Rolling stock planning

adapting the Composition Model to the operational planning phase

Mieras, W.

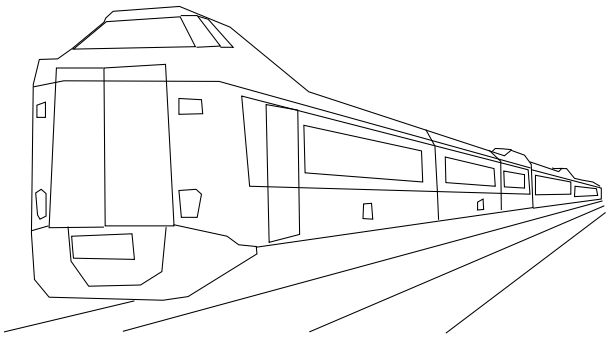
Award date:
2008

TECHNISCHE UNIVERSITEIT EINDHOVEN
Department of Mathematics and Computer Science

Rolling Stock Planning:

Adapting the Composition Model to the Operational Planning Phase

by Wouter Mieras



Supervisors:

Bert Gerards (TU/e)
Leo Kroon (NS)

Oktober 2007

Abstract

This report considers the rolling stock planning problem – the problem of assigning rolling stock units to trains in some feasible and optimal way. The shunting movements introduced by this assignment should be feasible and several other requirements should be met. Different criteria exist for optimality: for example the passenger demand should be satisfied, the number of carriage kilometers should be minimized and the number of shunting movements should be as low as possible.

A model is described in this report that deals with this problem for the tactical planning phase, the Composition Model. The tactical planning phase takes place at least several months before the execution date of the plans and here a plan is made 'from scratch'. This report focuses on modifying the composition model to be able to deal with the operational planning phase, which takes place several months to several days before the execution time of the plans. Here an original plan must be modified without making too many changes, and other criteria become more important.

Several additions are introduced to the composition model. Exceptional shunting movements are accepted in the model, which allows it to use an original plan as a feasible input. More details of the shunting process are taken into account, including a more detailed description of the inventory at stations and a special type of fast shunting movements. Implementation issues are considered besides theoretical considerations. Furthermore a heuristic is introduced which can reduce the solution time significantly, which is especially important in the operational planning phase.

Preface

From February 2007 till September 2007 I did an internship at NS, the main Dutch railway operator. Here I worked on the Final Project of my Master's degree program Industrial and Applied Mathematics (IAM), with specialization Discrete Mathematics and Applications (DMA), of the Technical University of Eindhoven (TU/e). I was introduced to a model that determines a rolling stock plan, the Composition Model, which describes the rolling stock planning problem as an integer programming problem. My assignment was to investigate how to modify this model to make it capable of planning closer to the execution time of the plans, where a previously created plan should be taken into account.

During my first few months I've studied the model and its implementation, and developed some intuition for it. As a first addition, I implemented an addition that makes the model capable of accepting exceptional shunting movements. By analyzing an earlier created rolling stock plan and some discussions with my supervisors, I developed a few ideas for extensions to the model, which focused on taking into account more details about shunting movements. I implemented these ideas and described them in a more theoretical context in this report. During the final stage of my internship I've worked on solution methods to reduce the solution time for the model, and implemented a heuristic that can significantly reduce this solution time.

I would like to thank my supervisors Bert Gerards, Leo Kroon and Gábor Maróti. Furthermore I would like to thank Cor Hurkens for some valuable advice and my colleagues at NS for creating a nice working environment.

I hope the reader will enjoy reading this report.

Wouter Mieras

Contents

Chapter 1

Introduction

The most important Dutch railway operator NS (Nederlandse Spoorwegen) is responsible for the transport of over a million passengers per day. A lot of problems need to be dealt with. To mention a few, train drives, conductors and engineers need to be assigned to tasks in order to keep the trains rolling, schedules have to be made for train lines and for rolling stock, and decisions have to be made about building new tracks and buying new rolling stock.

In this report the rolling stock planning problem is addressed: the problem of assigning rolling stock to trains. A model is introduced that was designed to deal with this problem and that has been used by NS. This report presents some additions to this model, with the purpose of making the model more widely applicable, especially to make it capable of modifying plans shortly before the execution time of the plans. Also some solution methods to improve the running time of the model are discussed.

In section 1.1 a short overview will be given of the general planning process at NS, and section 1.2 describes some problems specific to NS and the Dutch railway system. Finally, an outline of the rest of the report is given in section 1.3.

1.1 The Planning Process

In order to keep the trains rolling a lot of planning needs to be done. This section gives a short overview of the planning process and what type of problems need to be dealt with. One can divide the planning process for keeping the train services operational into four distinct time phases in which specific decisions have to be made:

- **Strategic planning**: Here long term decisions are made, usually a year or even a decade in advance. New train lines are planned and old train lines are altered or cancelled. Strategic decisions are made for example about hiring and training new crew members, buying new rolling stock and refurbishing old rolling stock.

- **Tactical planning**: In this phase a timetable is made for the train lines. Generic hours, days and weeks are specified. Also a first plan is made that assigns rolling stock to train lines. For every train the composition of rolling stock is determined and also which composition changes take place at stations, the **shunting movements**. This gives anonymous **duties** for the rolling stock units, in a later phase these anonymous duties are assigned to real rolling stock units. Also generic crew rosters are created. Tactical planning usually takes place at least several months before the plan is executed.
- **Operational planning**: Several months till a few days in advance the plans are detailed and altered. Generic schedules are converted to specific schedules which take into account for example festivals, for which more and longer trains are needed, and maintenance work on tracks. If changes are made in the planning of shunting movements at stations they need to be checked by local planners to make sure they are feasible, so the communication between central and local planners needs to work smoothly. Generally, in this phase it is more important that the plans are feasible than that they are optimal as there is not enough time to change much.
- **Short-term planning**: This phase includes planning from a few days ahead till and including the real-time execution of the plans. The anonymous rolling stock duties are now assigned to real rolling stock units and similarly duties are assigned to crew members. When delays or disruptions occur during the execution of the plans quick solutions need to be found: trains and crew need to be rerouted in order to minimize the number of passengers that are delayed, and to deal with other problems like getting crew members home and minimizing the extra kilometers that need to be made by rolling stock units. Also the maintenance of rolling stock units needs to be managed: specific types of rolling stock can only be checked or repaired in specific stations, so they need to be routed to these locations.

Summarizing the planning process above, three major problems need to be dealt with: creating a timetable for train lines, making a rolling stock schedule and making a crew schedule. A lot of changes are made during the planning process, the original plans and the final execution of the plans are usually quite different. Also note that many people are involved in the planning process who need to work together, the central planners have a better overview of global problems but they need the local planners to check if their plans are feasible locally.

1.2 Further Characteristics

This section describes some characteristics of the Dutch railway system which pose some unique challenges.

The **traffic density** of trains on the railway tracks is very high in the Netherlands. This means that trains cannot stay too long at stations since they must make place for new trains. Since shunting movements at stations can take a lot of time and cause delays it is preferable to minimize the amount of shunting movements. To some train lines so many trains are assigned

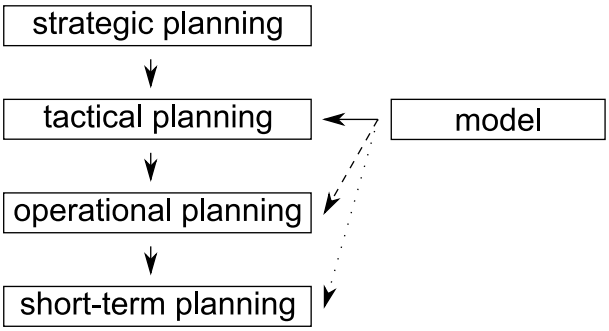


Figure 1.1: The role of the model in the planning process for rolling stock. This report introduces some additions to the model to make it more useful for operational planning. Eventually it would be nice if the model could also be used for real-time planning.

that 'queues' of trains emerge: due to the safety requirement that the railway track should be free for at least a few minutes before the next train can pass, trains have to wait for each other. This means that if one train is delayed, a lot of successive trains could be delayed as well.

Another characteristic of the railway system in the Netherlands is that most passenger trains consist of **train units** that can be operated independently instead of locomotive hauled carriages. There are different types of units of which some can be combined in one train and others cannot. On some train lines trains are split in two parts or two trains are combined in a single new train. Using units instead of locomotive hauled carriages adds a lot of flexibility to the railway system. For example, the train length can more easily be adapted to the passenger demand. But since one needs to keep track of the units and the order of the units in the train, the planning process becomes more difficult.

In short, there are two important characteristics of the Dutch railway system that create challenges in the planning process: the high traffic density of trains on the railway tracks and the usage of units.

1.3 Overview of this Report

In chapter 2 a more detailed description is given of the tactical rolling stock problem. A way to model this problem, the **composition model**, is introduced and some implementation issues are discussed. Chapter 3 describes some additions to the composition model which primarily aim to make the model more suitable for planning closer to the execution time of the plans. Some techniques to improve the running time of the model are discussed in chapter 4. Finally chapter 5 discusses the usefulness of the additions and gives suggestions for further research.

Figure 1.1 shows the position of the model relative to the four time phases in the planning process for rolling stock. The model described in chapter 2 is primarily designed for the

tactical planning phase, this report aims to make it more useful for the operational planning phase. While in the tactical planning phase a schedule for rolling stock is made 'from scratch', in the operational planning phase a previously created plan exists and needs to be modified. This report describes how the model can be extended to be able to do this, and adds some extra components to the model that are relevant in the operational planning phase. These components focus mainly on shunting: how including some details about shunting can help to create better plans. As time is a more critical factor in the operational planning phase also some attention is given to improving the solution time.

Some additions described in this report could also be used in the tactical planning phase. Future research could be aimed at making the model also useful for the short term planning phase, including modifying rolling stock plans in real-time.

Chapter 2

Tactical Rolling Stock Planning

In this chapter one of the major problems in the planning process of NS is considered – the **tactical rolling stock planning**. Based on previously created timetables, one needs to determine duties for train units. This used to be done by hand, but the **composition model** described in this chapter can automatically generate a rolling stock plan. A major advantage of automatically generating a rolling stock plan is that it can take into account entire train lines at once, including many details and objectives. A disadvantage is that detailed knowledge of planners of for example the local situation at train stations cannot easily be incorporated in the model.

A lot of research has been done on this and similar problems. The composition model turns out to work well for the situation in the Netherlands and is able to provide fast and reasonable solutions, often with better objective values than obtained from planning by hand.

Section 2.1 describes the rolling stock problem in more detail. It considers when a rolling stock plan is feasible and also the most important criteria for evaluating a solution. In section 2.2 the composition model is described, and some additions to make the model able to handle more realistic scenarios and to be able to obtain solutions faster are described in section 2.3. Finally, section 2.4 discusses the practical implementation of the composition model. The majority of this chapter (mainly sections 2.1 - 2.3) was based on [1]. There more background information and another way to model the tactical rolling stock problem can be found.

The composition model described in this chapter is designed for planning 'from scratch'. In chapter 3 the model is extended to be able to modify a previously created rolling stock schedule, thereby taking into account the existing plans as much as possible.

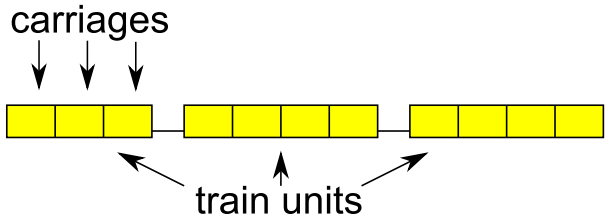


Figure 2.1: An example of the composition of a train. A train consists of units which in turn contain a number of carriages. Every unit has its own engine and two driver's cabins which allow it to move independently.

2.1 Problem Description

This section gives a more detailed description of the tactical rolling stock planning. First the basic problem is described in section 2.1.1. Then some requirements for the feasibility of a plan are given in section 2.1.2, which include restrictions for the shunting possibilities at stations and some inventory constraints. Section 2.1.3 describes several important objective criteria needed to judge a particular solution of the tactical rolling stock problem. These include the amount of carriage kilometers and passenger satisfaction. Finally, section 2.1.4 summarizes the most important issues in the tactical rolling stock planning problem.

2.1.1 Basic Problem

The timetable consists of **train lines** that describe the movements of **trains** from a certain starting station to an end station. For example, train 1648 describes the train that departs from Enschede station at 13:27 and arrives at Schiphol station at 15:41. Trains consist of **units**, which can be seen as short trains that have their own engines and can be operated independently. These units consist in turn of **carriages**, and units of a similar type can be combined in one train. See figure 2.1 for an example of the composition of a train in units and carriages. Train lines usually use only a particular type of units or two different units of a similar type that can be combined. So one can focus on one type of unit or two types of similar units at a time corresponding to particular train lines, and solve the rolling stock problem separately for these types, under the assumption that the plans for different types of units do not interfere.

The timetable for the train lines can be decomposed into **trips** – sequences of train movements where no composition changes can take place. For example, for train 1648 composition changes can take place in the intermediate stations Deventer and Amersfoort, which gives three trips for this train: Enschede - Deventer, Deventer - Amersfoort and Amersfoort - Schiphol, see figure 2.2.

A **duty** for a rolling stock unit consists of a sequence of trips the unit serves in successively, including the position of the unit in the train. Now the rolling stock problem can be described as follows:

Create duties for train units in such a way that the resulting schedule is feasible and optimal in some predetermined way.

Note that the duties for rolling stock units determine the compositions of the trips. Vice versa, if the compositions of the trips are known and some additional feasibility requirements are met, duties for rolling stock units can be determined. In the latter case there are usually several possible solutions for the duties. The composition model introduced in section 2.2 will determine compositions for trips and uses this to create the duties.

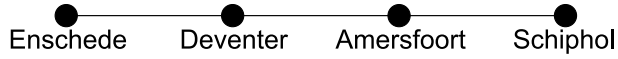


Figure 2.2: Train 1648. The train departs from Enschede and arrives at Schiphol. Composition changes can only take place in Deventer and Amersfoort which gives a division of the route into three trips.

To restrict the problem a bit, only one day at a time is considered in this report. During the night there are not much train services scheduled. This makes it possible, though not desirable, to move units and make other changes in order to start the new day in a desired configuration. Furthermore, one can connect successive days by setting restrictions on the initial and final inventory of units at the stations.

The main focus in this report is on train lines where two similar types of units are used that can be combined instead of train lines where only one type of unit is used. An advantage of using two similar types of units is that one has more possibilities to control the length of the train and optimize on that. It is much harder to solve the problem with two different types of units than when only one type of unit is used, since there are many more possible compositions of units in trains possible when two different units are used – this also makes it a more interesting problem to consider. An example of a train unit with two variations that can be combined is the **koploper** train unit. It can consist of 3 carriages or 4 carriages and it is the main example in this report as it is the most complex. Because a koploper train can contain units of either 3 or 4 carriages, its length can be 3, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14 or 15 carriages. See figure 2.3 for a picture of a train consisting of koploper units.

2.1.2 Feasibility

Important questions are when a rolling stock plan is **feasible** and when a feasible plan is **optimal**. For feasibility there are many possible restrictions. **Shunting** movements at the stations are perhaps the most important issue to consider. Since there is a limited time available for shunting, usually only minor changes can be made between two successive trips. For example, uncoupling one train unit from the rear of the train or coupling one train unit to the front of the train might be possible, but doing both at the same stop takes too much time and requires a lot of manpower. See figures 2.4, 2.5 and 2.6 for some examples of shunting movements. After a train unit is uncoupled it is usually stored at a local shunting yard where it stays until it is needed again. Storing a train unit at the shunting yard and retrieving a train unit from the shunting yard takes some time, in this report usually a reallocation time of

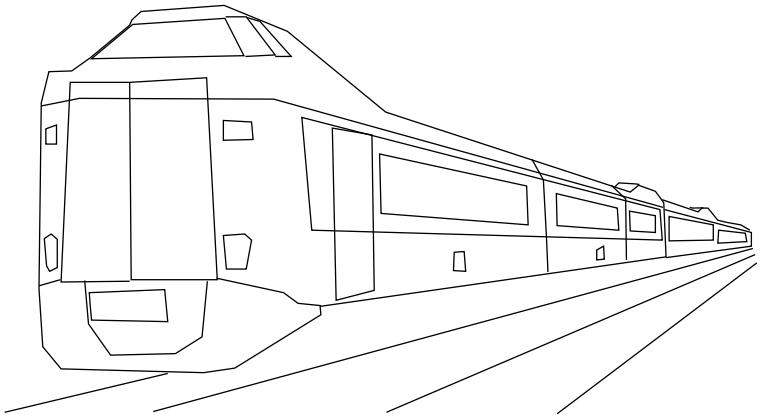


Figure 2.3: A train with koploper units. It is mainly used on the 'Noord-Oost' line group, a group of intercity lines that connect several big cities in the west of the Netherlands (Amsterdam, Rotterdam, The Hague) to cities in the northeast (Groningen, Leeuwarden, Enschede). Notice the characteristic front of the train, it used to be possible to combine two such heads in order to allow passengers and staff to switch between two koploper units, but they are now welded shut. The koploper units are the main example in this report.

30 minutes is assumed. There are general directives to determine which shunting movements are possible, but it also depends on the local situation. For example, there is no shunting yard in Deventer, so all uncoupled units must remain at the station and are usually redeployed very quickly. In section 2.4 the most common shunting movements are described and given standard shunting codes.

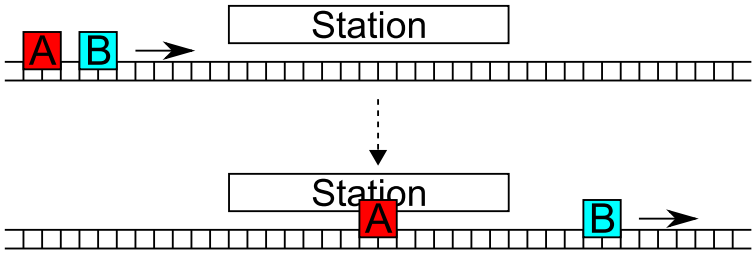


Figure 2.4: Uncoupling: a train consisting of units A and B arrives at the station and unit A is uncoupled.

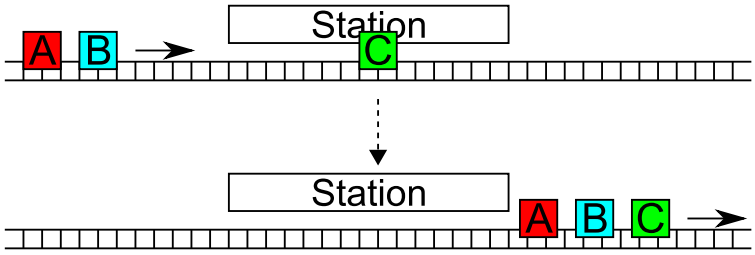


Figure 2.5: Coupling: a train consisting of units A and B arrives at the station and unit C is coupled to the train.

A lot of things can happen at the **shunting yard**: train units get cleaned and undergo maintenance checks for example. The details of what happens at the shunting yard are not

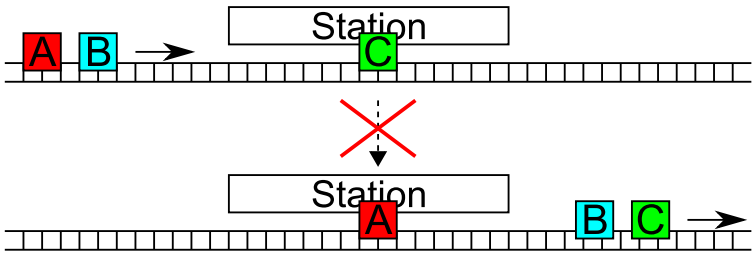


Figure 2.6: Coupling and Uncoupling: a train consisting of units A and B arrives at the station. Unit A is uncoupled from the train and unit C is coupled to the train. In theory this is a valid shunting movement, but in practice this type of shunting movement is avoided since it takes a lot of time and manpower.

taken into account in the model although they certainly have influence, they determine for example which train unit at the shunting yard can most easily be redeployed. In chapter 3 an attempt is made to include some more details of what happens at the shunting yard. But since many details of the precise shunting process are not available, especially when planning months ahead, feedback from local planners remains necessary after constructing a rolling stock plan.

Another issue to consider is the **inventory** at stations. There is a limited number of units available. Sometimes units become unavailable due to breakdowns or maintenance, which makes it difficult to predict the exact number of units available during the execution time of the plans, especially when planning a long time ahead. This is one of the reasons why it is desirable to make the model suitable for planning closer to the execution time of the plans. Another issue to consider is the maximal capacity of the shunting yard at stations, some stations can only store only a few or even no units.

There are many more requirements for a plan to be feasible. The maximal train length cannot be longer than the length of the platforms at a station. One needs to take into account combining and splitting of trains: sometimes a trip has two successor trips which both take some of the units of the trip, or the units of two trips are combined in one successor trip. The **continuity requirement** states that for all trains there should be at least one unit that is in all trips of that train. A simple version of the latter requirement is implemented in the composition model described in this chapter, the next chapter presents a more elaborate way to model this.

It is difficult if not impossible to make an entirely feasible plan at once, both automatically and also when constructing a plan manually. Feedback from local planners is needed to make sure that a plan can be executed, and even then a lot of changes will have to be made to accommodate changes on shorter term. The model described in this chapter implements some general directives which aim to make a good approximation of reality, thus minimizing the number of changes needed in order to make a plan feasible in practice instead of only in theory.

2.1.3 Objective

Another important question is which feasible solution is the best, or at least to have some measure to determine how 'good' a feasible solution is. In the basic model three main criteria are used to determine whether solutions are good or not:

- Firstly, it is important that the trains are long enough to carry all passengers that are expected. To determine the expected number of passengers, surveys are conducted and observations from conductors are used. Since determining the expected number of passengers is not an exact science, the numbers can be unreliable, which makes it questionable to claim that a certain rolling stock plan is the 'best' plan in practice. This is an important thing to keep in mind when trying to find an optimal solution: since the basic input is somewhat unreliable it makes not much sense to spend a lot of time on finding the absolutely best solution.
- The number of carriage kilometers, the total number of kilometers ridden by all carriages, is preferably as low as possible. Driving train units around is expensive due to energy costs and maintenance, so if one can use a short train instead of a longer train, the short train is preferable.
- Shunting movements are a third criterion. Shunting movements take time and require crew. They may also cause disruptions. For example, sometimes coupling two train units can fail. Another problem is that shunting movements tax the capacity of the infrastructure as usually multiple railway tracks are required to carry them out, making them temporarily unavailable for other train movements. Considering the above, one would like to minimize the number of shunting movements.

