching Out to Find Stakeholders,* Promotor(s): Prof.dr. C.B.M. van Riel,
EPS-2013-271-ORG, `http://hdl.handle.net/1765/38675`

Weenen, T.C., *On the Origin and Development of the Medical Nutrition Industry,* Promotors: Prof.dr. H.R. Commandeur & Prof.dr. H.J.H.M. Claassen,
EPS-2014-309-S&E, `http://hdl.handle.net/1765/51134`

Wolfswinkel, M., *Corporate Governance, Firm Risk and Shareholder Value of Dutch Firms,* Promotor(s): Prof.dr. A. de Jong,
EPS-2013-277-F&A, `http://hdl.handle.net/1765/39127`

Xu, Y., *Empirical Essays on the Stock Returns, Risk Management, and Liquidity Creation of Banks,* Promotor(s): Prof.dr. M.J.C.M. Verbeek,
EPS-2010-188-F&A, `http://hdl.handle.net/1765/18125`

Zaerpour, N., *Efficient Management of Compact Storage Systems,* Promotor(s): Prof.dr. M.B.M. de Koster,
EPS-2013-276-LIS, `http://hdl.handle.net/1765/38766`

Zhang, D., *Essays in Executive Compensation,* Promotor(s): Prof.dr. I. Dittmann,
EPS-2012-261-F&A, `http://hdl.handle.net/1765/32344`

Zhang, X., *Scheduling with Time Lags,* Promotor(s): Prof.dr. S.L. van de Velde,
EPS-2010-206-LIS, `http://hdl.handle.net/1765/19928`

Zhou, H., *Knowledge, Entrepreneurship and Performance: Evidence from Country-level and Firm-level Studies,* Promotor(s): Prof.dr. A.R. Thurik & Prof.dr. L.M. Uhlaner,
EPS-2010-207-ORG, `http://hdl.handle.net/1765/20634`

Zwan, P.W. van der, *The Entrepreneurial Process: An International Analysis of Entry and Exit,* Promotor(s): Prof.dr. A.R. Thurik & Prof.dr. P.J.F. Groenen,
EPS-2011-234-ORG, `http://hdl.handle.net/1765/23422`

CREW MANAGEMENT IN PASSENGER RAIL TRANSPORT

Crew management in passenger rail transport is an important factor that contributes to both the quality of service to the railway passengers and to the operational costs of the train operating company. This thesis describes how the (railway) Crew Management process can be improved with the introduction of advanced decision support systems, based on advanced mathematical models and algorithms. We provide a managerial perspective on the change process, related to the introduction of these systems, and give an overview of the lessons learned.

We have shown that introducing decision support can give substantial improvements in the overall performance of a railway company. Within NS, the support for the Crew Management process has led to a stable relationship between management and train crew. In addition, the lead-time of the planning process is shortened from months to hours and NS is now able to perform scenario analyses, e.g., to study effects of adjusting the labour rules.

Also, NS can adjust their service when severe weather conditions are expected, by creating a specific winter timetable shortly before the day of operation. Finally, we also introduced a decision support system for real-time rescheduling of crew duties on the day of operations. This enables us to adapt the actual crew schedules very quickly. As a result, we reduce the number of cancelled trains and fewer trains will be delayed in case of unforeseen disruptions.

E R i M

The Erasmus Research Institute of Management (ERIM) is the Research School (Onderzoekschool) in the field of management of the Erasmus University Rotterdam. The founding participants of ERIM are the Rotterdam School of Management (RSM), and the Erasmus School of Economics (ESE). ERIM was founded in 1999 and is officially accredited by the Royal Netherlands Academy of Arts and Sciences (KNAW). The research undertaken by ERIM is focused on the management of the firm in its environment, its intra- and interfirm relations, and its business processes in their interdependent connections.

The objective of ERIM is to carry out first rate research in management, and to offer an advanced doctoral programme in Research in Management. Within ERIM, over three hundred senior researchers and PhD candidates are active in the different research programmes. From a variety of academic backgrounds and expertises, the ERIM community is united in striving for excellence and working at the forefront of creating new business knowledge.

ERIM PhD Series
Research in Management

Erasmus Research Institute of Management - E R i M	Tel. +31 10 408 11 82
Rotterdam School of Management (RSM)	Fax +31 10 408 96 40
Erasmus School of Economics (ESE)	E-mail info@erim.eur.nl
Erasmus University Rotterdam (EUR)	Internet www.erim.eur.nl
P.O. Box 1738, 3000 DR Rotterdam,	
The Netherlands	

Erasmus Research Institute of Management - ERIM