When planning closer to the execution time of the rolling stock schedule and using a previously created plan, as this report tries to describe, other criteria might become more important. Changing shunting movements, for example uncoupling a train unit one station later than in the original plan, might be very expensive since the staff members need to be reallocated. And there just might not be enough time to change the local plan, since communicating the plan back and forth and asking for approval from the railway operator takes a lot of time. Often it is more important to come up with a plan that is feasible than with a plan that is optimal.

2.1.4 Summary

A short summary of what was described above is given in this section.

One needs to create duties for train units, where a duty consists of a sequence of successive trips the unit participates in, including the position of the unit in the train. This also determines the compositions of the trips. In order for this assignment to be feasible one needs among other things:

- The composition changes, or shunting movements, between two trips need to be feasible;
- There are only a limited number of units available and stations only have a limited storage capacity;
- The continuity requirement must be satisfied;
- A plan must be locally feasible – the shunting movements at the stations must be executable, which depends on factors like availability of crew and planned maintenance of train units. This requires feedback from local planners.

There are several criteria to judge the quality of a feasible plan:

- The passenger demand must be satisfied;
- Carriage kilometers should be minimized;
- Shunting movements are expensive;
- When planning closer to the execution date of the schedule and using a previously created plan other criteria might become more important.

2.2 The Composition Model

In this section a mixed integer programming problem formulation is given that models the tactical rolling stock problem described in the previous section, this is called the **composition model**.

One main observation used here is that it is only necessary to determine the **compositions** of the trips and that the duties for the train units can always be determined from this, if one makes sure that the inventories at stations never become negative and that the shunting movements are feasible. Therefore, the composition model focuses on determining the compositions of trips and determining the actual duties for the rolling stock schedule is part of post processing the output of the model.

2.2.1 Notation

First some preliminary notation. Let $\mathcal{M}$ denote the set of rolling stock types used in the lines considered, so the interesting problems have $|\mathcal{M}| \geq 2$. For all $m \in \mathcal{M}$ let n_m denote the number of units of type m that are available and let c_m denote the number of carriages in units of type m. Denote the set of service classes by $\mathcal{C}$: there are two service classes namely the first class and the second class. A unit of type m has $k_{m,c}$ seats of class c.

The set of stations is denoted by $\mathcal{S}$. The values $i^0_{s,m}$ and $i^\infty_{s,m}$ denote the preferred initial and final inventory of type m at station s respectively. Let $\mathcal{T}$ denote the set of trips. Every trip has a departure station $s_d(t)$, an arrival station $s_a(t)$, a departure time $\tau_d(t)$ and an arrival time $\tau_a(t)$. The variable d_t gives the length of the trip in kilometers and the passenger demand of the trip is $\delta_{t,c}$ for passenger class c. If a trip is the follow-up trip of trip t, it is called the **successor trip** of trip t and is denoted by $\sigma(t)$. If a trip is split such that it has two successor trips, they are denoted by $\sigma^1(t)$ and $\sigma^2(t)$. The time before the units uncoupled from trip t can be reused, the reallocation time, is denoted by $\rho(t)$. In this report $\rho(t)$ is considered to be 30 minutes for all trips. Let $\mathcal{T}_0$ denote the set of trips with no predecessor trips and let $\mathcal{T}_\infty$ denote the set of trips with no successor trips. Trips from $\mathcal{T}_0$ are called **Starters** and trips from $\mathcal{T}_\infty$ are called **Finishers**.

An ordered sequence of units from $\mathcal{M}$ is called a **composition**. If p is a composition, $|p|$ denotes the number of units in it and $\nu(p)_m$ denotes the number of units of type m in this composition. Let $\nu(p) \in \mathbb{Z}^{\mathcal{M}}$ be a vector of values $\nu(p)_m$ describing the number of units in composition p for every $m \in \mathcal{M}$. Let $\mathcal{P}_t$ denote the set of compositions that are allowed for trip t. This set depends for example on the maximum length of a train allowed on trip t. Possible composition changes are described by $\mathcal{G}_t$, which is the set of pairs of compositions (p, p') where $p \in \mathcal{P}_t$ and $p' \in \mathcal{P}_{\sigma(t)}$ such that the composition change, called **transition**, between p and p' is allowed. This is determined by the shunting possibilities between trip t and its successor trip $\sigma(t)$. Let $c_m(p, p')$ denote the number of units of type m that are coupled to a train during the transition between composition p and p', and let $u_m(p, p')$ denote the number of units of type m that are uncoupled from the train during the transition between composition p and p'. Note that in most cases it holds that $c_m(p, p') = \max(0, \nu(p')_m - \nu(p)_m)$ and that $u_m(p, p') = \max(0, \nu(p)_m - \nu(p')_m)$. This is actually an assumption in the original implementation of the composition model.

The main problem is to assign compositions of units to trips, this is modelled with the decision variables $X_{t,p} \in \{0, 1\}$ which indicate whether composition p is used for trip t ($X_{t,p} = 1$) or not ($X_{t,p} = 0$). Transitions between trips are modelled with decision variables $Z_{t,p,p'} \in \{0, 1\}$ which indicate whether trip t has composition p and its successor trip $\sigma(t)$ has composition p' ($Z_{t,p,p'} = 1$) or not ($Z_{t,p,p'} = 0$).

Let $N_{t,m}$ denote the number of units of type m that are used on trip t. $C_{t,m}$ denotes the number of units of type m that are coupled to the train right before it starts trip t and $U_{t,m}$ denotes the number of units of type m that are uncoupled from the train right after it has completed trip t.

Finally, variables are needed to describe the inventory at stations. Let $I_{t,m}$ denote the inventory of units of type m at station $s_d(t)$, the station from which trip t departs, right after the departure of trip t. Let $I^0_{s,m}$ and $I^\infty_{s,m}$ denote the number of units of type m stored at station s at the start and at the end of the day respectively.

2.2.2 Important Constraints

In this section the important constraints for the model are described. They make sure that all composition changes are allowed and that the inventory at stations remains positive or zero during the day.

The following constraint makes sure that exactly one composition is used during a trip.

$$\sum_{p \in \mathcal{P}_t} X_{t,p} = 1 \qquad \text{for all } t \in \mathcal{T} \tag{2.1}$$

To link the compositions of trips to the transition variables $Z_{t,p,p'}$ the following two constrains are needed:

$$X_{t,p} = \sum_{\substack{p' \in \mathcal{P}_{\sigma(t)}: \\ (p,p') \in \mathcal{G}_t}} Z_{t,p,p'} \qquad \text{for all } t \in \mathcal{T} \setminus \mathcal{T}_\infty, p \in \mathcal{P}_t \tag{2.2}$$

$$X_{\sigma(t),p'} = \sum_{\substack{p \in \mathcal{P}_t: \\ (p,p') \in \mathcal{G}_t}} Z_{t,p,p'} \qquad \text{for all } t \in \mathcal{T} \setminus \mathcal{T}_\infty, p' \in \mathcal{P}_{\sigma(t)} \tag{2.3}$$

$N_{t,m}$, $C_{t,m}$ and $U_{t,m}$ are determined by the following five equations. Note that for trips t with no predecessor trip the number of units that are coupled to t right before the start of this trip is equal to the number of units in t, and similarly for trips t' with no successor trip the number of units that are uncoupled from t' right after the arrival of this trip is equal to the number of units in this trip.

$$N_{t,m} = \sum_{p \in \mathcal{P}_t} \nu(p)_m X_{t,p} \qquad \text{for all } t \in \mathcal{T}, m \in \mathcal{M} \tag{2.4}$$

$$C_{\sigma(t),m} = \sum_{(p,p') \in \mathcal{G}_t} c_m(p,p') \cdot Z_{t,p,p'} \qquad \text{for all } t \in \mathcal{T} \setminus \mathcal{T}_\infty, m \in \mathcal{M} \tag{2.5}$$

$$U_{t,m} = \sum_{(p,p') \in \mathcal{G}_t} u_m(p,p') \cdot Z_{t,p,p'} \qquad \text{for all } t \in \mathcal{T} \setminus \mathcal{T}_\infty, m \in \mathcal{M} \tag{2.6}$$

$$C_{t,m} = N_{t,m} \qquad \text{for all } t \in \mathcal{T}_0, m \in \mathcal{M} \tag{2.7}$$

$$U_{t,m} = N_{t,m} \qquad \text{for all } t \in \mathcal{T}_\infty, m \in \mathcal{M} \tag{2.8}$$

The inventory right after trip t has departed from the station is equal to the initial inventory minus all units that are coupled to trips departing from this station until and including the departure of trip t, plus all units that are uncoupled from trips arriving at this station until the departure of trip t, where the reallocation time is taken into account. This is described by

$$I_{t,m} = I^0_{s_d(t),m} - \sum_{\substack{t' \in \mathcal{T}: s_d(t') = s_d(t), \\ \tau_d(t') \leq \tau_d(t)}} C_{t',m} + \sum_{\substack{t' \in \mathcal{T}: s_a(t') = s_d(t), \\ \tau_d(t') \leq \tau_d(t) - \rho(t')}} U_{t',m} \qquad \text{for all } t \in \mathcal{T}, m \in \mathcal{M} \tag{2.9}$$

$$I^\infty_{s,m} = I^0_{s,m} - \sum_{t \in \mathcal{T}: s_d(t) = s} C_{t,m} + \sum_{t \in \mathcal{T}: s_a(t) = s} U_{t,m} \qquad \text{for all } s \in \mathcal{S}, m \in \mathcal{M} \tag{2.10}$$

Finally, the inventory is linked to the wished inventory by the following two equations:

$$I^0_{s,m} = i^0_{s,m} \qquad \text{for all } s \in \mathcal{S}, m \in \mathcal{M} \tag{2.11}$$

$$I^\infty_{s,m} = i^\infty_{s,m} \qquad \text{for all } s \in \mathcal{S}, m \in \mathcal{M} \tag{2.12}$$

The valid domains of the variables are given by

$$X_{t,p} \in \{0,1\} \qquad \text{for all } t \in \mathcal{T}, p \in \mathcal{P}_t \tag{2.13}$$

$$N_{t,m}, C_{t,m}, U_{t,m}, I_{t,m} \in \mathbb{R}_+ \qquad \text{for all } t \in \mathcal{T}, m \in \mathcal{M} \tag{2.14}$$

$$I^0_{s,m}, I^\infty_{s,m} \in \mathbb{R}_+ \qquad \text{for all } s \in \mathcal{S}, m \in \mathcal{M} \tag{2.15}$$

$$Z_{t,p,p'} \in \mathbb{R}_+ \qquad \text{for all } t \in \mathcal{T}, (p,p') \in \mathcal{G}_t \tag{2.16}$$

Note that only the variables $X_{t,p}$ are forced to be integral. To see why, first note that $I^0_{s,m}$ and $I^\infty_{s,m}$ also have integral values. The variables $Z_{t,p,p'}$ are completely determined by $X_{t,p}$ and $X_{\sigma(t),p'}$ and thus are integral. All the other variables, $N_{t,m}, C_{t,m}, U_{t,m}$ and $I_{t,m}$, are determined by $X_{t,p}, Z_{t,p,p'}, I^0_{s,m}$ and $I^\infty_{s,m}$.

2.2.3 Objective criteria

In the basic model, there are three major objectives: to have enough seats for all the passengers, to limit the number of carriage kilometers, and to minimize the number of shunting movements. Using the decision variables defined before, it is not difficult to design an objective function that takes these things into account.

The number of **carriage kilometers** CKM can be described using the lengths of the trips d_t, the number of carriages per unit c_m and the number of units used per type in trips $N_{t,m}$:

$$CKM = \sum_{t \in \mathcal{T}} \sum_{m \in \mathcal{M}} d_t \cdot c_m \cdot N_{t,m} \tag{2.17}$$

In order to see if there are **enough seats** to accommodate all passengers in a certain feasible solution of the model, the expected number of passengers can be compared to the number of seats in the train compositions calculated by the model. If the latter number is lower, then there are seat shortages $s_{t,p,c}$ for passenger class c on trip t with composition p. Now a useful measure for passenger satisfaction for a trip could be the length of that trip times the number of seat shortages, denoted by SKM. So, noting that d_t is the length in kilometers of trip t:

$$SKM = \sum_{c \in \mathcal{C}} \sum_{t \in \mathcal{T}} \sum_{p \in \mathcal{P}_t} d_t \cdot s_{t,p,c} \cdot X_{t,p} \tag{2.18}$$

The total number of **shunting movements** CCH is equal to the total number of trips where units are coupled to or uncoupled from the train. In most cases this number is equal to the total number of transitions between trips where the trips have a different composition.

$$CCH = \sum_{t \in \mathcal{T}} \sum_{\substack{(p,p') \in \mathcal{G}_t: \\ u_m(p,p') \neq 0 \text{ for any } m \in \mathcal{M} \vee \\ c_m(p,p') \neq 0 \text{ for any } m \in \mathcal{M}}} Z_{t,p,p'} \tag{2.19}$$

Now the objective function for the basic composition model is a linear combination of CKM, SKM and CCH where one can assign weights to the different criteria. Together with the constraints (2.1 - 2.16) from the previous section, this is the basic composition model.

2.2.4 The Transition Graph

The composition model can be interpreted as a single-commodity network flow problem with some additional constraints. This can be seen by considering the graph with the node set

$$\{(t,p) | t \in \mathcal{T}, p \in \mathcal{P}_t\}$$

together with the set of arcs

$$\{((t,p),(\sigma(t),p')) | t \in \mathcal{T} \setminus \mathcal{T}_\infty, (p,p') \in \mathcal{G}_t\}$$

This graph is called the **transition graph**.

Now the constraints (2.1), (2.2) and (2.3) define a network flow in this graph. Figure 2.7 gives an example of a transition graph and figure 2.8 gives an example of a network flow in this graph, which corresponds to some feasible solution of the model.

The transition graph will be used again in chapter 3 to provide more insight in the additions described there. Note that the transition graph only describes part of the problem, since trips are linked in more complex ways, for example by the inventory constraints.

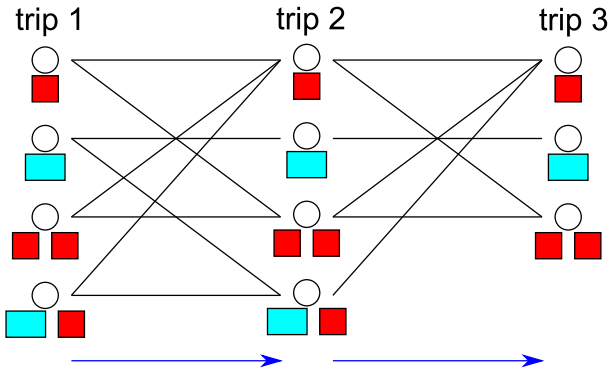


Figure 2.7: An example of a transition graph. This graph describes the possible transitions between trips. For example, if trip 1 has the composition with one red unit, its successor trip (trip 2) can have as possible compositions one red unit or two red units, but not one blue unit or one blue unit at the rear end and a red unit at the front end of the train.

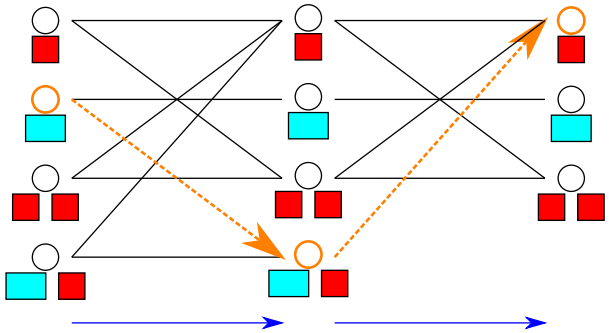


Figure 2.8: An example of a feasible solution of the composition model in the transition graph. Here the first trip consists of one blue unit. After this trip one red unit is coupled to the front of the train. When the second trip is completed, the blue unit is uncoupled from the rear of the train and on the last trip the train only consists of a red unit.

2.3 Additional Constraints

Some extensions to the basic model are needed to be able to handle more realistic scenarios. Furthermore, by introducing some new decision variables one can improve the formulation of the model which can reduce computation time. The latter is discussed in section 2.3.1. Section 2.3.2 describes how one can incorporate combining and splitting into the model and in section 2.3.3 a way to model the continuity constraint is described.

2.3.1 Additional decision variables

In order to reduce the number of binary variables in the model, one can introduce new binary variables which only describe the number of units of a certain type in the train. It turns out that then one can always assign compositions to trips in an optimal solution without demanding that the $X_{t,p}$ variables are integral.

Let $\mathcal{B}_t = \{\nu(p)|p \in \mathcal{P}_t\}$, so $\mathcal{B}_t$ is a set of vectors in $\mathbb{Z}_+^{\mathcal{M}}$ which describes the number of units in the train on trip t for every type of unit. Now new binary decision variables can be defined: $Y_{t,b} \in \{0,1\}$ indicates whether the number of units specified in $b \in \mathcal{B}_t$ is used in trip t. It is easy to connect this with variables $X_{t,p}$:

$$Y_{t,b} = \sum_{p \in \mathcal{P}_t : \nu(p) = b} X_{t,p} \qquad \text{for all } t \in \mathcal{T}, b \in \mathcal{B}_t \tag{2.20}$$

The variable $Y_{t,b}$ can intuitively be seen as a more abstract decision variable than $X_{t,p}$. It turns out that one can drop the integrality constraint on $X_{t,p}$: the composition model extended with $Y_{t,b}$ and with the integrality constraint on $X_{t,p}$ relaxed has an integral optimal solution if and only if it has a feasible solution.

To see this, consider an optimal solution of the problem where $X_{t,p}$ and $Z_{t,p,p'}$ may be fractional. Now consider the transition graph, where the capacity of an arc is set to zero if $Z_{t,p,p'}$ is zero and one if $Z_{t,p}$ is greater than zero. In this graph one can find an integer valued network flow. One can also show that the variables $N_{t,m}, C_{t,m}, U_{t,m}$ as well as the objective criteria depend only on $Y_{t,b}$. When some more additions are introduced later, it can be shown that the integrality of $Y_{t,b}$ does not always guarantee the integrality of $X_{t,p}$, but in practice it still works and improves the solution time.

2.3.2 Combining and Splitting

A major concept not included in the basic model is combining and splitting of trains: some trips can have two successor trips or two predecessor trips. Since the modelling of splitting and combining are roughly the same, only a detailed description of splitting is included here.

Firstly, some new notation is needed. Let $\mathcal{T}^s$ denote the set of trips for which the train is split after the trip and both parts continue in different trips. So for $t \in \mathcal{T}^s$ there are two successor trips $\sigma^1(t)$ and $\sigma^2(t)$. Let $\mathcal{G}_t^s$ denote the set of triples of compositions (p, p_1, p_2) such that $p \in \mathcal{P}_t$, $p_1 \in \mathcal{P}_{\sigma^1(t)}$ and $p_2 \in \mathcal{P}_{\sigma^2(t)}$, and also such that this composition change is allowed by the shunting restrictions. Normally, p will be a concatenation of the two compositions p_1 and p_2, although one can also model splitting with coupling and uncoupling of units. Note that the way p_1 and p_2 are concatenated in p depends on whether $\sigma^1(t)$ or $\sigma^2(t)$ continue in the opposite direction of trip t or not: when the two departing trains continue in the reverse direction from the arriving train that was split, p is actually the reverse of the concatenation of the two compositions p_1 and p_2.

Let variables $Z^s_{t,p,p_1,p_2} \in \{0,1\}$ be 1 if trip t has composition p, trip $\sigma^1(t)$ has composition p_1 and trip $\sigma^2(t)$ has composition p_2 for all $t \in \mathcal{T}^s$ and $(p, p_1, p_2) \in \mathcal{G}_t^s$, and 0 otherwise. Analogously to constraints in the basic model these variables can be linked to variables $X_{t,p}$ by the following constraints:

$$X_{t,p} = \sum_{p_1,p_2:(p,p_1,p_2)\in\mathcal{G}_t^s} Z^s_{t,p,p_1,p_2} \qquad \text{for all } p \in \mathcal{P}_t \tag{2.21}$$

$$X_{\sigma^1(t),p_1} = \sum_{p,p_2:(p,p_1,p_2)\in\mathcal{G}_t^s} Z^s_{t,p,p_1,p_2} \qquad \text{for all } p_1 \in \mathcal{P}_{\sigma^1(t)} \tag{2.22}$$

$$X_{\sigma^2(t),p_2} = \sum_{p,p_1:(p,p_1,p_2)\in\mathcal{G}_t^s} Z^s_{t,p,p_1,p_2} \qquad \text{for all } p_2 \in \mathcal{P}_{\sigma^2(t)} \tag{2.23}$$

Some additional changes are needed to incorporate splitting completely into the model. If coupling is allowed to the first departing train $\sigma^1(t)$ then $C_{\sigma^1(t),m}$ is given by

$$C_{\sigma^1(t),m} = \sum_{(p,p_1,p_2)\in\mathcal{G}_t^s} c(p,p_1,p_2)\cdot Z^s_{t,p,p_1,p_2} \qquad \text{for all } m \in \mathcal{M} \tag{2.24}$$

$$C_{\sigma^2(t),m} = 0 \qquad \text{for all } m \in \mathcal{M} \tag{2.25}$$

where $c(p,p_1,p_2)$ describes the number of units that are coupled to $\sigma^1(t)$ during a transition from p to p_1 and p_2. Here $C_{\sigma^2(t),m} = 0$ since normally no units are coupled to the rear of the train, especially not during splitting since that is already a quite complex shunting movement.

And if uncoupling from trip t is allowed $U_{t,m}$ is given by

$$U_{t,m} = \sum_{(p,p_1,p_2)\in\mathcal{G}_t^s} u(p,p_1,p_2)\cdot Z^s_{t,p,p_1,p_2} \qquad \text{for all } m \in \mathcal{M} \tag{2.26}$$

Where $u(p,p_1,p_2)$ describes the number of units that are uncoupled from the train during a transition from p to p_1 and p_2. Depending on the objective function, more changes must be made to the model to incorporate splitting. For example, a term

$$\sum_{t\in\mathcal{T}^s} \sum_{\substack{(p,p_1,p_2)\in\mathcal{G}_t^s:\\ c(p,p_1,p_2)\neq 0 \vee u(p,p_1,p_2)\neq 0}} Z^s_{t,p,p_1,p_2}$$

needs to be added to the objective function with some weighting factor in order to take into account the number of shunting movements during splitting.

Introducing combining of trains into the model is quite similar to splitting. Let $\mathcal{T}^c$ be the set of trips t which have two predecessor trips t_1 and t_2, so trips in $\mathcal{T}^c$ are combined trains. Let $\mathcal{G}_t^c$ denote the set of triples of compositions (p,p_1,p_2) such that $p \in \mathcal{P}_t$, $p_1 \in \mathcal{P}_{t_1}$ and $p_2 \in \mathcal{P}_{t_2}$ and such that this composition change is allowed by the shunting restrictions. Note again that normally p will be a concatenation of the two compositions p_1 and p_2 unless coupling or decoupling takes place which is rarely the case, and that the specific concatenation depends on the directions from which t_1 and t_2 arrive and the direction to which t departs.

Now one can introduce variables Z^c_{t,p,p_1,p_2} with $t \in \mathcal{T}^c$ and $(p,p_1,p_2) \in \mathcal{G}_t^c$ with similar constraints as given for splitting.

It turns out that in the basic model integrality for combining and splitting still holds: if the composition variables $X_{t,p}$ are forced to be integral, the transition variables Z_{t,p_1,p_2} will be integral without explicitly requiring this. But with the additional decision variable $Y_{t,b}$ there can theoretically be some problems: one can construct examples where the composition variables $X_{t,p}$ are relaxed and the variables $Y_{t,b}$ are forced to be integral and where the transition variables are fractional in an optimal solution. These examples are quite pathological though, and in practice it turns out that relaxing the $X_{t,p}$ variables is never a problem for integrality.

2.3.3 Continuity constraints

A constraint on composition changes is that for each train there should be at least one unit that participates in all corresponding trips. In this section a simple way to model this is given, a more elaborate way that needs less assumptions and can include combining and splitting is introduced in the next chapter.

The idea is to do a kind of bookkeeping on how much units are coupled and uncoupled from both sides of the train, and to check whether there is a train unit which is never uncoupled during these operations. Consider the sequence of trips $t_1, ..., t_k$ where $t_{i+1} = \sigma(t_i)$. By theoretically allowing coupling and uncoupling from both sides of the train it can be assumed without loss of generality that the train moves in one direction, so the train never turns at a station. This way it is easier to define the 'left' and 'right' side of the train. Units in the train are numbered increasingly starting from the left side of the train. Let α_i^L be the number of units that are uncoupled from the left of the train after trip t_i and let β_i^L be the number of units that are coupled to the left of the train after trip t_i. Note that normally either β_i^L or α_i^L will be zero. Similarly one can define α_i^R and β_i^R to be the number of units that are uncoupled from respectively coupled to the right of the train after trip t_i. Let γ_i be the number of units that are in the train used for trip t_i. Note that $\gamma_i = \gamma_{i-1} - \alpha_{i-1}^L - \alpha_{i-1}^R + \beta_{i-1}^L + \beta_{i-1}^R$ and that the variables introduced here follow directly from the composition variables $X_{t,p}$.

If the continuity constraint holds, there must be a unit that is in all trips. Let the position of this unit in the train that is assigned to trip t_i be ℓ_i. Using α_i^L, α_i^R, β_i^L and β_i^R, one can 'follow' the position of the unit. The new position ℓ_{i+1} of the unit in trip t_{i+1} is equal to the old position ℓ_i minus all units that are uncoupled from the left plus all units that are coupled to the left:

$$\ell_i = \ell_{i-1} - \alpha_{i-1}^L + \beta_{i-1}^L \qquad \text{for all } i = 2, ..., k \tag{2.27}$$

Since the unit cannot be uncoupled from the right after trip t_i the following constraint must hold:

$$\ell_i \leq \gamma_i - \alpha_i^R \qquad \text{for all } i = 2, ..., k \tag{2.28}$$

Furthermore we have that $1 \leq \ell_i \leq \gamma_i$ and $\ell_i \in \mathbb{Z}$. Now if there is a sequence $\ell_1, ..., \ell_k$ such that the equations above hold then the continuity constraint holds. Since $\ell_2, ..., \ell_k$ are

determined by l_1 another way to describe the constraints is:

$$1 \leq \ell_1 \leq \gamma_1 \tag{2.29}$$

$$1 \leq \ell_1 - \sum_{j<i} \alpha_j^L + \sum_{j<i} \beta_j^L \leq \gamma_i - \alpha_i^R \qquad \text{for all } i = 2, ..., k-1 \tag{2.30}$$

$$1 \leq \ell_1 - \sum_{j<k} \alpha_j^L + \sum_{j<k} \beta_j^L \leq \gamma_k \tag{2.31}$$

Since there are only integral bounds in the equations above, we can choose $\ell_1 \in \mathbb{R}$.

In section 3.6 another way to model the continuity constraint is presented where some assumptions will be dropped.

2.4 Implementation

Translating the more abstract constraints from the previous section to a practical situation and implementing it is far from trivial. In this section some insight is given into how this can be done, and some important implementation issues are pointed out.

Based on several input files, including a file with information about the trips and a file with information about the different train units used, constraints are formulated using the program OPL Studio from ILOG. OPL Studio then uses the CPLEX MIP solver to find an (optimal) solution to the problem. In section 2.4.1 is explained how to determine which shunting movements are allowed between trips using **standard shunting codes**. Section 2.4.2 sketches how the MIP problem is solved and gives some ways to fine tune the CPLEX MIP solver in order to obtain good solutions quickly. More is said about optimizing the solution process in chapter 4. As the output of the model is only a list of compositions for trips, it needs to be processed to determine duties for rolling stock. An outline of how this is done is given in section 2.4.3.

2.4.1 Shunting Movements

In formulating the constraints, one of the main difficulties is to determine what kind of shunting movements are allowed for specific trips. This requires a lot of knowledge of local stations that is often hard to obtain. Also some shunting movements might be feasible but not desirable, for example if it would take a lot of manpower to execute the shunting plans. In the model some codes for standard shunting movements have been introduced and every trip is assigned such a code. This shunting code determines which transitions are allowed, so which elements belong to $\mathcal{G}_t$. One can assign shunting codes to trips based on information about particular stations. Or if one already has an original plan, one can interpret the shunting movements used there and assign shunting codes to trips based on that.

The following standard shunting codes are the most commonly used shunting codes in the model:

0 After this trip, the train has no successor trips and goes to the shunting yard. Later on, after the reallocation time $\rho(t)$, units of this train can be reused in other trips.

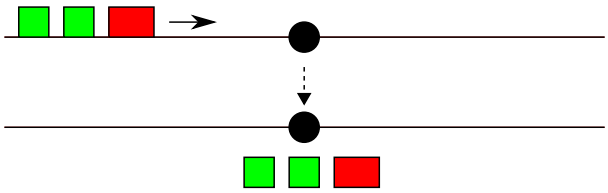


Figure 2.9: Shunting code **0**: The entire train goes to the shunting yard.

X The train goes on in the same riding direction after this trip without composition changes.

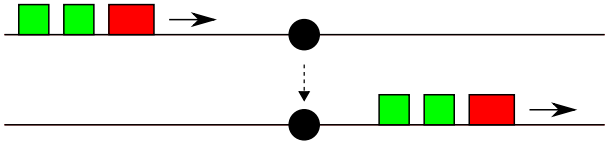


Figure 2.10: Shunting code **X**: The entire train continues in the same riding direction.

aXb The train goes on in the same riding direction after this trip. Train units can be coupled to the front of the train or uncoupled from the rear of the train, but not both at the same time.

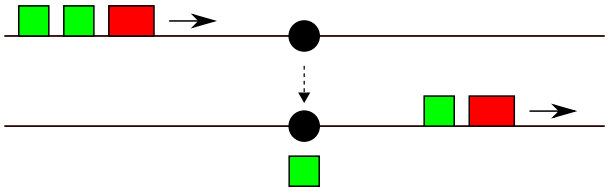


Figure 2.11: Shunting code **aXb**: In this example a train unit is uncoupled from the rear of the train.

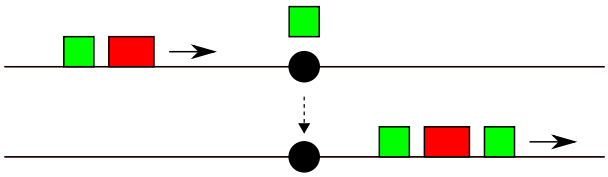


Figure 2.12: Shunting code **aXb**: In this example a train unit is coupled to the front of the train.

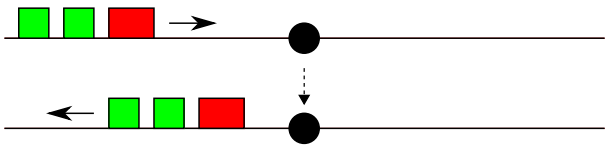


Figure 2.13: Shunting code **K**: The entire train continues in the opposite direction.

K The train changes riding direction after the trip.

Kab The train changes riding direction after the trip. Units can be coupled to the front of the train or uncoupled from the front of the train, where the front of the train is defined to be the side of the train that enters the station first when arriving. This is often the case when the shunting yard is on the other side of the station as the direction the train arrives from.

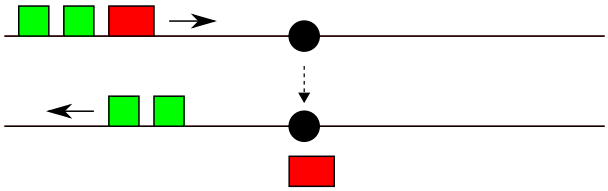


Figure 2.14: Shunting code **Kab**: In this example the train changes riding direction and a unit is uncoupled from the front of the train.

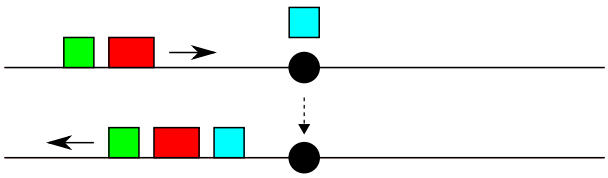


Figure 2.15: Shunting code **Kab**: In this example the train changes riding direction and a unit is coupled to the front of the train.

abK The train changes riding direction after the trip. Units can be coupled to the rear of the train or uncoupled from the rear of the train, where the rear of the train is defined to be the side of the train that enters the station lastly when arriving. This is often the case when the shunting yard is on the same side of the station as the direction the train arrives from.

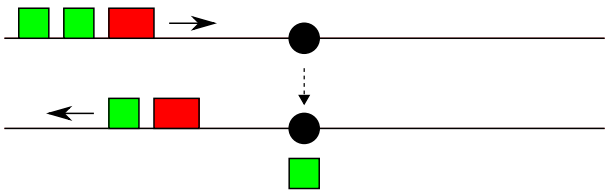


Figure 2.16: Shunting code **abK**: The train changes riding direction and a train unit is uncoupled from the back of the train.

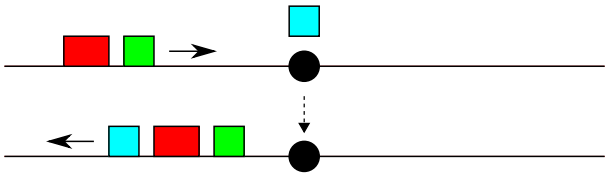


Figure 2.17: Shunting code **abK**: The train changes riding direction and a train unit is coupled to the back of the train.

S The train is split into two parts after this trip. Both parts will continue in the same riding direction as this train.

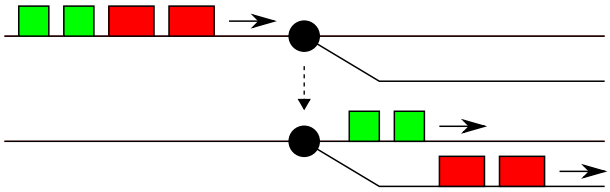


Figure 2.18: Shunting code **S**: The train is split into two parts after this trip.

C The train will be combined with another train after this trip.

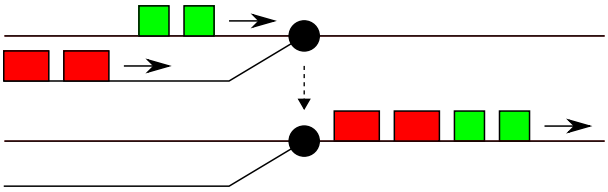


Figure 2.19: Shunting code **C**: The train is combined with another train.

SaXb The train is split into two parts after this trip, and uncoupling from the back or coupling to the front of the train is allowed.

CaXb The train is combined with another train after this trip, and uncoupling from the back or coupling to the front of the train is allowed depending on whether this train will be the front or the back part of the combined train.

SK The train is split into two after this trip and both parts change riding direction.

CK The train will be combined with another train and the resulting train changes riding direction.

Based on these standard shunting codes, certain composition changes are allowed for trips. Usually at most two units are allowed to be uncoupled or coupled to a train, excluding trips with no successor or predecessor trip. In some cases the standard shunting codes do not suffice, for example if one wants to take into account an original plan where an exceptional

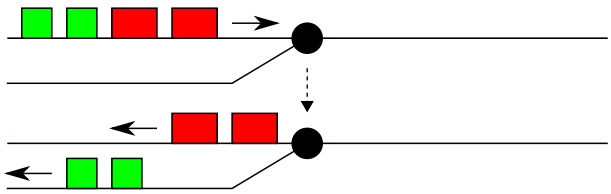


Figure 2.20: Shunting code **SK**: The train is split into two trains and both change riding direction.

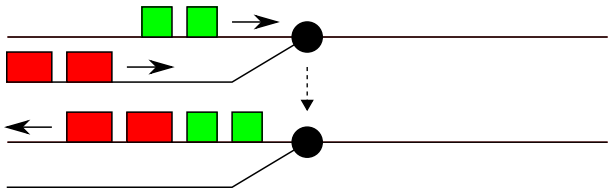


Figure 2.21: Shunting code **CK**: The train is combined with another train and the resulting train changes riding direction.

shunting code was used. In section 3.3 a way to allow exceptional shunting movements is described.

2.4.2 Solving the Mixed Integer Programming Problem

Fine tuning the solution process by the CPLEX MIP solver is another problem. The Mixed Integer Programming problem that is described by the composition model is quite complex and can take hours of calculation time. Therefore it is important to choose the right solving parameters. In this section an outline is given of how the MIP problem is solved by CPLEX and in what way this solution process can be influenced.

The MIP problem is solved by a **branch and bound** process. First the **linear relaxation** of the problem is solved. If there are fractional values in the solution, two subproblems are created by branching on a fractional variable. This process continues until an optimal integral solution is found. To solve the initial linear relaxation an **interior point algorithm** is used, the Barrier algorithm, and for the subproblems in the branching process the **dual simplex algorithm** is used. The reason for choosing these two algorithms is that the initial linear relaxation is quite a large problem, and interior point algorithms seem to work better here. When solving the subproblems, information is used about the initial solution (or other solved subproblems) and the dual simplex method works well here.

Several techniques are used to improve the solution time. **Probing techniques** are used to improve the problem formulation before solving the initial linear relaxation. In order to find integral solutions faster several techniques are used to reduce the solution space, including cutting plane techniques. Parameters can be set that influence how often CPLEX applies these techniques.

One can set **priorities** for variables to influence the branching process. So if there is more than one fractional variable in a solution of a subproblem, CPLEX can choose the new branching variable based on their priorities. A way that seems to work well for the composition model is to first determine the integral variables corresponding to the morning rush hours and then solve the remaining day. An explanation for this is that the morning rush hours are the bottleneck of the model since then the passenger demand is the highest. If the problem is optimized here firstly, the rest of the problem becomes a lot easier. Also, first determining solutions for variables that 'belong' together instead of randomly determining variables might work better in the branching process. If two unrelated variables are fixed in a particular subproblem, it might take a lot of time before it is found out that there is no integral solution corresponding to the values chosen for them. But if the variables are 'related' then it can be expected that incompatibility in their values is detected earlier. This is also an argument for setting priorities based on to what time of the day the variables correspond.

When not planning 'from scratch' as will be described in the next chapter, this method of assigning priorities does not seem to work well. There the objective function is different which gives different relations between variables and also gives different bottlenecks in the problem. Several other ways to get a solution faster are discussed in chapter 4, where the focus lies on solving the model presented in the next chapter. One major observation in the model described in this chapter, which also holds for the modified model from the next chapter, is that the linear relaxation of the problem lies quite close to the optimal integral solution. This information can be used to obtain solutions quickly.

Preprocessing, creating the constraints and variables that are used as input for CPLEX, can take quite a lot of time. Some programming techniques to reduce this are briefly discussed in appendix B.

2.4.3 Determining Duties

The output of the model is a list of trips with determined compositions. As said at the beginning of this chapter, it is always possible to determine the duties for the rolling stock units from this. In this section it is outlined how this is done. Note that there can be more ways to create a duty roster for the rolling stock based on a given solution for the composition model.

The basic idea to determine the duties is:

1. Identify all **tasks** that need to be carried out, basically for every unit in every trip a task is created.
2. Make **chains** of tasks, determine what tasks will need to be carried out by the same unit. For example, if a unit is assigned to trip t it will also be assigned to its successor trip $\sigma(t)$ unless the unit is uncoupled after trip t.
3. Combine the chains into **duties**.

The chains obtained in step 2 describe a part of a duty from the moment a unit leaves the shunting yard until a unit returns to the shunting yard. For step 3 one can apply different strategies. A possible solution is to connect a chain that ends with a unit arriving at the shunting yard of a particular station to the chain that starts with a unit of the same type departing from that station, where the time between arriving and departing is minimal. This is a sort of greedy search strategy. More complex strategies are possible, for example when one wants the duties to resemble earlier created duties.

Chapter 3

Additions to the Model

In this chapter some additions to the model are described. The composition model described in the previous chapter was designed to solve the tactical rolling stock problem, which is solved 'from scratch' months before the execution time of the plans. The additions described in this chapter aim to make the composition model capable of modifying an existing plan and taking into account some relevant factors when planning closer to the execution time of the plan.

Section 3.1 describes how a previously created plan could be used as a basis to create a new plan with the composition model. Furthermore, some important observations are made from studying such an original plan, which form the basis of several additions described in later sections. Section 3.2 describes a more general variant of the composition model introduced in the previous chapter, which will make it easier to introduce the additions presented here.

Next, section 3.3 describes how some exceptional shunting movements can be included in the model, which allows a previously designed plan to be a feasible input for the model even if manual changes were made in that plan. In order to take into account more details of the shunting movements at stations, section 3.4 introduces an addition to the model which keeps track of combined units in the inventory at stations instead of only keeping a list of individual units in the inventory, and section 3.5 describes a way to incorporate a special kind of fast shunting movements. An alternative way to implement the continuity constraint described in the previous chapter is given in section 3.6, which is more robust and can incorporate all relevant trips. When introducing the additions into the model, an important factor to keep in mind is the **integrality** of the problem. Where relevant, remarks will be made about this.

Finally, section 3.7 gives some remarks about how to use the additions and discusses their relevancy.

The additions described in this chapter can make the model more suitable for short term planning where an original plan must be modified and shunting details play a more important

role. But they could also add some more relevant details to the tactical rolling stock planning problem.

3.1 Using a Previously Created Plan

An important goal of this report is to describe how to take into account an original plan in order to plan closer to the execution date of the plan. One would like the model to **modify** the original plan instead of building a new rolling stock plan 'from scratch'. Furthermore, observations about an original rolling stock plan are the basis for most of the additions described in this chapter. This section discusses what an original plan consists of and how it could be used.

Note that most additions described in this chapter do not necessarily require an original plan in order to be implemented.

3.1.1 Input for the Model

An original plan mainly consists of a list of duties for train units and information about trips. From this plan information can be extracted that can be useful as input for the model – the output of the model should be a modification of the original plan instead of an entirely new plan when planning in the operational rolling stock phase. Several possible usages are:

- Determine standard shunting codes for trips, which give allowed transitions in the model.
- The model can take into account the original compositions from trips and try to minimize changes in the new plan, or give penalties to bad changes.
- The model can take into account the original shunting plans for transitions, for example by giving penalties for extra shunting movements or changed shunting movements.

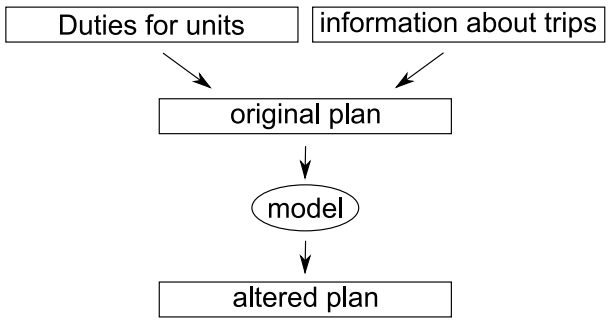


Figure 3.1: Using an original plan to create a new plan.

Obtaining the right information from a list of duties for train units and information about trips is not trivial. The standard shunting codes used in the implementation of the model are not standard notions used at NS, so one needs to interpret what happens with different train units on trips in order to fit a standard shunting code to the transition after a trip. If no such standard shunting code can be found, either there is an error in the input files or there is an exceptional shunting movement that cannot be described by the standard shunting codes. In section 3.3 more is said about exceptional shunting movements.

Often it is useful to relax the detected shunting codes a bit to allow some more freedom in the model. For example shunting code **X** may be replaced by **aXb**, so instead of only allowing a train to pass through a station, units can be coupled to the front of the train and uncoupled from the rear of the train. Although relaxing shunting codes gives more freedom for the model, it also makes the model more complex since it increases the possibilities and thus also the size of the solution space. Therefore, when time is a critical factor it may be better to avoid relaxing shunting codes.

3.1.2 Observations from an Original Plan

In order to analyze what happens at the shunting yard of a station one can make a table of the shunting movements at the shunting yard of a particular station, see figure 3.2. This table is based on planned duties for trips. It describes every unit or composition of units that arrives at the shunting yard and everything that departs from the shunting yard. A distinction is made between starters, complete trains that depart from the shunting yard, and units that are coupled to trains that arrive at the station. Similarly a distinction is made between finishers, complete trains that arrive at the station, and units that are uncoupled from trains arriving at the station.

Two important observations can be made when studying this type of shunting tables:

- Firstly, sometimes it occurs that a train arrives and is redeployed very quickly (within 30 minutes) without having a planned successor trip in the timetable for train lines. Although the original model would interpret the arrival as an arrival to the shunting yard and assign a reallocation time of 30 minutes, in practice the train will just wait a while at the station instead of being taken to the shunting yard.
- A second observation is that entire trains arrive at the shunting yard and later on leave the shunting yard without any composition changes.

Both examples can be observed in figure 3.2. In section 3.4 a way to keep track of which trains are stored at the shunting yard is introduced, and in section 3.5 a way to incorporate the observed ‘fast shunting movements’ is described.

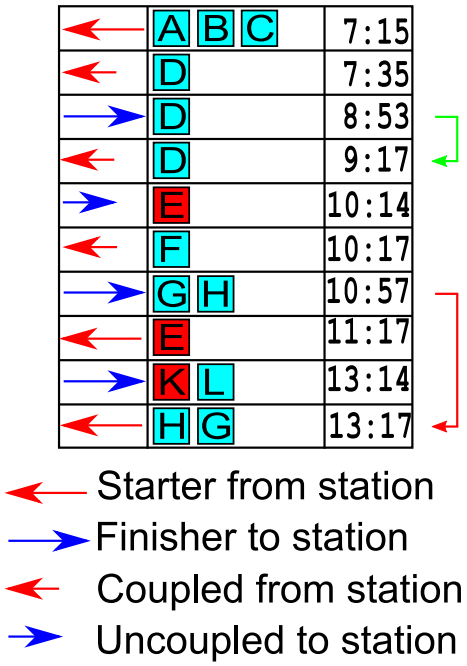


Figure 3.2: Part of a shunting table for a station. Note the green arrow indicating a fast shunting movement and the red arrow indicting a complete train entering and leaving the shunting yard without composition changes.

3.2 Generalizing the composition model

In this section the composition model is generalized and redefined a bit. This will make it easier to introduce new notation and constraints in the upcoming sections. The idea is to generalize transitions and compositions such that they can include additional information. Parallel transitions between trips with given compositions will become possible and in the model a composition of a trip can occur more than once with different properties, see figure 3.3 of a picture of how this effects the transition graph.

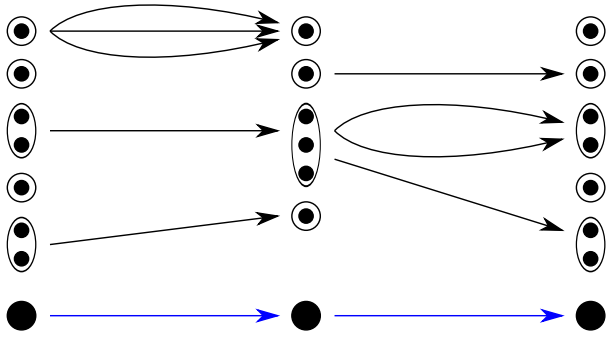


Figure 3.3: The transition graph in the generalized composition model.

Let $\mathcal{P}_t$ denote the set of **extended compositions**, and let $p(e)$ denote the actual composition of $e \in \mathcal{P}_t$ without extra information added. Similarly, let $\mathcal{G}_t$ denote the set of **extended**

transitions, where $p(a) \in p(\mathcal{P}_t)$ denotes the composition of trip t and $p'(a) \in p(\mathcal{P}_{\sigma(t)})$ denotes the composition of trip $\sigma(t)$ for $a \in \mathcal{G}_t$. Conceptually, in the generalized composition model a composition of units for a trip can be seen as a 'super-node' consisting of 'sub-nodes' corresponding to the possible extended compositions for that composition. The super-nodes are connected through transition arcs, where parallel arcs are allowed.

The main decision variables are $X_{t,e} \in \{0,1\}$ which denote whether extended composition $e \in \mathcal{P}_t$ is used in trip t and $Z_{t,a} \in \{0,1\}$ which denote whether extended transition $a \in \mathcal{G}_t$ is used between trips t and $\sigma(t)$.

As in the original composition model, $C_{t,m}$ denotes the number of units of type m that are coupled to the train right before it starts trip t and $U_{t,m}$ denotes the number of units of type m that are uncoupled from the train right after it has completed trip t. Let $c_m(a)$ denote the number of units of type m that are coupled to a train during extended transition a, and let $u_m(a)$ denote the number of units of type m that are uncoupled from the train during extended transition a. For starters $t \in \mathcal{T}_0$, let $c_m^S(e)$ denote the number of units of type m in trip t if extended composition $e \in \mathcal{P}_t$ is chosen, where S stands for Starter. And for finishers $t \in \mathcal{T}_\infty$ let $u_m^F(e)$ denote the number of units of type m in trip t if extended composition $e \in \mathcal{P}_t$ is chosen, where F stands for Finisher.

Now one can redefine the constraints of the composition model.

Exactly one extended composition is used on each trip:

$$\sum_{e \in \mathcal{P}_t} X_{t,e} = 1 \qquad \text{for all } t \in \mathcal{T} \tag{3.1}$$

Extended compositions and extended transitions need to be linked, for every possible composition q we have:

$$\sum_{\substack{e \in \mathcal{P}_t: \\ p(e)=q}} X_{t,e} = \sum_{\substack{a \in \mathcal{G}_t: \\ p(a)=q}} Z_{t,a} \qquad \text{for all } t \in \mathcal{T} \setminus \mathcal{T}_\infty \tag{3.2}$$

$$\sum_{\substack{e \in \mathcal{P}_{\sigma(t)}: \\ p(e)=q}} X_{\sigma(t),e} = \sum_{\substack{a \in \mathcal{G}_t: \\ p'(a)=q}} Z_{t,a} \qquad \text{for all } t \in \mathcal{T} \setminus \mathcal{T}_\infty \tag{3.3}$$

Coupling and uncoupling:

$$C_{\sigma(t),m} = \sum_{a \in \mathcal{G}_t} c_m(a) \cdot Z_{t,a} \qquad \text{for all } t \in \mathcal{T} \setminus \mathcal{T}_\infty, m \in \mathcal{M} \tag{3.4}$$

$$U_{t,m} = \sum_{a \in \mathcal{G}_t} u_m(a) \cdot Z_{t,a} \qquad \text{for all } t \in \mathcal{T} \setminus \mathcal{T}_\infty, m \in \mathcal{M} \tag{3.5}$$

$$C_{t,m} = \sum_{e \in \mathcal{P}_t} c^S_m(e) \cdot X_{t,e} \qquad \text{for all } t \in \mathcal{T}_0, m \in \mathcal{M} \tag{3.6}$$

$$U_{t,m} = \sum_{e \in \mathcal{P}_t} u^F_m(e) \cdot X_{t,e} \qquad \text{for all } t \in \mathcal{T}_\infty, m \in \mathcal{M} \tag{3.7}$$

Inventory constraints remain the same:

$$I_{t,m} = I^0_{s_d(t),m} - \sum_{\substack{t' \in \mathcal{T}: s_d(t') = s_d(t), \\ \tau_d(t') \le \tau_d(t)}} C_{t',m} + \sum_{\substack{t' \in \mathcal{T}: s_a(t') = s_d(t), \\ \tau_d(t') \le \tau_d(t) - \rho(t')}} U_{t',m} \qquad \text{for all } t \in \mathcal{T}, m \in \mathcal{M} \tag{3.8}$$

$$I^\infty_{s,m} = I^0_{s,m} - \sum_{t \in \mathcal{T}: s_d(t) = s} C_{t,m} + \sum_{t \in \mathcal{T}: s_a(t) = s} U_{t,m} \qquad \text{for all } s \in \mathcal{S}, m \in \mathcal{M} \tag{3.9}$$

$$I^0_{s,m} = i^0_{s,m} \qquad \text{for all } s \in \mathcal{S}, m \in \mathcal{M} \tag{3.10}$$

$$I^\infty_{s,m} = i^\infty_{s,m} \qquad \text{for all } s \in \mathcal{S}, m \in \mathcal{M} \tag{3.11}$$

The valid domains of the variables are given by

$$X_{t,e} \in \{0, 1\} \qquad \text{for all } t \in \mathcal{T}, e \in \mathcal{P}_t \tag{3.12}$$

$$N_{t,m}, C_{t,m}, U_{t,m}, I_{t,m} \in \mathbb{R}_+ \qquad \text{for all } t \in \mathcal{T}, m \in \mathcal{M} \tag{3.13}$$

$$I^0_{s,m}, I^\infty_{s,m} \in \mathbb{R}_+ \qquad \text{for all } s \in \mathcal{S}, m \in \mathcal{M} \tag{3.14}$$

$$Z_{t,a} \in \mathbb{R}_+ \qquad \text{for all } t \in \mathcal{T}, a \in \mathcal{G}_t \tag{3.15}$$

Note that in the model above the integrality of $Z_{t,a}$ is not ensured. As long as there are no parallel transitions between two compositions the integrality of $X_{t,e}$ ensures the integrality of $Z_{t,a}$, but otherwise one needs to be careful. The following sections will introduce parallel transitions and where needed extra decision variables are introduced to make sure that exactly one transition is chosen between two trips.

Also note that introducing the integral $Y_{t,b}$ variables from the previous chapter into the generalized model and dropping the integrality constraints on variables $X_{t,e}$ might not be sufficient. More will be said about this in later sections.

3.3 Exceptional Shunting Movements

In most cases, the standard shunting movements described in section 2.4 suffice to describe what is happening at stations, but sometimes exceptional shunting movements take place that cannot be interpreted in terms of the standard shunting movements. In this section a way to incorporate these exceptional shunting movements into the model is described. In general one would like to avoid exceptional shunting movements as they require complex shunting operations, but if they were already planned out of necessity then it is desirable to keep them in a new plan. It also adds some more flexibility to the model and makes an original plan a feasible solution to the model. The latter can improve the solution time and also fits into the idea of this chapter to alter the purpose of the composition model – to modify an original plan instead of planning 'from scratch'.

Section 3.3.1 gives some examples of exceptional shunting movements. Then a way to incorporate these shunting movements into the model is described in section 3.3.2. It turns out that this is not difficult. Much more difficult is the implementation, as exceptional shunting movements need to be detected in an original plan and one needs to keep track of what happens with the units after the shunting process. This problem is dealt with in section 3.3.3 where the concept of the shunting index is introduced.

3.3.1 Examples

Figures 3.4 and 3.5 give some examples of exceptional shunting movements that were planned manually but that are not covered by the standard shunting movements used in the composition model as described in section 2.4.

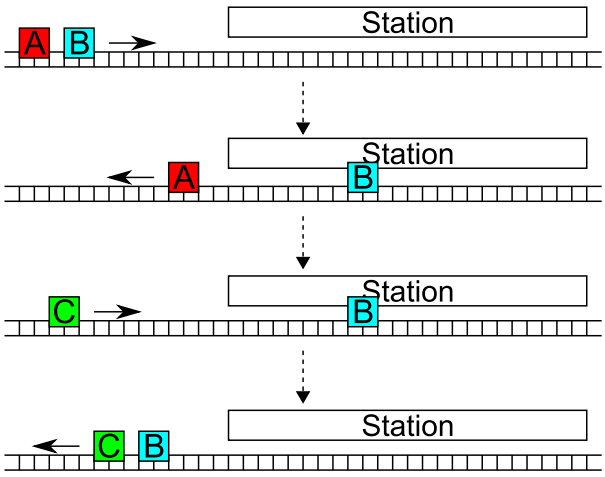


Figure 3.4: An example of an exceptional shunting movement. A train with units A and B arrives at the station, unit A is uncoupled from the rear of the train and unit C which was waiting at the station is coupled to the front of the train. This shunting movement is not incorporated in the standard shunting movements since it requires a lot of crew members and time.

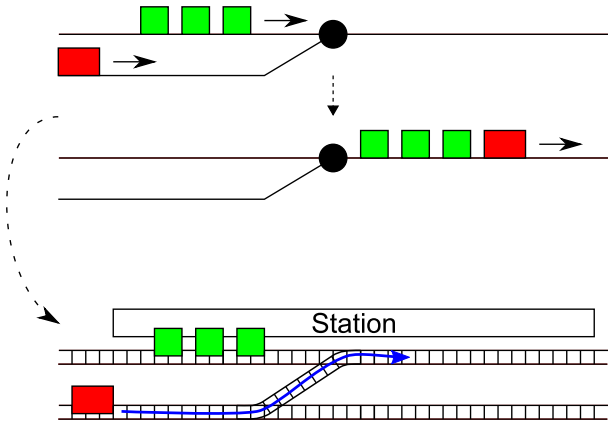


Figure 3.5: An example of an exceptional shunting movement when combining two trains. Normally the first arriving train (in this case the green train consisting of 3 units) will become the front of the departing combined train and the last arriving train (in this case the red train consisting of one unit) will become the back of the departing combined train but in this case the order is switched. In the lower part of the figure a possible explanation for this is given.

3.3.2 Model

Incorporating exceptional shunting movements into the model is not difficult since one only needs to add new elements to $\mathcal{G}_t$ and the model description remains the same. This corresponds to adding an arc to the transition graph, see figure 3.6. One important issue is that the integrality of the variables needs to be guaranteed, remarks about this are made below.

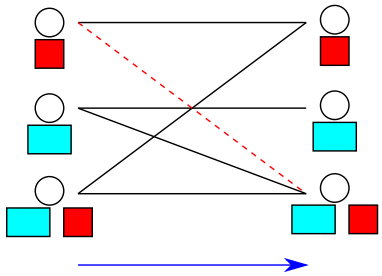


Figure 3.6: Adding an arc to the transition graph. In this case, coupling a blue unit to the rear of the red unit is included as an extra allowed transition.

Integrality

Although it is not common, even in exceptional shunting movements, it is possible that adding arcs to the transition graph gives parallel lines. That is, there may be more than one way to go from one composition to another. An example of this is given in figure 3.7. Here a unit is added to a train, but adding the unit to the rear of the train gives the same composition as adding the unit to the front of the train. Because choosing either shunting movement has the same effect on the inventory, both shunting movements are feasible or both are infeasible. Therefore, if one of the two possible transitions is better in the objective function it will be chosen, and if both transitions have the same effect on the objective function, one can be chosen arbitrarily. This means that there is always an integral solution corresponding to a feasible solution.

But if there are parallel transitions which have a different effect on the inventory, then it might be possible that a fractional solution is optimal with no corresponding integral optimal solution of the same value. These pathological cases may cause trouble but have not been observed so far and could be countered with extra decision variables. Later in this chapter examples where additional decision variables will be needed are given.

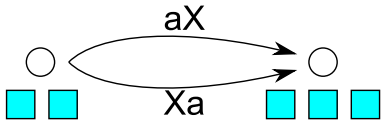


Figure 3.7: Parallel extension: there are two transitions possible that give the same composition change.

3.3.3 Implementation

Incorporating the exceptional shunting movements into the model turns out to be rather trivial conceptually, no real changes need to be made to the model. But actually implementing this change is much more difficult. There are three major problems that need to be solved:

- Exceptional shunting movements need to be detected if an original plan is used as input for the model.
- Exceptional shunting movements need to be fed into the model.
- They need to be interpreted correctly in order to keep track of what happens with units during the shunting process, otherwise one cannot construct the duties for units correctly from the output of the model which only gives compositions for trips.

Detecting exceptional shunting codes

Detecting exceptional shunting codes is done by checking if the standard shunting codes given to trips correspond to what is actually happening with the units in the original plan. If this turns out to be not the case, either an exceptional shunting movement was planned or there was an error in the input files. In the latter case feedback with the creator of the original plan might be needed. The obtained exceptional shunting movements are stored in a separate file and fed into the model. A standard shunting code is also assigned to the trip, so the model can choose between transitions allowed by the standard shunting code and the extra transitions from planned exceptional shunting movements.

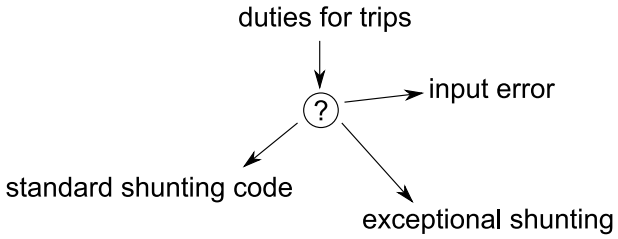


Figure 3.8: Shunting movements can be interpreted as standard shunting movements, exceptional shunting movements or errors in the input file.

The Shunting Index

For the model to be able to work with exceptional shunting movements, some additional information is needed – it needs to know what happens exactly with train units during the shunting, how many units are uncoupled from the train or coupled to the train. One could introduce new shunting codes for every possible exception, but this makes the implementation of the model more complex and inflexible. Therefore, in order to make sense of exceptional shunting movements, a special description of the shunting movements is introduced,

the **shunting index**. The shunting index can also be used to simplify the implementation of the model by giving every possible transition a shunting index. As output, the model gives the compositions of trips, including the shunting index used for transitions. From this the duties can be determined.

The shunting index for trip t is a code that describes what happens to the units in t. The code consists of five symbols, corresponding to the five possible positions a unit can have in a train, since there is a maximum of five train units per train. The first symbol describes what happens to the unit at the front of the train, the second symbol describes what happens to the unit next to the front unit, and so on. Train units that are uncoupled from the train are given the symbol * and units that remain in the train are given a number that indicates the position they have in the successor trip, where again the position is counted from the front of the train to the rear. With splitting and combining, the two separate parts of the train are thought of as one train, where the first arriving or departing part is defined to be the front part of the 'train'. Note that no information about whether the train changes direction is included in the shunting index. See figures 3.9, 3.10 and 3.11 for examples of how the shunting index works.

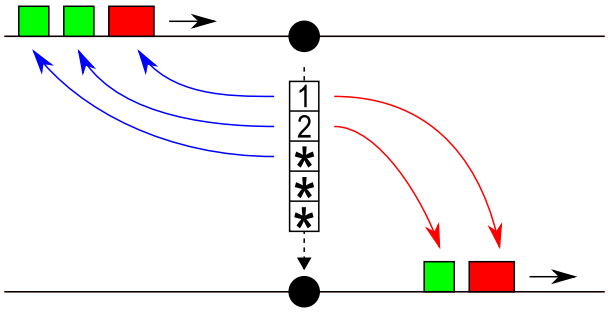


Figure 3.9: An example of the shunting index. The shunting movements in the figure can be described with shunting index **12*****

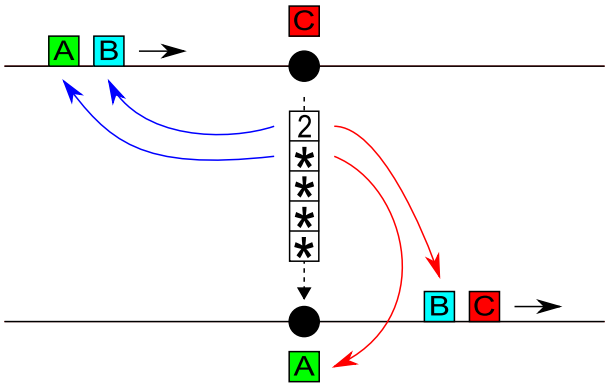


Figure 3.10: An example of the shunting index with exceptional shunting. The shunting movements in the figure can be described with shunting index **2******

Besides the shunting index, also the compositions of the trips involved are needed to determine what happens exactly during a transition. For example, in figure 3.10 it follows from the fact that the successor trip consists of two units that one unit (unit 'C') is coupled to the train.

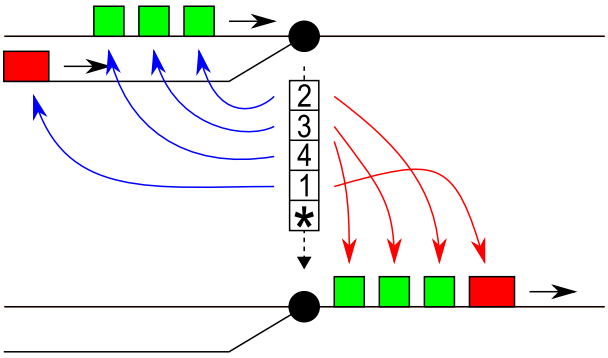


Figure 3.11: An example of the shunting index with combining. The shunting movements in the figure can be described with shunting index **2341***

That this unit is coupled to the front of the train instead of the rear of the train follows since unit 'B' is the second unit in the train, which is described by the shunting index.

The shunting index can also be used to make other parts of the implementation of the composition model easier. In later sections some references to the shunting index will be made.

3.4 Adding Combined Units to the Inventory

When a train has no successor trip after completing a trip, it goes to the shunting yard. Quite often, the train is reused later without any changes to its composition. The composition model does not take this into account, the shunting yard is modelled as a black box where a particular number of units is stored, but no information about stored combined units is saved. This section describes how to extend the model to take this kind of situations into account.

3.4.1 Problem Description

As described above, the model just describes the number of units of a particular type that are stored at the shunting yard. Consider the situation in figure 3.12. At the shunting yard the following trains are stored: two trains with a green unit at the front of the train and a red unit at the rear of the train, one train with two red units and a green unit in the middle and one train with only one green unit. Now suppose that a train is needed from the shunting yard that contains two units. Furthermore, suppose for simplicity that which units are chosen has no influence on the objective function. For the model, the shunting yard contains four green units and four red units, so it cannot distinguish between using a train with two red units, a train with two green units, a train with a green unit at the front and a red unit at the rear, and a train with a red unit at the front and a green unit at the rear. But when looking at the real situation at the shunting yard, it is clear that all options except using a train with a green unit at the front and a red unit at the rear of the train will require extra

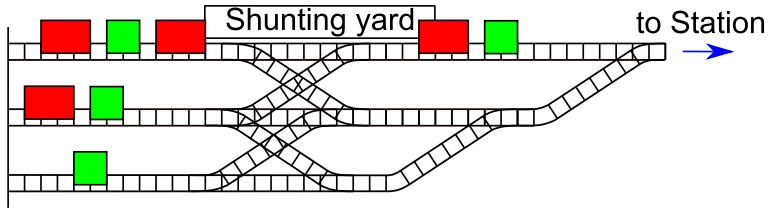


Figure 3.12: A possible arrangement of trains at the shunting yard.

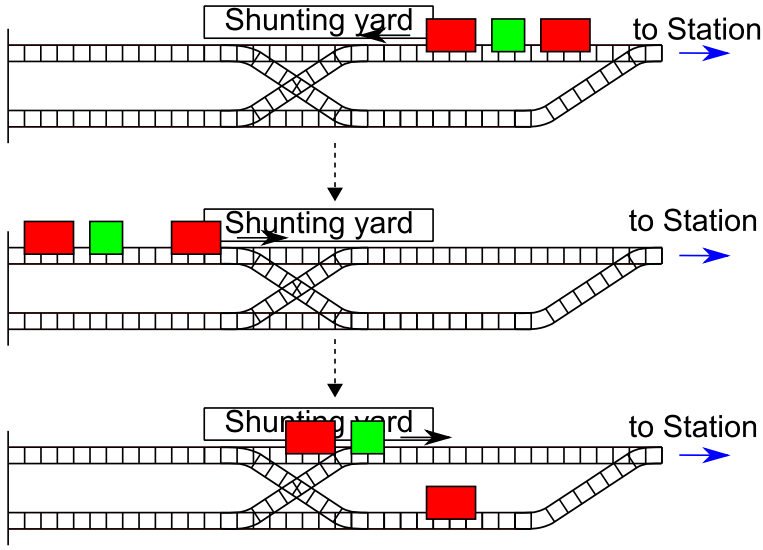


Figure 3.13: Example of a type of shunting movement that sometimes occurs at the shunting yard. Taking into account this kind of shunting movements adds not much value.

shunting movements at the shunting yard which are undesirable. So in this case one would like the model to choose the train with a green unit at the front and a red unit at the rear.

In practice it often occurs that an entire train is first stored at the shunting yard and later reused without changing its composition. Also, when two units are uncoupled from a train they often stay together at the shunting yard. Sometimes it is not possible to keep the units in the same composition at the shunting yard, for example when only a long train is stored at the shunting yard and a short train is needed, or if one of the units is washed or has to be checked or repaired.

3.4.2 Model

Taking into account **everything** that happens at the shunting yard is practically not feasible since a lot of information is not available or available only very shortly before the execution time of the plans. Also, taking into account shunting movements as depicted in figure 3.13 that very rarely occur does not add much value – one would prefer to avoid this type of shunting entirely. Instead, a way to add some more details about what happens at the shunting yard

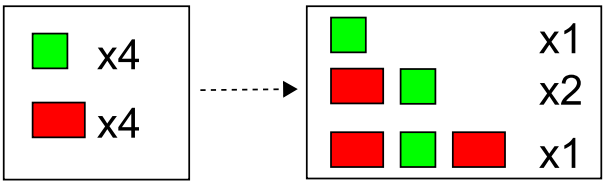


Figure 3.14: The original inventory model is depicted at the left, the new inventory model is depicted at the right.

is to allow one to keep track of what trains are stored at the station instead of only knowing how many units are stored of a particular type.

To model this, the inventory is extended in the model with the so called **combined inventory** which contains compositions of units instead of individual units, see figure 3.14. The inventory of individual units will sometimes be referred to as the **normal inventory** in the following. Since one would also want to keep the flexibility to rearrange units at the shunting yard when needed, an additional rule is introduced. Whenever units arrive at the shunting yard or depart from the shunting yard, a **decision variable** models whether they 'use' the normal inventory or the combined inventory: In the model, arriving units are either 'split up' completely or stored as one complete train which cannot be broken up again at the station, and similarly trains that depart from the shunting yard are either made up entirely of units that were not coupled before or consist of one train that was already stored at the shunting yard. In the example shown in figure 3.13, the shunting movements would be modelled as follows: first the three arriving units are split up, so the inventory consists of three individual units. Then two of the units are combined again to form a new train that departs from the shunting yard.

Adding Constraints

The basic idea to put the concepts introduced above into the composition model is to extend the set $\mathcal{M}$ in the generalized composition model. Starters and finishers may have multiple nodes in the transition graph depending on whether they use the inventory of individual units or the combined inventory. Extra transitions are added which indicate whether the normal inventory or the combined inventory is used when coupling or uncoupling two units.

Denote the set of compositions of units allowed in the combined inventory with $\mathcal{M}^C$, this set may consist for example of all combinations of length two or three of units $m \in \mathcal{M}$. Note that the **order** of the units is important here, a train with one red unit at the front and a green unit at the rear is fundamentally different from a train with one green unit at the front and a red unit at the rear in the combined inventory – revolving a train without uncoupling and later coupling its units is not possible. For units that arrive in the combined inventory the order is determined based on the side of the station they use to arrive at the shunting yard.

To extend the possible compositions $\mathcal{P}_t$, consider a starter $t \in \mathcal{T}_0$. If t has a composition $p \in \mathcal{M}^C$ it could come from the combined inventory. For each such composition an element

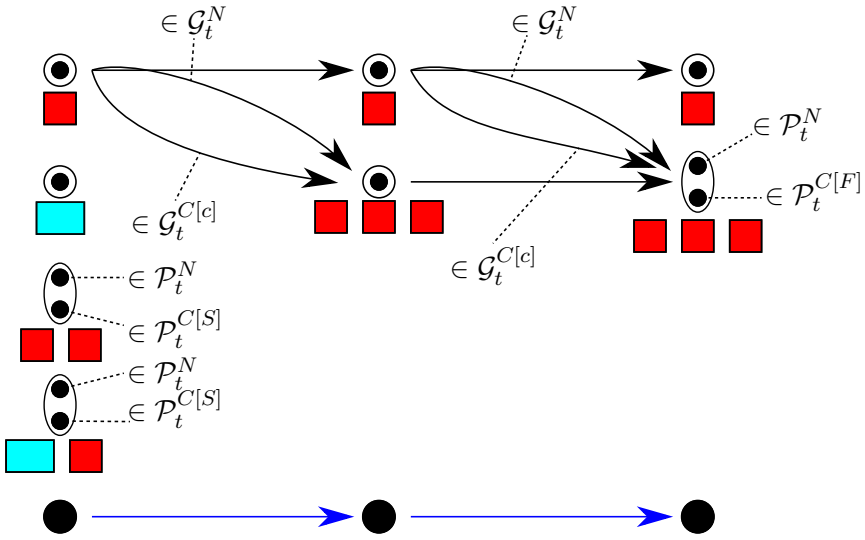


Figure 3.15: An example of how the combined inventory is incorporated in the model.

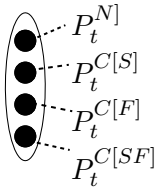


Figure 3.16: Trips that are both starters and finishers have at most four nodes for the same composition in the transition graph.

is added to $\mathcal{P}_t$ to indicate this possibility and the set of elements added in this way is called $\mathcal{P}_t^{C[S]}$. Similarly, consider finisher $t \in \mathcal{T}_\infty$. If t has a composition $p \in \mathcal{M}^C$ it could be ridden to the combined inventory. For each such compositions an element is added to $\mathcal{P}_t$ to indicate this possibility, and the set of elements added in this way is called $\mathcal{P}_t^{C[F]}$. If trip t is a starter as well as a finisher, so it has no successor and no predecessor trips, $t \in \mathcal{T}_0 \cap \mathcal{T}_\infty$, then for compositions $p \in \mathcal{M}^C$ of this trip it is possible that they come from the combined inventory and are returned to the combined inventory after the trip. Also for these possibilities elements are added to $\mathcal{P}_t$ and they are denoted by $\mathcal{P}_t^{C[SF]}$. Defining $\mathcal{P}_t^N = \mathcal{P}_t \setminus \{\mathcal{P}_t^{C[S]} \cup \mathcal{P}_t^{C[F]} \cup \mathcal{P}_t^{C[SF]}\}$ we have that any composition in the extended set $\mathcal{P}_t$ is in exactly one of the sets $\mathcal{P}_t^N, \mathcal{P}_t^{C[S]}, \mathcal{P}_t^{C[F]}$ or $\mathcal{P}_t^{C[SF]}$. Since most trips are neither a starter nor a finisher, for most trips it holds that $\mathcal{P}_t = \mathcal{P}^N$.

The above describes how trips with no predecessor trips or no successor trips can come from the combined inventory or can be returned to the combined inventory. Also units that are uncoupled or coupled to a train during transitions $\mathcal{G}_t$ between trips can use the combined inventory. If the group of units uncoupled from a train during a certain transition is in $\mathcal{M}^C$ then it is possible that these units go to the combined inventory, this possibility is modelled by adding an element to $\mathcal{G}_t$. Possible elements that could be added in this way are in the set $\mathcal{G}_t^{C[u]}$. Similarly, if a group of units coupled to a train during a transition is in $\mathcal{M}^C$ then it is possible that these units came from the combined inventory, which is modelled by adding an

element to $\mathcal{G}_t$. Elements that could be added in this way are in the set $\mathcal{G}_t^{C[c]}$. For trips where the units that are uncoupled and the units that are coupled to the train are both in $\mathcal{M}^C$ a transition is added to $\mathcal{G}_t$ and also stored in $\mathcal{G}_t^{C[cu]}$, although this type of transition normally does not occur. All transitions that do not use the combined inventory are in $\mathcal{G}_t^N$, so we have a partition of $\mathcal{G}_t$ in $\mathcal{G}_t^N, \mathcal{G}_t^{C[u]}, \mathcal{G}_t^{U[c]}$ and $\mathcal{G}_t^{C[cu]}$. See figures 3.15 and 3.16 for an example of how the extended versions of $\mathcal{P}_t$ and $\mathcal{G}_t$ are implemented in the model.

The constants $c_m(a), u_m(a), c_m^S(e)$ and $u_m^F(e)$ need to be defined for the compositions allowed in the combined inventory $m \in \mathcal{M}^C$, the extended transitions $\mathcal{G}_t$ and the extended compositions $\mathcal{P}_t$:

$$c_m(a) = \begin{cases} \#\text{ of units of type } m \text{ coupled to } \sigma(t) & \text{for all } m \in \mathcal{M}, a \in \mathcal{G}_t^N \\ 0 & \text{for all } m \in \mathcal{M}, a \in \mathcal{G}_t \setminus \mathcal{G}_t^N \\ 0 & \text{for all } m \in \mathcal{M}^C, a \in \mathcal{G}_t^N \cup \mathcal{G}_t^{C[u]} \\ 1 & \text{for all } m \in \mathcal{M}^C, a \in \mathcal{G}_t^{C[c]} \cup \mathcal{G}_t^{C[cu]} \end{cases} \tag{3.16}$$

$$u_m(a) = \begin{cases} \#\text{ of units of type } m \text{ uncoupled from } t & \text{for all } m \in \mathcal{M}, a \in \mathcal{G}_t^N \\ 0 & \text{for all } m \in \mathcal{M}, a \in \mathcal{G}_t \setminus \mathcal{G}_t^N \\ 0 & \text{for all } m \in \mathcal{M}^C, a \in \mathcal{G}_t^N \cup \mathcal{G}_t^{C[c]} \\ 1 & \text{for all } m \in \mathcal{M}^C, a \in \mathcal{G}_t^{C[u]} \cup \mathcal{G}_t^{C[cu]} \end{cases} \tag{3.17}$$

$$c_m^S(e) = \begin{cases} \#\text{ of units of type } m \text{ in } t & \text{for all } m \in \mathcal{M}, e \in \mathcal{P}_t^N, \\ 0 & \text{for all } m \in \mathcal{M}, e \in \mathcal{P}_t \setminus \mathcal{P}_t^N \\ 0 & \text{for all } m \in \mathcal{M}^C, e \in \mathcal{P}_t^N \cup \mathcal{P}_t^{C[F]} \\ 1 & \text{for all } m \in \mathcal{M}^C, e \in \mathcal{P}_t^{C[S]} \cup \mathcal{P}_t^{C[SF]} \end{cases} \tag{3.18}$$

$$u_m^F(e) = \begin{cases} \#\text{ of units of type } m \text{ in } t & \text{for all } m \in \mathcal{M}, e \in \mathcal{P}_t^N, \\ 0 & \text{for all } m \in \mathcal{M}, e \in \mathcal{P}_t \setminus \mathcal{P}_t^N \\ 0 & \text{for all } m \in \mathcal{M}^C, e \in \mathcal{P}_t^N \cup \mathcal{P}_t^{C[S]} \\ 1 & \text{for all } m \in \mathcal{M}^C, e \in \mathcal{P}_t^{C[F]} \cup \mathcal{P}_t^{C[SF]} \end{cases} \tag{3.19}$$

After extending the composition variables $\mathcal{P}_t$ and the transition variables $\mathcal{G}_t$ and defining the corresponding variables $c_m(a), u_m(a), c_m^S(e)$ and $u_m^F(e)$, most constraints remain the same in the generalized composition model with combined inventory included.

Coupling and uncoupling are described by:

$$C_{\sigma(t),m} = \sum_{a \in \mathcal{G}_t} c_m(a) \cdot Z_{t,a} \qquad \text{for all } t \in \mathcal{T} \setminus \mathcal{T}_\infty, m \in \mathcal{M} \cup \mathcal{M}^C \tag{3.20}$$

$$U_{t,m} = \sum_{a \in \mathcal{G}_t} u_m(a) \cdot Z_{t,a} \qquad \text{for all } t \in \mathcal{T} \setminus \mathcal{T}_\infty, m \in \mathcal{M} \cup \mathcal{M}^C \tag{3.21}$$

$$C_{t,m} = \sum_{e \in \mathcal{P}_t} c^S_m(e) \cdot X_{t,e} \qquad \text{for all } t \in \mathcal{T}_0, m \in \mathcal{M} \cup \mathcal{M}^C \tag{3.22}$$

$$U_{t,m} = \sum_{e \in \mathcal{P}_t} u^F_m(e) \cdot X_{t,e} \qquad \text{for all } t \in \mathcal{T}_\infty, m \in \mathcal{M} \cup \mathcal{M}^C \tag{3.23}$$

Inventory constraints also remain roughly the same. Variables $i^0_{s,m}$ and $i^\infty_{s,m}$ need to be defined for $m \in \mathcal{M}^C$ in order to describe the initial and final inventory of combined units at the stations.

$$\begin{aligned} I_{t,m} = I^0_{s_d(t),m} \quad & - \sum_{\substack{t' \in \mathcal{T}: s_d(t') = s_d(t), \\ \tau_d(t') \le \tau_d(t)}} C_{t',m} \\ & + \sum_{\substack{t' \in \mathcal{T}: s_a(t') = s_d(t), \\ \tau_d(t') \le \tau_d(t) - \rho(t')}} U_{t',m} \qquad \text{for all } t \in \mathcal{T}, m \in \mathcal{M} \cup \mathcal{M}^C \end{aligned} \tag{3.24}$$

$$I^\infty_{s,m} = I^0_{s,m} - \sum_{t \in \mathcal{T}: s_d(t) = s} C_{t,m} + \sum_{t \in \mathcal{T}: s_a(t) = s} U_{t,m} \qquad \text{for all } s \in \mathcal{S}, m \in \mathcal{M} \cup \mathcal{M}^C \tag{3.25}$$

$$I^0_{s,m} = i^0_{s,m} \qquad \text{for all } s \in \mathcal{S}, m \in \mathcal{M} \cup \mathcal{M}^C \tag{3.26}$$

$$I^\infty_{s,m} = i^\infty_{s,m} \qquad \text{for all } s \in \mathcal{S}, m \in \mathcal{M} \cup \mathcal{M}^C \tag{3.27}$$

Ensuring Integrality

Because there sometimes is more than one transition possible between two given compositions of trips, extra decision variables need to be added if the transition variables are allowed to be fractional. Consider for example the following situation. Suppose the shunting yard at a particular station is empty and the following events occur successively:

1. Two units of type **A**, denoted by **AA**, are uncoupled from a train and arrive at the station.
2. A unit of type **A** is coupled to a train and departs from the station.
3. A unit of type **A** is uncoupled from a train and arrives at the station.

4. Two units **AA** are coupled to a train and depart from the station.

Clearly, in this case the first arriving finisher train **AA** should be split up because some time later one unit **A** is required from the shunting yard. So the combined inventory remains empty during this example. But if the integrality of the transitions is not required in the model, then problems can occur. Suppose that there is a bonus for using the combined inventory in the objective function. Now the model would see the following scenario as the optimal solution to the way the shunting movements are handled:

1. Two units **AA** are uncoupled from a train and arrive at the station. $\frac{1}{2}$ of them go to the combined inventory and $\frac{1}{2}$ of them go to the normal inventory, so now one unit **A** is stored at the normal inventory and $\frac{1}{2}$ composition **AA** is stored at the combined inventory.
2. A unit of type **A** is coupled to a train and departs from the station. This is possible since there was one unit **A** stored at the normal inventory.
3. A unit of type **A** is uncoupled from a train and arrives at the station.
4. Two units **AA** are coupled to a train and depart from the station. $\frac{1}{2}$ of them come from the last arriving unit which was stored in the normal inventory, and $\frac{1}{2}$ of them come from the combined inventory.

The scenario above is clearly practically infeasible as two units cannot be 'half' uncoupled. But there is no practically feasible solution with the same objective function. What needs to be added are decision variables that keep the transition variables $Z_{t,a}$ integral in the solution, or equivalently require that the relevant $Z_{t,a}$ variables themselves are integral.

So consider the decision variables $z^{C[c]}(t) \in \{0,1\}$ for all trips where $G_t^{C[c]} \neq \emptyset$, where $z^{C[c]}(t) = 1$ if units coupled to the train during this transition come from the combined inventory and $z^{C[c]}(t) = 0$ otherwise. Also needed are variables $z^{C[u]}(t) \in \{0,1\}$ for all trips where $G_t^{C[u]} \neq \emptyset$, where $z^{C[u]}(t) = 1$ if units uncoupled from the train during this transition are stored in the combined inventory. Now the following constraints ensure the integrality of the transition variables $Z_{t,a}$:

$$\sum_{a \in \mathcal{G}_t^{C[c]}} Z_{t,a} + \sum_{a \in \mathcal{G}_t^{C[cu]}} Z_{t,a} = z_t^{C[c]} \tag{3.28}$$

$$\sum_{a \in \mathcal{G}_t^{C[u]}} Z_{t,a} + \sum_{a \in \mathcal{G}_t^{C[cu]}} Z_{t,a} = z_t^{C[u]} \tag{3.29}$$

To see why this is sufficient, note that the equations that link the compositions to the transitions, equations 3.2 and 3.3, imply that

$$\sum_{a \in \mathcal{G}_t^N} Z_{t,a} + \sum_{a \in \mathcal{G}_t^{C[c]}} Z_{t,a} + \sum_{a \in \mathcal{G}_t^{C[u]}} Z_{t,a} + \sum_{a \in \mathcal{G}_t^{C[cu]}} Z_{t,a} = 1$$

- In case $z^{C[c]}(t) = 0$ and $z^{C[u]}(t) = 0$ this implies that $\sum_{a \in \mathcal{G}_t^N} Z_{t,a} = 1$.
- In case $z^{C[c]}(t) = 1$ and $z^{C[u]}(t) = 0$ this implies that $\sum_{a \in \mathcal{G}_t^{C[c]}} Z_{t,a} = 1$.
- In case $z^{C[c]}(t) = 0$ and $z^{C[u]}(t) = 1$ this implies that $\sum_{a \in \mathcal{G}_t^{C[u]}} Z_{t,a} = 1$.
- In case $z^{C[c]}(t) = 1$ and $z^{C[u]}(t) = 1$ this implies that $\sum_{a \in \mathcal{G}_t^{C[cu]}} Z_{t,a} = 1$.

The sums in the four cases described here make sure that there is only one possible transition between two compositions, which implies the integrality of $Z_{t,a}$.

An alternative to adding these decision variables might be to force the combined inventory to be integral. But the decision variables above make intuitively more clear that the transition variables must be integral and are possibly more robust when future changes are incorporated.

Objective Function

The combined inventory was introduced in the model to be able to reduce the amount of shunting movements at the station. Changes must be made in the objective function to be able to do this.

Simply giving a bonus to trips that use the combined inventory does not suffice. This follows since then for trips where one unit is uncoupled normally, the model might attempt to uncouple two units to be able to obtain a bonus for getting units into the combined inventory, and this does not make sense in this case at all. Instead the following rule is introduced:

> *If during a certain shunting movement units could come from the combined inventory or could go to the combined inventory but they do not, they are penalized.*

So when two units are uncoupled from a train and go to the normal inventory they are penalized. This makes sense since in this case extra shunting movements (splitting up the units) are required.

In order to describe the penalties, some additional notation is introduced. The set $\mathcal{P}_t^{C[S]'}$ is defined to be the set of extended compositions $\mathcal{P}_t^N$ where there is a similar extended

composition which comes from the combined inventory, and similarly sets $\mathcal{P}_t^{C[F]'}, \mathcal{G}_t^{C[u]'}$ and $\mathcal{G}_t^{C[c]'}$ are defined:

$$\mathcal{P}_t^{C[S]'} := \{e \in \mathcal{P}_t^N : \{x \in \mathcal{P}_t^{C[S]} \cup \mathcal{P}_t^{C[SF]} : p(x) = p(e)\} \neq \emptyset\} \tag{3.30}$$

$$\mathcal{P}_t^{C[F]'} := \{e \in \mathcal{P}_t^N : \{x \in \mathcal{P}_t^{C[F]} \cup \mathcal{P}_t^{C[SF]} : p(x) = p(e)\} \neq \emptyset\} \tag{3.31}$$

$$\mathcal{G}_t^{C[u]'} := \{a \in \mathcal{G}_t^N : \{x \in \mathcal{G}_t^{C[u]} \cup \mathcal{G}_t^{C[cu]} : (p(x), p'(x)) = (p(a), p'(a))\} \neq \emptyset\} \tag{3.32}$$

$$\mathcal{G}_t^{C[c]'} := \{a \in \mathcal{G}_t^N : \{x \in \mathcal{G}_t^{C[c]} \cup \mathcal{G}_t^{C[cu]} : (p(x), p'(x)) = (p(a), p'(a))\} \neq \emptyset\} \tag{3.33}$$

Now the penalties can be described. For this the number of shunting movements that need to be penalized needs to be determined. The number of starters that need to be penalized is:

$$\sum_{t \in \mathcal{T}_0} \sum_{e \in \mathcal{P}_t^{C[S]'}} X_{t,e} \tag{3.34}$$

Similarly, the number of finishers that need to be penalized is:

$$\sum_{t \in \mathcal{T}_\infty} \sum_{e \in \mathcal{P}_t^{C[F]'}} X_{t,e} \tag{3.35}$$

The number of shunting movements where train units are uncoupled from a train and could go to the combined inventory but do not are given by the constraints:

$$\sum_{t \in \mathcal{T} \setminus \mathcal{T}_\infty} \sum_{a \in \mathcal{G}_t^{C[u]'}} Z_{t,a} \tag{3.36}$$

And similarly the number of shunting movements where train units are coupled to a train and could come from the combined inventory but do not are given by the constraints:

$$\sum_{t \in \mathcal{T} \setminus \mathcal{T}_\infty} \sum_{a \in \mathcal{G}_t^{C[c]'}} Z_{t,a} \tag{3.37}$$

3.5 Fast Shunting Movements

Sometimes train units that arrive at the station or that are uncoupled from a train remain at the station and are reused very quickly where they would usually be sent to the shunting yard. This section explains what happens and how to incorporate this into the model.

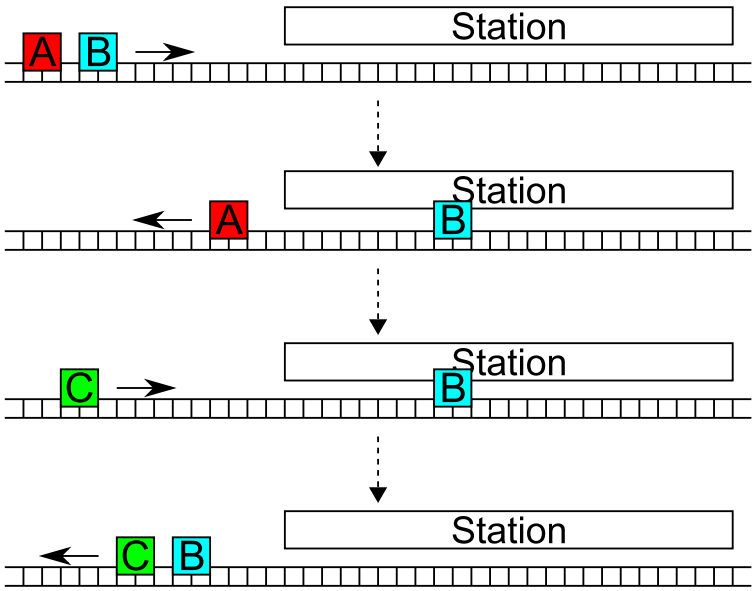


Figure 3.17: A train unit is uncoupled from a train, waits a while at the station and is coupled to another train

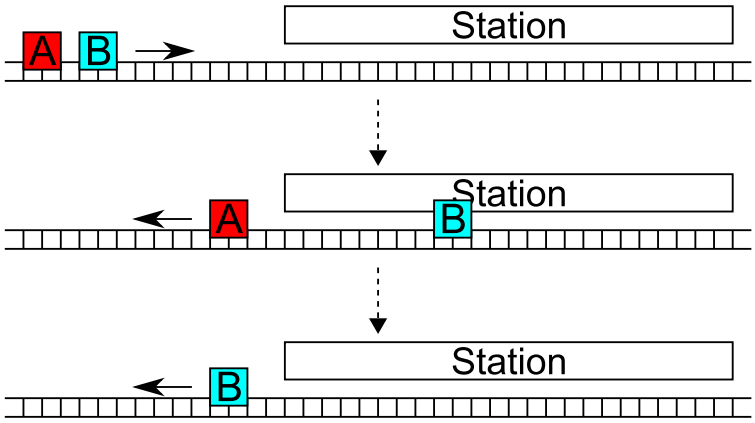


Figure 3.18: A train unit is uncoupled from a train, waits a while and departs as a starter train.

3.5.1 Problem Description

Figures 3.17, 3.18 and 3.19 give three situations where train units that would normally go to the shunting yard remain at the station and are coupled to an arriving train or depart as a new train. In this report these types of shunting movements are called shunting movements with a **lock**: units uncoupled from one trip (where 'uncoupling' includes finisher trains) are coupled to another trip (where 'coupling' includes starter trains), so there is a lock between the two trips.

Note that the situations in figures 3.18 and in 3.19 look like combining and splitting. The difference is that in these situations one would usually not use combining and splitting.

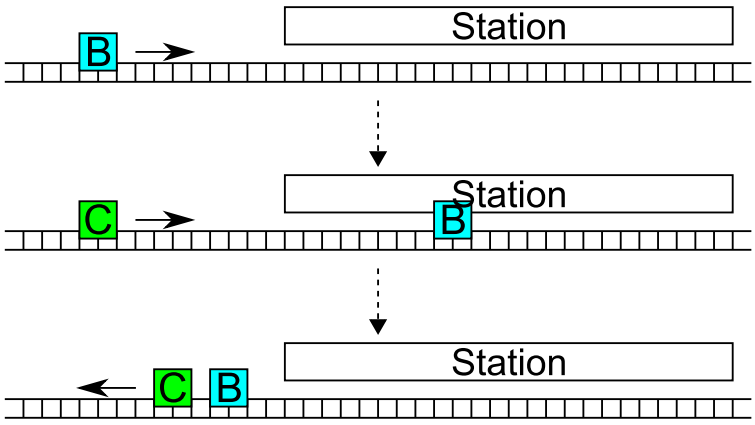


Figure 3.19: A train unit arrives at the station, waits a while and is coupled to an incoming train.

Instead the arriving train units would be sent to the shunting yard and other units from the shunting yard would be coupled to the departing trains. Although that might seem like a more difficult way to handle the situation, it is more reliable. For example, when the arriving train is delayed there is no influence on the departing train in this case. But when the inventory is empty this option is not available, in that case a lock is needed. While combining and splitting normally don't occur at end stations, the fast shunting movements usually occur at end stations. Also, in the model one would like to allow the possibility **not** to use a lock. So there are four conceptual differences between locking and combining / splitting:

- Combining or splitting is usually used when it is convenient, locking is used when no other option is possible.
- Combining and splitting usually don't occur at end stations, locking usually does occur at end stations.
- Locking should be a choice in the model, one should be able **not** to use a lock when possible.
- With locking it is possible to link units that are uncoupled from one train to units that are coupled to another train. This cannot be modelled with combining and splitting without difficulties.

In some situations the desirability of obtaining units from the inventory instead of using a lock might not be that clear. In the paragraph above it was pointed out that a lock introduces dependencies, a delayed train might cause another delayed train because units must be transferred from the first train to the second train. But the delayed train may already cause the other train to be delayed since they may use the same track. In this case it doesn't matter whether a lock would be used or the inventory. This type of situations is important to consider when including locking in the objective function.

Depending on the position of the shunting yard relative to the station and to the entering point of the station, it is possible that train units that are coupled **without** a lock to a train

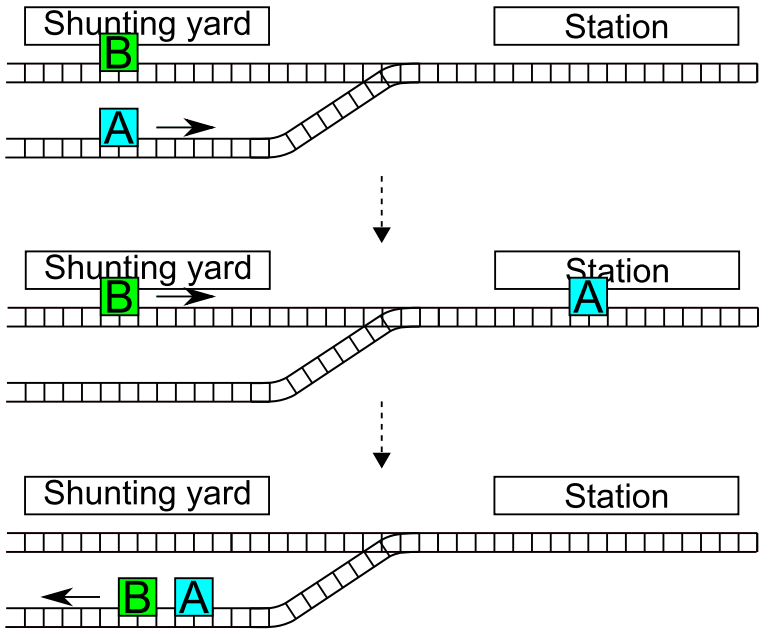


Figure 3.20: Coupling without lock. Here the trip with only unit A is given shunting code **abK**.

are coupled to the other side of the train than units that are coupled **with** a lock to a train. For example, see figures 3.20 and 3.21. In figure 3.20 train units are coupled to a train from the shunting yard without using a lock. Due to the location of the shunting yard, the units are coupled to the rear of the train. So in this case the trip after which the units are coupled to the train is given shunting code **abK**. In figure 3.21 train units are coupled to a train with a lock. In this case the units are coupled to the front of the train and the trip after which the units are coupled to the train is given shunting code **Kab**. So a lock might require an alternative shunting code.

3.5.2 Model

In order to model the fast shunting movements or locks described above, one needs to make additional links between trips and specify under which conditions these links are used. To accomplish this, elements need to be added to the extended compositions and the extended transitions and some new constraints will be added to the generalized composition model. One also needs to take into account that sometimes links require an alternative shunting code as described in the previous section and figures 3.20 and 3.21.

Note that there are three cases to consider that are modelled slightly differently:

- Train units are uncoupled from a train, remain at the station for a while and continue as a starter train. Figure 3.18 gives an example of this.

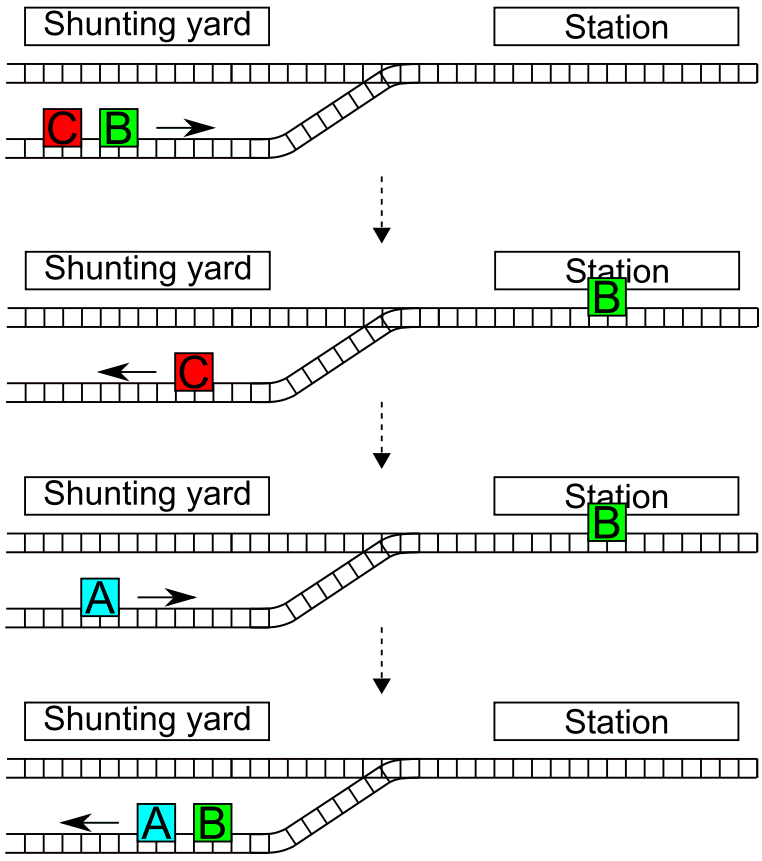


Figure 3.21: Coupling with lock. Here the trip with only unit **A** is given shunting code **Kab**.

- Train units are uncoupled from a train, remain at the station for a while and are coupled to an arriving train. Figure 3.17 gives an example of this.
- A finisher train arrives at the station, remains at the station for a while and is coupled to an arriving train. Figure 3.19 gives an example of this.

Adding Constraints

Incorporating the fast shunting movements in the generalized composition model is not very different from incorporating the combined inventory addition. New elements need to be added to the compositions $\mathcal{P}_t$ and to the transitions $\mathcal{G}_t$ and some new connections need to be made between trips. Also one needs to add extra decision variables to make sure that the optimal solution is integral.

In all three cases there is a **lock** between two trips: everything that is uncoupled from trip t_1 (including 'uncoupling' the entire train in case it is a finisher) is coupled to trip t_2 (including 'coupling' where the train departing on trip t_2 is a starter). Let $\mathcal{L}$ be the collection of all locks, where $(t_1, t_2) \in \mathcal{L}$ if there is a lock between t_1 and t_2. Let $(t_1, t_2) \in \mathcal{L}$. If t_1 is a finisher, extended compositions are added to $\mathcal{P}_{t_1}$ that describe the lock possibility. Denote these compositions with $\mathcal{P}^L_{t_1}$. If t_1 is not a finisher, extended transitions are added to $\mathcal{G}_{t_1}$ for all transitions where a lock would be possible. This will be all the transitions where units are uncoupled from the train and the right type of shunting is used, so for example only uncoupling from the front or from the rear will be allowed. Denote these transitions with $\mathcal{G}^L_{t_1}$. If t_2 is a starter, extended compositions $\mathcal{P}^L_{t_2}$ are added to $\mathcal{P}_{t_2}$ and otherwise extended transitions $\mathcal{G}^L_{t_2}$ are added to $\mathcal{G}_{t_2}$ to describe the possible lock. Again, only transitions with the right type of shunting are included.

The values $c_m(a), u_m(a), c^S_m(e)$ and $u^F_m(e)$ are set to zero for $a \in \mathcal{G}^L_t$ and $e \in \mathcal{P}^L_t$ which makes the constraints for coupling / uncoupling and for the inventory the same as in the generalized composition model, even with the additions made to $\mathcal{P}_t$ and $\mathcal{G}_t$. New binary values $c^L_p(a), u^L_p(a), c^{L[S]}_p(e)$ and $u^{L[F]}_p(e)$ are introduced for locks that describe the **ordered** composition of what is coupled (again including the case that the train is actually a starter) or uncoupled (including the case that the train is actually a finisher). For example, $c^L_p(a)$ indicates whether ordered composition p is coupled to the train during extended transition a. The ordering of the train is determined by the side of the station the train arrives at, this is similar to what happens for the combined inventory.

Now the most important additional constraints for locks can be introduced, everything that is uncoupled from t_1 is coupled to t_2:

$$\sum_{a \in \mathcal{G}^L_{t_1}} u^L_p(a) \cdot Z_{t_1,a} = \sum_{\substack{t \in \mathcal{T}: \sigma(t) = t_2: \\ a \in \mathcal{G}^L_t}} c^L_p(a) \cdot Z_{t,a} \quad \text{for all } (t_1, t_2) \in \mathcal{L}, t_1 \notin \mathcal{T}_\infty, t_2 \notin \mathcal{T}_0. \quad (3.38)$$

$$\sum_{e \in \mathcal{P}^L_{t_1}} u^{L[F]}_p(e) \cdot X_{t_1,e} = \sum_{\substack{t \in \mathcal{T}: \sigma(t) = t_2: \\ a \in \mathcal{G}^L_t}} c^L_p(a) \cdot Z_{t,a} \quad \text{for all } (t_1, t_2) \in \mathcal{L}, t_1 \in \mathcal{T}_\infty, t_2 \notin \mathcal{T}_0. \quad (3.39)$$

$$\sum_{a \in \mathcal{G}_{t_1}} u^L_p(a) \cdot Z_{t,a} = \sum_{e \in \mathcal{P}^L_{t_2}} c^{L[S]}_p(e) \cdot X_{t_2,e} \quad \text{for all } (t_1, t_2) \in \mathcal{L}, t_1 \notin \mathcal{T}_\infty, t_2 \in \mathcal{T}_0. \quad (3.40)$$

Where the constraints above hold for all ordered compositions p of units.

Ensuring Integrality

Because the lock addition above can introduce parallel transitions between a pair of compositions, the integrality of the compositions might not be enough to ensure the integrality of the transitions, which can result in an infeasible solution. Consider the following example. A station has one unit of type **A** stored at the shunting yard. Now the following two events take place successively at the station:

1. Two units of type **A** are uncoupled from the front of an arriving train.
2. A few minutes later two units of type **A** are coupled to the rear of a departing train.

In this case the only feasible solution of this situation is clearly to allow a lock between the two trips. But suppose that using a lock is expensive, more expensive than using the inventory. This is a reasonable assumption since a lock is quite complicated and introduces dependencies: the later departing train will have to wait for the earlier arriving train if the latter is delayed. If there are no integrality constraints on the transitions, the following solution would be optimal:

1. Two units of type **A** are uncoupled from the front of an arriving train. One of them goes to the inventory, which means that the inventory is increased with one unit of type **A**. But because of the reallocation time it will only become available some time later. The other unit is locked with the train that will depart later, that is, it remains at the station and will be coupled with the departing train.
2. A few minutes later two units of type **A** are coupled to the rear of a departing train. One of the units comes from the inventory (note that this is not the unit that was sent to the inventory but the unit that was already stored at the inventory) and one of the units is locked from the earlier arriving train.

Although this scenario is theoretically feasible, it is a bad solution: there is no reason why not to let **both** the two units be locked to the next train instead of just one, the latter only requires more shunting movements. The solution is to introduce extra decision variables for locks of the type where units uncoupled from one train are coupled to another train, excluding starters and finishers. For such pairs $(t_1, t_2) \in \mathcal{L}$ let $z^L(t_1, t_2) \in \{0, 1\}$ be equal to 1 if a lock is used and 0 otherwise. The following constraints link the transitions to this decision variable:

$$\sum_{a \in \mathcal{G}^L_{t_1}} Z_{t_1,a} = z^L(t_1, t_2) \qquad \text{for all } (t_1, t_2) \in \mathcal{L}, t_1 \notin \mathcal{T}_\infty, t_2 \notin \mathcal{T}_0. \tag{3.41}$$

$$\sum_{\substack{t \in \mathcal{T}: \sigma(t) = t_2, \\ a \in \mathcal{G}^L_t}} Z_{t,a} = z^L(t_1, t_2) \qquad \text{for all } (t_1, t_2) \in \mathcal{L}, t_1 \notin \mathcal{T}_\infty, t_2 \notin \mathcal{T}_0. \tag{3.42}$$

When a lock between two trips contains only one unit, normally there are no problems with the integrality of the transitions even without an additional decision variable. As the inventory contains an integral number of units, either a unit can be obtained from the shunting yard or it cannot, but there is no scenario where it would be desirable or needed to obtain $\frac{1}{2}$ unit from the inventory when one whole unit is available. So if a restriction that at most one unit can be locked is included, the decision variable introduced here is not needed. This is an important point to consider, since in practice it turns out that locking one unit happens in most cases.

Objective Function

In the objective function a penalty or bonus can be given for using locks. As said before, it can be difficult to determine whether a lock is desirable or not. Because locks introduce dependencies between trains they can cause extra delays, but sometimes these trains already depend on each other since they use the same track. Locks induce complex shunting movements, but they also save shunting movements at the shunting yard. So in practice it is not always clear whether locks should be avoided or not. In this report though locks are penalized, they are seen as difficult shunting movements that should in general be avoided, only when the inventory is limited they should be used.

The following term calculates the total number of locks used by summing over all trips where units are uncoupled from:

$$\sum_{(t_1,t_2) \in \mathcal{L}:\, t_1 \in \mathcal{T}_\infty} \sum_{e \in \mathcal{P}^L_{t_1}} X_{t_1,e} + \sum_{(t_1,t_2) \in \mathcal{L}:\, t_1 \notin \mathcal{T}_\infty} \sum_{a \in \mathcal{G}^L_{t_1}} Z_{t_1,a} \tag{3.43}$$

3.5.3 Implementation

Implementing the fast shunting movements poses some problems:

- Detecting locks from an original plan.
- Taking into account unusual shunting codes; as described before a lock is often accompanied by a different shunting code than is normally used for a similar trip.
- Including the detected locks in the model.

An original plan does not describe concepts such as locks or even combining and splitting and contains no information about what happens at the shunting yard. Therefore a script is used to determine what kind of shunting occurs. As the concept of a lock does not differ very much from combining and splitting, for some shunting movements it is difficult to say what the best way is to describe it. The following rule is currently used:

> *If the shunting movement is completed within 20 minutes, it is interpreted as combining or splitting. If it is completed in between 20 and 30 minutes it is interpreted as a fast shunting movement or lock. Otherwise, the relevant units are interpreted to go to the shunting yard.*

The reason for this is that 'normal' combining and splitting is usually not done at end stations. Therefore the trains are part of an ongoing train line which should not wait too long at a station. When a train would stay more then 30 minutes at the station it would be possible to send the train to the shunting yard and retrieve the train from the shunting yard when using the reallocation time of 30 minutes, keeping the tracks at the station clear for other trains to depart. The remaining trains are then good candidates to be modelled by locks. Note that the rule introduced here should not be interpreted as a 'fundamental truth'. The 20 and 30 minute rules seem to work well in practice though. A further addition to the model would be to use the actual shunting plans which give more information.

Trips can be assigned standard shunting codes based on departure and arrival stations, using local information about the stations such as the location of the shunting yard. Trips that are locked might need an alternative shunting code. These alternative shunting codes need to be given to the model. The locks are also given to the model in a separate file with relevant trips and information about which shunting code is needed when using this lock. Then the model creates extra transitions for locks using only this shunting code.

3.6 The Continuity Constraint

As described before, the continuity constraint states that for every train line from its starting station to its ending station there should be at least one unit that is included in every trip. In the previous chapter a way to include this constrained was introduced, which works under the assumption that there is no combining and splitting of trains and also that there are no exceptional shunting movements. In this section a flow model is introduced that can be included in the model and that does not need these assumptions.

3.6.1 Model

The continuity constraint can be modelled as number of small flow problems, one for each relevant train. Consider a train with trips $t_1, t_2, ..., t_k$. Given the set of compositions and transitions determined by the model a graph is constructed in the following way: for every trip five nodes are added corresponding to positions of units in the train, counted from the front of the train. The number five is chosen since trains can consist of at most five units. If the train unit with position p_1 in trip t_i goes to position p_2 in trip t_{i+1} then an arc is added between the two corresponding nodes. Now, if it is possible to reach a node in trip t_k from a node in trip t_1 then the continuity constraint is satisfied, and this corresponds to a flow greater than zero between trip t_1 and trip t_k. See figures 3.22 and 3.23 for two examples of how this flow model works.

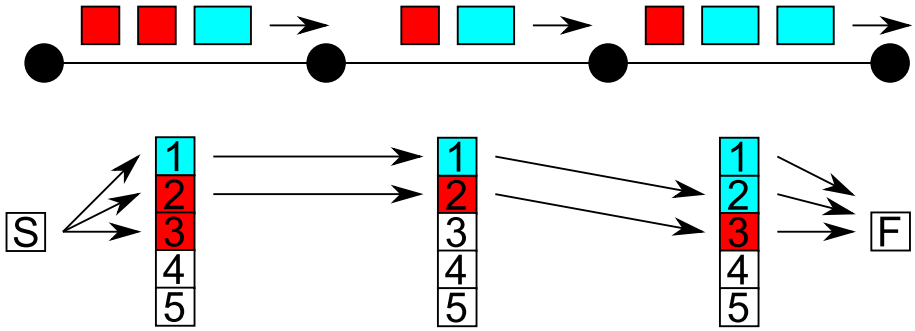


Figure 3.22: An example of the flow model. Since there are two units that are in all three trips, there is a flow of two possible between the first trip and the last trip.

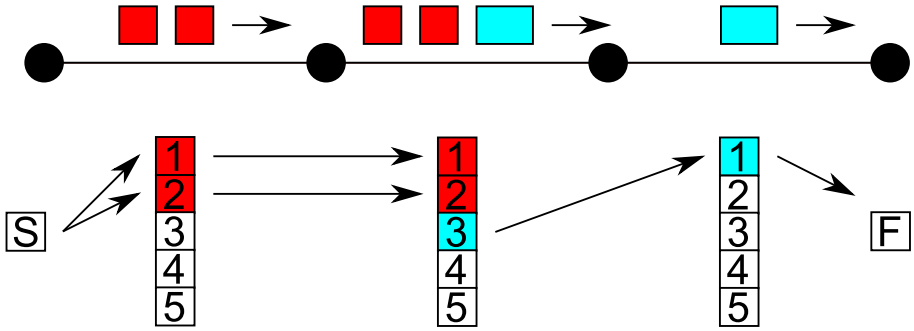


Figure 3.23: An example of the flow model. In this case no units of the first trip are in the last trip, so there is no flow between the first trip and the last trip possible.

For the more formal definition of the flow model to describe the continuity constraint, let $\mathcal{G}_t(p_1, p_2)$ denote the set of transitions where the unit with position p_1 in trip t goes to position p_2 in trip $\sigma(t)$. The variable D_{t,p_1,p_2} describes the amount of flow between the node corresponding to the unit with position p_1 in trip t and the node corresponding to the unit with position p_2 in trip $\sigma(t)$.

There can only be a flow between two nodes if a proper transition is selected:

$$D_{t,p_1,p_2} \leq \sum_{a \in \mathcal{G}_t(p_1,p_2)} Z_{t,a} \qquad \text{for all } t \in \mathcal{T} \setminus \mathcal{T}_\infty, p_1, p_2 \in \{1,..,5\}. \tag{3.44}$$

The amount of flow into a node must be equal to the amount of flow out of a node:

$$\sum_{p_1=1}^{5} D_{t,p_1,p} = \sum_{p_2=1}^{5} D_{\sigma(t),p,p_2} \qquad \text{for all } t \in \mathcal{T} \setminus \mathcal{T}_\infty, p \in \{1,..,5\}. \tag{3.45}$$

And the total amount of flow should be at least one:

$$\sum_{p_1=1}^{5} \sum_{p_2=1}^{5} D_{t,p_1,p_2} \geq 1 \qquad \text{for all } t \in \mathcal{T}_0. \tag{3.46}$$

3.6.2 Implementation

In order to find which train units go to where during a transition, the shunting index introduced in section 3.3 is ideal since it describes exactly that. One does need to be careful with splitting and combining, since there the shunting index describes where train units go to in **two** successor trains or where train units from **two** predecessor trains go to in their successor train, and only the information of one successor train or one predecessor train is needed. But for splitting and combining the shunting index is still a good tool to find which train units go where.

In practice, it rarely occurs that the continuity constraint is violated, even if it is not taken into account at all. Clearly, on train lines that contain only one or two trips the continuity constraint cannot be violated, since no entire composition changes are allowed between two trips. So it is not necessary to formulate constraints for these train lines. But also for many more situations it will rarely or not occur that the continuity constraint is violated, so one could try to remove constraints here as well.

One could ask oneself if it is useful to incorporate the continuity constraint into the model if it is rarely violated, and if a situation occurs where it would be violated if it might not be better to allow that violation instead of having to change compositions for trips. But currently the continuity is a strict condition which must be satisfied before a plan will be accepted.

One can implement the flow model for the continuity constraint into the model ore use it as a separate tool. The former makes sure that the continuity constraint holds for any feasible solution of the model, the latter has the advantage that it has no significant influence on the calculation time.

3.7 Remarks

In this section some additional remarks are made about the additions to the model introduced in this chapter. In section 3.7.1 important issues concerning the objective function of the modified model are pointed out. Section 3.7.2 reintroduces the $Y_{t,b}$ variable for the modified model. It turns out that in practice one can drop the integrality constraints on $X_{t,e}$ when using the $Y_{t,b}$ variables and some extra decision variables. Some remarks about the creation of duties for rolling stock units are made in section 3.7.3. Possible further extensions to the model are discussed in section 3.7.4.

3.7.1 Objective Function

An important question is what the objective function should look like for the modified model. There are many different criteria that could be incorporated and given different penalties or bonuses, depending on what is considered important or desirable. As an already existing plan needs to be modified, criteria such as passenger demand and carriage kilometers become less important while criteria such as avoiding different shunting movements become more important.

In the implementation which will be used in chapter 4 the following criteria are considered:

- The inventory deviation for stations. The **inventory** at the start of the day and at the end of the day should preferably be the same as in the original plan.
- **Extra shunting**: Shunting movements introduced by the model at places where no shunting took place in the original plan should be avoided.
- **Different shunting**: Modifying shunting movements compared to the original plan, for example coupling instead of uncoupling, are also undesirable since they require changes in crew member assignment and also require making a new local shunting plan.
- The number of shunting movements at the shunting yard should be minimized, therefore the **combined inventory** should be used when possible and desirable.
- **Locks** can be penalized or given a bonus depending on whether they are desirable.
- Using **shorter trains** than in the original plan is also considered undesirable, as it is assumed that in the original plan a good way of assigning units to trains was used.

Some of these criteria could also be incorporated in the objective function for the original model. For example, the combined inventory and locks are also relevant factors in the tactical rolling stock planning, although they are less essential there. But preferably exceptions should be avoided in an original plan.

3.7.2 Using the $Y_{t,b}$ Variables in the Modified Model

In the additions described in the previous sections the use of the $Y_{t,b}$ variables, which only describe the number of units per type which are used on a trip instead of the actual composition, was largely ignored. Instead the $X_{t,e}$ variables were assumed to be integral. It turns out that the $Y_{t,b}$ variables are still useful to reduce the number of binary variables in the model and that in practice one can even drop the integrality constraint on $X_{t,e}$ when introducing some additional decision variables. Despite this, an example will be given here which shows that there can exist optimal solutions with no feasible integral valued equivalent when dropping the integrality constraint on $X_{t,e}$. Note that introducing combining and splitting to the model already introduced theoretically possible violations of the integrality of variables. In both cases the examples seem to be somewhat pathological.

The extra decision variables needed that were mentioned above are needed for the case where there is more than one extended composition that describes a composition for a trip. This is the case for starters and finishers that can come from and arrive at the combined inventory. When $X_{t,e}$ is not integral, one can construct for starters and finishers similar examples to the example described for coupling and uncoupling in section 3.4 that violate the integrality of variables. One can introduce integral decision variables similar to $z^{C[c]}(t)$ and $z^{C[u]}(t)$ to avoid violations of integrality.

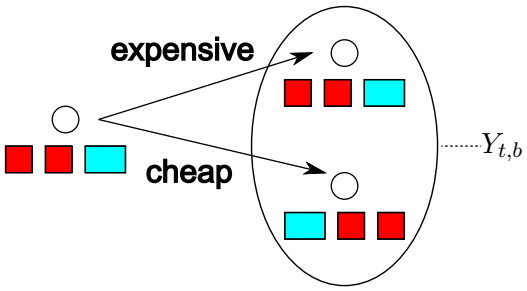


Figure 3.24: A (pathological) example where using the $Y_{t,b}$ variables gives an infeasible fractional solution.

For an example where trouble could emerge even when the extra decision variables for starters and finishers in the combined inventory are included, consider the scenario depicted in figure 3.24. A trip with composition **AAB** has two possible follow-up trips:

- Another trip with composition **AAB**. No shunting movements are needed here.
- A trip with composition **BAA**. In this case, two units are coupled to the front of the train and two units are uncoupled from the rear of the train.

Now suppose that in this case it is cheaper to make the transition to the trip with composition **BAA**. This could for example be the case when after the second trip all units go to the shunting yard and it is preferable that a **BAA** is driven to the combined inventory and the

extra shunting movements are relatively cheap. Suppose also that only one unit of type **A** is stored in the inventory, so in practice the transition to **BAA** is infeasible. But since the latter transition is cheaper and the integrality of $Y_{t,b}$ does not demand that exactly one of these transitions is chosen, an optimal solution would be to use 'half' of both transitions. There is no problem with the inventory here, as only 'half' of two units **A** are needed for the transition from **AAB** to **BAA**. So in this case the optimal solution to the problem is not feasible.

Note that this example is quite pathological. Firstly, a transition from **AAB** to **BAA** would be a very exceptional shunting movement which was not observed even once in the original plans used for this report. Secondly, letting the extra shunting movements be cheaper than using the combined inventory is probably not reasonable. In practice no problems were observed with the integrality when using the $Y_{t,b}$ variables in combination with the extra decision variables for the combined inventory. To avoid any problems, one could first branch on the $Y_{t,b}$ variables and then on the $X_{t,e}$ variables.

3.7.3 Duties

The output of the model consists of compositions assigned to trips and a shunting index for the transitions between trips. Based on this information the duties for train units need to be determined by an extra script. For the modified model the procedure of first determining tasks, then making chains of tasks and finally combining chains of tasks to duties remains the same as in the original model (see also section 2.4.3). Some changes are needed though.

The shunting index introduced in the modified model makes it easier to determine what happens with a unit after a trip. This can be used to simplify the generation of the chains, especially for exceptional shunting movements. Being able to track what happens with the units during an exceptional shunting movement was the main reason to include the shunting index.

When combining chains one needs to take into account whether a unit goes to the normal inventory or to the combined inventory after a trip.

3.7.4 Further Extensions

As the model attempts to describe a very complicated process, the variation in the details that could be added to the model is almost unlimited. In this section some possible further extensions are discussed, focusing on how the additions presented in this chapter could be extended.

Currently, exceptional shunting describes new transitions from a **specific** composition to another **specific** composition, which could for example occur when such a transition was observed in an earlier created plan. In some cases it would be nice if similar transitions would

also be generated automatically. For example, suppose that from a trip with composition **BA** a unit of type **A** is coupled to the front of the train and a unit of type **B** is decoupled from the rear of the train, giving a composition **AA** in the successor trip. Now, if the first trip has composition **BB** instead of **BA**, it might be reasonable to assume that it is also possible to uncouple a **B** from the rear of **BB** and couple a **A** to the front of **BB** giving **BA**. One does need to take into account the shunting capacities of the station: the same exceptional shunting movement might not be possible if the train is longer. But in the case considered in the example, if the unit of type **A** has a length similar to that of unit **B**, there should be no problems. So it would be nice to be able to add a group of possible exceptional transitions based on one observed exceptional transition.

Another possible extension is to create a list of standard shunting codes independently of an original plan. Then one can compare these shunting codes with the shunting codes obtained from the original plan, and in some cases include both codes to increase flexibility. This would also be nice in case of exceptional shunting codes where the original plan does not have a well defined standard shunting code.

When the inventory of units is low it might be desirable to use more locks in the model, as locks can reduce the amount of inventory needed. Currently locks can only be used when they are detected in an original plan, but one might try to incorporate them more systematically. For example one could allow locks for specific times or for specific stations.

Chapter 4

Solution Methods

The composition model is quite complex, and for some problem instances the time needed to find a good solution can be quite high – up to several hours. This chapter discusses some methods to reduce this solution time and also analyzes the effect of the additions presented in the previous chapter on the solution time. In this chapter the main focus is again on solving the operational rolling stock planning problem. For this problem it is even more important that solutions are obtained quickly, so efficient solution methods are needed. Also the scenarios used in this chapter to test the performance of different solution methods are designed for modifying an original plan instead of making a new plan 'from scratch'.

Section 4.1 discusses the complexity of the problem and introduces several scenarios that will be used for testing throughout this chapter. In section 4.2 the most important addition in this chapter is described: a heuristic that uses the LP relaxation of the problem to obtain a good IP solution quickly. A comparison in computation time between the original model and the modified model is made in section 4.3. Finally, section 4.4 describes some other possible ways to reduce the solution time.

4.1 Introduction

In this section some remarks are made about why the composition model is complex and why some instances of the composition model require a lot of computation time. Also, several scenarios are introduced that will be used for testing in later sections.

4.1.1 Complexity

There are different factors that make the composition model a difficult problem for large instances. In this section several complexifying factors are pointed out.

To start with, the size of the problem is quite large. The main example considered in this chapter, the rolling stock problem for the koploper units, consists of over 900 trips and there are 30 different compositions possible for a train, which means that for the transitions between trips there are on average 130 possibilities. In total, there are roughly 160.000 constraints and over 200.000 variables in the IP formulation of the extended composition model. Of these variables roughly 10.000 are integral variables.

As described in section 2.2.4, the rolling stock problem can be described by a transition graph where a network flow needs to be found. But the composition model includes a lot of side constraints which create extra **dependencies** between variables. For example, the inventory constraints link variables that are seemingly unrelated in the transition graph. Also combining and splitting, which are especially common for the train lines operated by koploper units, link a lot of variables. The latter is an important reason why the problem instance for the koploper units is one of the most difficult ones.

Because of the huge size of the problem, **memory consumption** during the calculations can be quite large. Some techniques to reduce the amount of memory needed for preprocessing are discussed in appendix B, but even after applying these techniques memory usage is still quite high. OPL Studio will often need over 800 MB of RAM during calculations and sometimes this memory usage becomes a bottleneck in the solution process. During the experiments described in this chapter some extra restrictions on the model were needed to reduce memory usage.

Since variables are linked to other variables in many different ways, the problem not only becomes more difficult to solve, but it is also intuitively less transparent. One of the places where this becomes very apparent is in the **branch and bound** algorithm. It turns out to be quite difficult to predict beforehand how this will take place. For example the number of nodes that will need to be searched and the maximal search depth can differ a lot from instance to instance without any apparent reason. Since the branch and bound algorithm is also the most important time consuming part of the solution process, it is difficult to predict how much time will be needed to solve a problem instance. This makes it difficult to compare different solution methods where the branch and bound algorithm is influenced: one method could be much better than the other but due to 'bad luck' in branching the other method might still be faster for specific instances.

There are also some factors that reduce the complexity of the problem. When the model is used to modify an existing plan instead of planning 'from scratch', which is the main problem under consideration in this chapter, a lot of extra information is available. For example, one of the objectives is to make sure that trains are not much shorter than in the original plan. The original plan might even be a feasible integral solution to the problem, which would be a good starting point in finding a new optimal solution.

Another property of the rolling stock planning problem is that the solution of the LP relaxation of the problem lies quite close to the IP solution. This also gives a lot of information which will be used in a heuristic introduced later in this chapter, which can greatly improve the solution time.

4.1.2 Scenarios

In order to test how the model performs under different circumstances, several scenarios have been created. Fifteen scenarios are considered, divided into three cases with five different objectives each. The scenarios all consider the rolling stock planning problem for a specific day for the koploper units, where for the different cases the total number of units in the inventory is varied. The following cases are considered:

1. The first case has the same number of train units available as the original plan, which makes the original plan a feasible solution of the model.
2. In the second case the number of units available is reduced a bit. This is done to try to 'simulate' train units being unavailable due to maintenance. The original plan is infeasible in this case, although most of the plan can still be reused.
3. In the third case the number of units available is reduced even more. This case requires quite a few changes to be made to the original plan, making this the most difficult case.

The number of units available in each case is shown in table 4.1.

Case	# ICM-3	# ICM-4
1	78	45
2	72	39
3	66	33

Table 4.1: The three cases considered during testing. ICM-3 and ICM-4 are the two types of koploper units that are used, where an ICM-3 unit consists of 3 carriages and an ICM-4 unit consists of 4 carriages. For each case five different objectives are considered.

For each case five different objectives are considered. The objective function consists of the criteria described in section 3.7.1, they are repeated here for clarity:

- The inventory deviation for stations. The **inventory** at the start of the day and at the end of the day should preferably be the same as in the original plan.
- **Extra shunting**: Shunting movements introduced by the model at places where no shunting took place in the original plan should be avoided.
- **Different shunting**: Changing shunting movements compared to the original plan, for example coupling instead of uncoupling, are also undesirable since they require changes in crew member assignment and also require making a new local shunting plan.
- The shunting movements at the shunting yard should be minimized, therefore the **combined inventory** should be used when possible and desirable.
- **Locks** can be penalized or given a bonus depending on whether they are desirable.

- Using **shorter trains** than in the original plan is also considered undesirable, as it is assumed that in the original plan a good way of assigning units to trains was used.

In the different objectives for the cases, the weight factors for the different criteria are varied. The following five objectives are considered:

1. The 'normal' objective. Here the start and end inventory at stations is not taken into account and the other objective criteria have similar weighting factors.
2. In the second objective the start and end inventory at stations is also taken into account.
3. The third objective considers the combined inventory to be less important and thus it gets a lower weighting factor.
4. A shorter train length than planned in the original plan is not considered important in the fourth objective.
5. In the fifth objective different shunting is not penalized, only extra shunting is considered to be bad.

Table 4.2 shows the weighting factors in the different scenarios. Note that most weighting factors used are in the order of 10^6. For readability in the results presented in the following sections, this factor 10^6 will be omitted in the values of the objective functions of solutions.

Objective	Inventory Deviance	Extra Shunting	Different Shunting	Combined Inventory	Shorter Trains	Locks
1	1	200000	200000	200000	100000	100000
2	100000	200000	200000	200000	100000	100000
3	1	200000	200000	20000	100000	100000
4	1	200000	200000	200000	10000	100000
5	1	200000	1	200000	100000	100000

Table 4.2: The weighting factors for the different criteria in the objective function in the five different objectives considered.

Note that in order to give more credibility to the tests many more dimensions can be considered in the scenarios. Different days for the koploper units can be considered and different types of rolling stock can be used. Also other objective functions might be considered, for example if one wants to plan 'from scratch' using some of the additions. But the case considered with the koploper units is one of the hardest cases and therefore it is expected that similar results can be obtained when varying other dimensions.

In order to reduce the size of the problem, some restrictions were made on the combined inventory in the scenarios described here. Only starters and finishers can use the combined inventory, units coupled to a train and units uncoupled from a train must use the normal inventory. This is done because for some problem instances memory problems occurred during the solution process. It is not a big restriction because coupling or uncoupling two or more

units rarely occurs. The improved solution method described in the next section can easily take coupling and uncoupling for the combined inventory into account though.

Also, the continuity constraint was not taken into account in the different scenarios. As described in section 3.6, it might be more realistic to use the flow model for the continuity constraint as a tool to check whether this constraint holds for all train lines. In the original composition model higher priorities were assigned to variables in the rush hours to improve the branch and bound algorithm, but since this seems to have an arbitrary effect for the objective functions considered in the scenario's, the default priorities assigned by CPLEX are used for the variables in this chapter.

4.2 Using the LP Relaxation

It turns out that the solution of the LP relaxation of the composition model, where all integrality constraints are dropped, lies quite close to the IP solution. This section explains in what way the LP relaxation and the IP problem resemble each other, and how this information can be used to obtain good solutions in a short time. The basic idea is to use the LP relaxation to fix some variables in the IP problem, in order to make the latter problem computationally less expensive. This idea is explained in section 4.2.1, where also some indications are given about why this approach could work. Several approaches to fix variables and the results of these approaches are presented in section 4.2.2. Finally, some remarks about the new solution method are made in section 4.2.3.

4.2.1 Basic Idea

In this section is explained how the LP relaxation and the IP problem look alike and how this information can be used.

Consider table 4.3. For all fifteen scenarios the value of the objective function is given for the optimal solution, both for the LP relaxation of the problem and the IP problem itself. It turns out that the gap between the LP relaxation and the IP problem is quite small, in the experiments considered it is at most 3.5% and on average around 1%. This is a first indication that the LP relaxation might be used to simplify the IP problem.

Although the gap between the LP relaxation and the IP problem is quite small, the two corresponding solutions might still differ a lot, since only the value of the objective functions is compared. But when looking at the LP relaxation, it turns out that quite a lot of variables are integral and equal to the corresponding values in the IP problem. Table 4.4 quantifies this observation. Here for all fifteen scenarios the number of fractional variables in the optimal solution of the LP relaxation is given, and the number of differences between the non-fractional variables in the LP relaxation and their corresponding variables in the IP problem. The latter number is quite low for all scenarios considered, normally only a few dozen variables or even less. And the number of fractional variables in the LP relaxation is relatively small, for the

Case	Objective	LP Relaxation	IP Solution	Gap
1	1	3.87	3.90	0.8%
	2	28.47	28.50	0.1%
	3	0.39	0.39	0.0%
	4	3.83	3.86	0.8%
	5	1.00	1.00	0.0%
2	1	23.61	23.90	1.2%
	2	45.81	46.10	0.6%
	3	20.15	20.69	2.6%
	4	8.17	8.47	3.5%
	5	16.75	16.80	0.3%
3	1	64.06	64.90	1.3%
	2	83.86	84.70	1.0%
	3	61.09	61.77	1.1%
	4	14.75	15.22	3.1%
	5	51.46	51.60	0.3%

Table 4.3: The value of the objective funtion in the solution of the LP relaxation and the solution of the IP problem. Note that the gap between these two values is quite small.

more complex cases (cases 2 and 3) it can be up to 200 variables, about a quarter of the total number of trips. Note that for some scenarios there are zero fractional variables in the solution of the LP relaxation but several differences with the IP solution. This means that there are several optimal integral solutions, an indication of flexibility which will be elaborated later in this section.

Some explanation is needed on which variables are considered here. As they determine almost the entire solution, only the $Y_{t,b}$ variables are considered. Because the $Y_{t,b}$ variables for one trip are related, only one 'variable' is counted per trip: this 'variable' is fractional if two or more $Y_{t,b}$ variables are fractional for this trip, and it is integral if exactly one of the $Y_{t,b}$ variables is equal to 1. So there are around 900 (the total number of trips) variables considered in the comparison.

Considering table 4.4, the integral values of the optimal solution of the LP relaxation are usually a good indication of what will happen in the solution of the IP problem. This gives rise to the following idea which is also depicted in figure 4.1:

> *Fix (a part of) the variables in the IP problem to their corresponding values in the LP relaxation to make the IP problem smaller and thus easier to solve.*

If one would know in advance which integral variables in the LP relaxation have the same value as their corresponding variables in the IP solution one could fix precisely these. But finding these values is not easy, although some attempts will be described in section 4.2.3. Fortunately, it turns out that it is usually not a problem if some 'wrong' variables are fixed, there seems to be a lot of **flexibility** in finding solutions.

Case	Objective	# Fractional	# Different (int)
1	1	0	12
	2	0	4
	3	0	2
	4	22	1
	5	6	4
2	1	64	22
	2	64	12
	3	79	26
	4	105	22
	5	4	2
3	1	210	0
	2	210	5
	3	205	12
	4	251	66
	5	97	16

Table 4.4: A comparison of the variables in the solution of the LP relaxation and the solution of the IP problem. The first column ('# Fractional') gives the number of fractional variables in the LP relaxation, the second column ('# Different (int)') gives the number of variables that are integral in the LP relaxation and that have a different value for the IP problem.

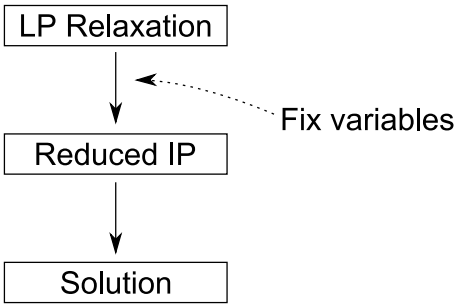


Figure 4.1: Using the LP relaxation.

Tables 4.5 and 4.6 attempt to quantify this flexibility. For table 4.5, for every scenario 20 runs have been made, where every run has as extra restriction that it should differ in at least 20 variables from all previous runs in the same scenario. The table gives the worst objective value for a solution obtained in this way. As it turns out, the objective value of the solution does not become much worse for successive runs, indicating that there are a lot of 'good' solutions near the optimal solution.

Some more explanation of the table is needed. Recall here that a factor 10^6 was dropped from the objective value in the tables in this chapter. The relative gap between the objective values of the IP problem and the worst IP problem with extra restrictions is sometimes not a good indication of how much they differ. This is because the objective value makes big jumps due to the integrality constraints: in the 'normal' objective every criterion has a weight in the order of 100.000, so if the objective value becomes worse in a solution it will become worse

by at least 100.000. But in the table one can note that the absolute gap is usually not big considering the weight factors, it indicates just a few changes.

Case	Objective	IP Solution	Worst IP Sol.	Rel. Gap	Abs. Gap
1	1	3.90	4.40	11.4%	0.5
	2	28.50	29.00	1.7%	0.5
	3	0.39	1.00	61.0%	0.6
	4	3.86	4.06	4.9%	0.2
	5	1.00	1.00	0.0%	0.0
2	1	23.90	24.20	1.2%	0.3
	2	46.10	46.40	0.6%	0.3
	3	20.69	21.11	1.9%	0.4
	4	8.47	8.86	4.5%	0.4
	5	16.80	16.80	0.0%	0.0
3	1	64.90	65.20	0.5%	0.3
	2	84.70	85.00	0.4%	0.3
	3	61.77	62.12	0.6%	0.4
	4	15.22	15.59	3.8%	0.6
	5	51.60	51.80	0.4%	0.2

Table 4.5: The objective value of the IP solution compared to the worst IP solution obtained in successive runs where every run must differ in at least 20 variables from all previous runs. The fact that the absolute gaps between these two solutions is not that big indicates that there are a lot of 'good' solutions around the optimal solution.

Table 4.6 compares the IP solution to an IP solution where the restriction is made that at least 100 variables are different from the original solution. Although this is a quite large restriction, for at least 100 trips the composition must be different from the original solution, the results are not that bad although significantly worse than the original solution. But this shows that even with a lot of changes a reasonable solution can be found, hinting again that the model is quite flexible in finding good solutions.

As a side remark, a bit of cheating was used in generating these statistics to decrease the total running time. The method to obtain good IP solutions faster by fixing variables using the LP relaxation, that is described in this and the next section, was already used here. Otherwise the total computation time to obtain the tables would be up to several weeks. This means that the results actually give an **upper bound** of the worst IP solution found, when using the 'real' IP solutions the gaps between the original IP solutions and the solutions of the IP problems with extra restrictions could actually be even smaller.

Summarizing the above, there are three major observations:

1. The gap between the objective values of the solution of the LP relaxation and the solution of the IP problem is quite small;

Case	Objective	IP Solution	Constrained IP Sol.	Rel. Gap	Abs. Gap
1	1	3.90	5.80	32.8%	1.9
	2	28.50	30.40	6.3%	1.9
	3	0.39	2.63	85.2%	2.2
	4	3.86	4.56	15.4%	0.7
	5	1.00	1.00	0.0%	0.0
2	1	23.90	25.00	4.4%	1.1
	2	46.10	47.10	2.1%	1.0
	3	20.69	21.17	2.4%	0.5
	4	8.47	8.64	2.3%	0.2
	5	16.80	16.80	0.0%	0.0
3	1	64.90	65.10	0.3%	0.2
	2	84.70	84.90	0.2%	0.2
	3	61.77	61.94	0.3%	0.2
	4	15.22	15.57	2.6%	0.4
	5	51.60	51.80	0.4%	0.2

Table 4.6: A comparison between the objective value of the IP solution and the IP solution where at least 100 variables should be different. Although this means that the compositions of at least 100 trips need to be changed compared to the optimal IP solution, the absolute gap remains quite small. This is another indication of flexibility of the IP solutions.

2. The solution of the LP relaxation contains a lot of integral variables and most of these variables have the same value as their corresponding variables in the solution of the IP problem;

3. The IP problem is flexible – even if some variables are forced to be different from their value in the optimal IP solution it is still possible to find a good solution.

This indicates that the strategy suggested earlier in this section, to fix some of the variables in the IP problem to their corresponding values in the solution of the LP relaxation of the problem, might give good solutions. A remaining question is what is meant by fixing 'some' variables. This question is looked into in section 4.2.2, where three different approaches are tested and some results are given.

4.2.2 Results

As described in the previous section, a possible way to obtain solutions faster is to fix certain integral variables in the IP problem to their corresponding values in the solution of the LP relaxation of the problem. In this section different approaches to this are discussed and the results for the different scenarios are given.

Three different methods to fix integral variables in the IP problem are considered here:

1. A first approach is to find variables in the IP problem that are **'close'** to variables that are fractional in the solution of the LP relaxation, and allow these variables to be free (so not fixed in the IP problem). For example, if a trip has fractional variables and its follow-up trip has integral variables, it would be reasonable to allow the latter variables to be free in the IP problem. So here one tries to use the structure of the problem to determine which variables to fix. In this approach 60% of the variables are chosen to be free in the IP problem, including the variables corresponding to the fractional variables of the solution of the LP relaxation.

2. As a second approach one can fix **all** variables in the IP problem that have an integral value in the LP relaxation of the problem. So only the variables that are fractional in the solution of the LP relaxation can be chosen freely in the fixed IP problem.

3. A third approach is to fix some variables in the IP problem corresponding to the integral variables in the solution of the LP relaxation **arbitrarily** and in such a way that roughly 40% of the total number of variables are fixed. So 60% of the variables are free, including the variables that are fractional in the LP relaxation of the problem.

In tables 4.7 and 4.8 the results of using the approaches above are compared to the unrestricted IP problem for the fifteen scenarios. Here 'Fixed IP 1', 'Fixed IP 2' and 'Fixed IP 3' correspond to the three approaches given above.

Table 4.7 compares the values of the objective functions of the IP solutions, found for the different approaches and for the unrestricted IP problem. It turns out that often the value of the objective function is the same for all approaches and for the unrestricted IP problem. And if there is a difference in value of the objective functions between the approaches and the unrestricted IP problem this difference is usually small. This indicates that all three new approaches can still find good solutions for the IP problem even though a lot of variables are fixed, which should reduce the solution space a lot. Note especially the results of the second approach, where all variables are fixed that correspond to variables with non-fractional values in the solution of the LP relaxation. In most cases this implies that more than 75% of the variables are fixed. Nevertheless, good solutions are found even in this case.

In table 4.8 the total time in minutes required to solve the problems is given. This includes the time required for the three new approaches, the time until the best solution for the unrestricted IP problem was found and the time until this best solution was proven to be the best solution. The time required for the three approaches is the total computation time, which includes a separate run of the LP relaxation to determine which variables need to be fixed. A first observation is that the total solution time for the three new approaches is roughly the same for all scenarios, usually in the order of 5-6 minutes. For the unrestricted IP problem the solution time of the first five scenarios, corresponding to the first case, is quite low. But for the remaining scenarios the solution time can be over one hour in some cases.

Case	Objective	IP Problem	Fixed IP 1	Fixed IP 2	Fixed IP 3
1	1	3.90	3.90	3.90	3.90
	2	28.50	28.50	28.50	28.50
	3	0.39	0.39	0.39	0.39
	4	3.86	3.86	3.86	3.86
	5	1.00	1.00	1.00	1.00
2	1	23.90	24.00	24.00	24.00
	2	46.10	46.20	46.20	46.20
	3	20.69	20.81	20.93	20.93
	4	8.47	8.57	8.71	8.71
	5	16.80	16.80	16.80	16.80
3	1	64.90	64.90	64.90	64.90
	2	84.70	84.70	84.70	84.70
	3	61.77	61.86	61.86	61.85
	4	15.22	15.46	15.65	15.45
	5	51.60	51.80	51.80	51.70

Table 4.7: Comparing the solution of the unrestricted IP with the solutions of the new approaches. The solutions of the new approaches seem to be not much worse than the solution of the unrestricted IP.

4.2.3 Remarks

The value of the Fixed IP Approach

Comparing the new approaches with the unrestricted IP problem, the solution found by the three approaches often has the same objective value or is at least not much worse than the solution of the unrestricted IP problem. Furthermore, the solution time of the new approaches is often much lower than the solution time of the unrestricted IP problem.

Note that for the first five scenarios the solution time of the unrestricted IP problem is a bit better than for the three new approaches. The comparison here is a bit unfair. For the new approaches the preprocessing time is counted twice, once for the LP relaxation and once for the fixed IP problem, while this preprocessing is almost the same for both problems, only one extra constraint is added for the fixed IP problem. Also, note that the LP relaxation could be used as a starting point for the fixed IP problem. Some technical restrictions of OPL Studio make it infeasible to use these facts, in a future implementation these facts can be taken into account.

Sometimes the solution of the unrestricted IP problem is slightly better than the solutions obtained by the new approaches. A question is whether this is relevant, and even if one can speak of 'better' here. Because there are a lot of factors that are not taken into account, it is for example not sure at all that all planned shunting movements will be allowed by local planners, it is difficult to say something about this. Therefore it might not be interesting to spent effort to obtain the 'best' solution from a set of good solutions.

Case	Objective	Complete IP	Best Sol. IP	Fixed IP 1	Fixed IP 2	Fixed IP 3
1	1	4	4	5	5	5
	2	4	4	5	5	5
	3	4	4	5	5	5
	4	5	5	5	5	5
	5	4	4	5	5	5
2	1	11	11	6	5	5
	2	7	7	6	5	5
	3	31	24	8	5	6
	4	31	16	6	5	5
	5	5	5	6	5	5
3	1	112	96	6	5	6
	2	108	100	6	5	6
	3	23	12	7	5	6
	4	> 180	57	16	5	6
	5	26	26	6	5	6

Table 4.8: Comparing the solution times of the unrestricted IP problem and the new approaches, where for the unrestricted IP problem also the time until the best solution was found is given. The new approaches have a much shorter computation time in most cases.

The results presented in the previous section are quite nice, but one should be careful when generalizing the results to different problems. For other problems the gap between the LP relaxation and the IP solution might be larger, or the solution space could be less flexible if more constraints are introduced. In some cases it might be even be possible that the fixed IP problem is infeasible, then more variables should be set free.

Future research could be done on better approaches to find the best set of integral variables to fix, but the current approaches already give quite good results. As described above, the notion of the 'best' solution is already a dubious one, so there seems to be not much reason to prefer the unrestricted IP problem above the fixed IP problem. In any case one can calculate the gap between the LP relaxation and the solution of the fixed IP problem in order to determine whether the fixed IP problem has a reasonable solution.

Future Work

Interestingly, the more 'intelligent' approach to find the best variables to fix does not seem to perform better than arbitrarily choosing variables where the same amount of variables is fixed. Both these approaches do give equal or better solutions than the other approach where all variables are fixed that correspond to non-fractional variables of the solution of the LP relaxation. This makes sense since the solution space of the latter approach is strictly contained in the solution spaces of the two other approaches. An interesting question is why the intelligent approach does not work better than the arbitrary approach. One possible answer is that the intelligent approach used in the experiments might not be intelligent enough. When

analyzing the variables of the IP solution which have a corresponding but different valued integral variable in the solution of the LP relaxation, it turns out that sometimes they do seem related to fractional variables, but in such a way that is difficult to find these relations 'automatically' with an easily implementable intelligent approach.

Maybe a better way to find important variables to keep free is to use one or more extra runs of the LP relaxation of the problem, where some of the fractional variables of the first LP relaxation are forced to be different. This way one can observe a bit of the dynamics between the variables, how the fractional variables are related to each other and to other variables. Some initial attempts to do this have been made, where some variables that needed to be kept free were found, but more research could be done here.

As remarked before, the current implementation to fix integral variables is not very efficient. The entire model, including all constraints and variables, is rebuilt two times – once for the LP relaxation and once for the fixed IP problem. But the fixed IP problem actually contains only one extra constraint. Furthermore, the solution of the LP relaxation of the fixed IP problem is just the solution of the LP relaxation of the original problem, which was already calculated. In a better implementation the model should be built only once, and also the solution of the LP relaxation should be used in the fixed IP problem. This would reduce the total calculation time even more.

Although it was remarked earlier that it might not be relevant to search for solutions that have a better objective value than the solution found by fixing the IP problem, one could still use the solution found for the fixed IP problem as a starting point for finding a better solution. In the branch and bound algorithm parts of the tree could be discarded this way. In the current implementation this approach is not possible, but a future implementation in another program might use this method.

Other information could be extracted from the LP relaxation. In the approaches considered, the variables in the IP problem that correspond to variables that are fractional in the solution of the LP relaxation are set free. But often the values of the fractional variables give some information about what will happen in the IP problem. For example, for a trip one $Y_{t,b}$ variable could be equal to 0.8 and another variable could be equal to 0.2. Often the variable with value 0.8 will be chosen in the solution of the IP problem, although this is not always the case.

4.3 The Effects of the Additions on the Solution Time

This section compares the computational performance of the original composition model with that of the modified model. Although the modified model incorporates more details of the shunting process, it is also important that one can obtain good solutions quickly. So if the modified model performs much worse in this perspective, the original model might still be a better option in some cases.

One factor that makes it difficult to compare the two models was already mentioned in the introductionary section 4.1 about complexity. The branch and bound process can be quite different for problems that would appear to be quite similar. Therefore, the solution time can vary a lot, making it difficult to compare similar problems. The time required to solve the LP relaxation might be a better indication in these cases.

In this section the main focus is on the effects of the introduction of the combined inventory and the locks to the solution time, as this introduced most new constraints and links between variables. The composition model with these two additions is compared to the composition model without these two additions for the fifteen scenarios.

		Best		Total	
Case	Objective	Modified IP	Original IP	Modified IP	Original IP
1	1	4	3	4	3
	2	4	3	4	3
	3	4	3	4	3
	4	5	3	5	3
	5	4	3	4	3
2	1	11	14	11	19
	2	7	15	7	19
	3	24	15	31	19
	4	16	41	31	41
	5	5	6	5	6
3	1	96	66	112	68
	2	100	30	108	44
	3	12	66	23	68
	4	57	31	> 180	76
	5	26	13	26	13

Table 4.9: A comparison between the solution times of the original model and the modified model. Since the branch and bound procedure performs quite different even for similar looking problems, not much can be concluded from these results, although the original model seems to perform a bit better on average as could be expected.

The results of this comparison are shown in tables 4.9 and 4.10. Table 4.9 compares the solution times of the unrestricted IP problem for the original model and the modified model. Here the remarks about the branch and bound procedure become apparent: for the more difficult cases one of the two models is better in a quite arbitrary way, although in general the original model seems to be a bit faster.

Table 4.10 might give a better indication of the performance of the two models. Here the solution times of the LP relaxation of the problem and the fixed IP problem are compared. For most scenarios the original model performs better than the modified model, although the difference is not dramatical.

So the modified model has a significantly though not dramatically larger solution time compared to the original model, when considering the fixed IP problem. If only one run of the

		LP Relaxation		Fixed IP	
Case	Objective	Modified LP	Original LP	modified IP	Original IP
1	1	3.3	2.5	5.3	4.6
	2	3.2	2.5	5.2	4.6
	3	3.2	2.5	5.2	4.6
	4	3.0	2.4	5.1	4.5
	5	3.3	2.9	5.4	5.0
2	1	3.3	2.7	6.1	7.1
	2	3.3	2.7	6.2	5.4
	3	3.4	2.7	7.9	7.1
	4	3.0	2.5	6.4	6.4
	5	3.4	2.7	5.6	5.7
3	1	3.2	2.6	6.5	5.3
	2	3.2	2.7	6.3	5.7
	3	3.4	2.6	6.8	5.3
	4	3.1	2.5	16.3	10.0
	5	3.6	2.8	6.7	5.3

Table 4.10: A comparison between the original model and the modified model. Here the solution times of the LP relaxation and the fixed IP problem are compared. The original model performs clearly better, although the results are not dramatical.

problem is required the total solution time is still in the order of a few minutes, in which case this does not matter much, but if several runs are required this might become an important issue.

4.4 Other Improvements to the Solution Time

In this section several other approaches are suggested that could improve the solution time of the rolling stock planning problem. The unrestricted IP problem is considered instead of the fixed IP problem, although the ideas presented in this section might also be applied to the latter approach.

Three ideas are discussed in this section. The first one is to use Special Ordered Sets to improve the branch and bound algorithm, this is discussed in section 4.4.1. Section 4.4.2 discusses the use of perturbation to improve the dual simplex method used for subproblems that occur in the branch and bound algorithm. In section 4.4.3 some remarks are made about how one could assign priorities to variables in a useful way to improve the branch and bound algorithm.

Several technical restrictions make it impossible to implement and test some of the ideas. It is not possible in the current implementation to obtain detailed information about the branch and bound process, for example which nodes are chosen successively. The SOS constraints

described in section 4.4.1 can also not be implemented. Therefore the ideas presented in this section can be considered to be suggestions for future research.

4.4.1 SOS Constraints

The $Y_{t,b}$ variables have two important properties that might be exploited: for every trip t, they can be ordered by the number of carriages they contain, and exactly one of the variables corresponding to t is equal to one. These properties make the $Y_{t,b}$ variables into **Special Ordered Sets of Type I**, and this can be used to improve the branch and bound procedure.

The Theory

Consider the binary variables $x_1, ..., x_k$ in some IP problem where exactly one of the variables must be equal to 1, $\sum_{i=1}^{k} x_i = 1$. A weight w_i is associated with each variable giving an ordering of the variables, let $x_1, ..., x_k$ be ordered according to this weight, so $w_1 < w_2 < ... < w_k$. Now suppose that during some subproblem of the branch and bound process a fractional solution x^* emerges where some of the $x_1^*, ..., x_k^*$ variables are fractional. In this case, the standard branch and bound process would select one fractional variable x_j^* and divide the currently considered solution space S into $S_1 := S \cap \{x : x_j = 0\}$ and $S_2 := S \cap \{x : x_j = 1\}$. Note that the set S_1 will be in general much larger than S_2, since in S_2 all variables $x_1, ..., x_k$ are fixed but in S_1 only x_j is fixed and most other variables are still free. The branch and bound tree is unbalanced here, and this might cause a lot of extra nodes to be searched.

Another way to divide the solution space into two parts in this case is to split the variables $x_1, ..., x_k$ into two groups and divide the solution space in such a way that all variables in one group must be zero: let $S_1' := S \cap \{x_1 = ... = x_r = 0\}$ and $S_2' := S \cap \{x_{r+1} = ...x_k = 0\}$. The value $r \in \{1, ..., k\}$ is chosen here in such a way that the fractional solution x^* is not allowed, using the weights associated with the variables: let $w = \sum_{i=1}^{k} w_i x_i^*$ then $w_1 < ... < w_r \leq w \leq w_{r+1} < ... < w_k$. The idea here is that this subdivision of S in S_1' and S_2' is more balanced, so the associated subtrees have roughly the same size. And this could reduce the amount of nodes that need to be searched since the node search depth will not be too deep.

In the above, the variables $x_1, ..., x_k$ are called a Special Ordered Set (SOS) of type I. See for example [4] for more information about Special Ordered Sets.

Possible Application

The $Y_{t,b}$ variables for each trip form a Special Ordered Set of type I when ordered by the number of carriages, so one could adapt the branching process to this. Intuitively, this approach also makes sense: when branching on individual binary $Y_{t,b}$ variables, the solution space is divided by the decision *"the trip consists of x carriages or the trip does not consist of x carriages"*, when using the SOS constraints this decision becomes *"the trip has x or less*

carriages, or the trip has at least x carriages". The latter decision seems to be a more natural decision.

It is possible to specify in CPLEX which variables form a Special Ordered Set. Unfortunately, the program used to implement the composition model does not allow one to specify such sets, so no results are described in this report. Future research could determine whether using Special Ordered Sets would be a significant speed improvement to the model.

4.4.2 Perturbation

When solving the IP problem described by the composition model, first the linear relaxation of the problem is solved using an interior point algorithm, the barrier method. For the subproblems generated by the branch and bound process the dual simplex algorithm is used to find linear relaxations. When applying the dual simplex algorithm it can sometimes occur that during successive iterations the basic solutions found do not improve the objective. This phenomenon is called **stalling**. A technique called **perturbation** can be used to avoid stalling. In this section this technique is described briefly and some experiments are done to see if this technique can decrease the solution time of the composition model.

The main idea of perturbation is to make small random modifications to the objective function such that stalling cannot occur. This way the objective improves during successive simplex iterations and possibly less iterations are needed. More information about perturbation can for example be found in [2] and [3], although in these papers a more ad hoc approach is used.

In the current implementation of the composition model it is not possible to analyze the simplex iterations in order to determine if stalling occurs. For some cases CPLEX might use a form of perturbation automatically. But some experiments have been done to determine if manually including perturbation in the model can improve the solution time. To do this, some random small disturbances have been introduced in the objective function. The results of the experiments are given in table 4.11.

From table 4.11 can be seen that adding perturbation to the IP problem seems to give arbitrary results compared to the IP problem without perturbation: in some cases better solution times are be obtained, in other cases the solution time becomes much worse. The problem in comparing different solutions described earlier occurs here: slightly different problems can have very different branch and bound processes with very different solution times. But on average perturbation does not seem to improve the solution time. One of the reasons for this could be that not much stalling occurs in the dual simplex algorithm applied, another reason could be that CPLEX already does some perturbation automatically when necessary which would make manual perturbation unnecessary.

		Best Sol.		Total	
Case	Objective	Original IP	IP + Pert.	Original IP	IP + Pert.
1	1	4	4	4	4
	2	4	4	4	4
	3	4	4	4	4
	4	5	7	5	7
	5	4	5	4	5
2	1	11	11	11	11
	2	7	15	7	16
	3	24	15	31	15
	4	16	113	31	118
	5	5	6	5	6
3	1	96	169	112	169
	2	100	125	108	> 180
	3	12	25	23	58
	4	57	41	> 180	> 180
	5	26	20	26	20

Table 4.11: A comparison between the IP problem and the IP problem with perturbation, where the time until the best solution was found and the total solution time are compared. With perturbation no better solution times seem to be obtained.

4.4.3 Improving the Node Search

Another idea to improve the solution time is to give priorities to the different variables in order to reduce the amount of nodes that need to be searched during branch and bound. The idea here is that if the most important decisions are made quickly in the branching process the rest of the problem becomes easier to solve. For the original composition model this was done already: variables that describe the compositions of trains during the rush hours were given a high priority. The modified model uses an entirely different objective function so a similar assignment of priorities has not much effect. But a different assignment of priorities might significantly reduce the solution time.

Unfortunately, with the current implementation of the composition model it is not possible to determine what happens exactly during the branch and bound process, specifically what nodes are selected during branching. Also, at the start of the solution process CPLEX can automatically eliminate integral variables to simplify the problem, but because no information about which integral variables are eliminated can be obtained, it is difficult to say how priorities assigned to variables are used in the branch and bound process.

Nevertheless, an attempt was made to implement a better way to do branching by assigning high priorities to variables that should be 'important'. Important variables were considered to be variables that **influence** a lot of other variables, for example variables that have a lot of follow up trips and predecessor trips. A score was assigned to all variables based on this principle, where trips that are 'near' a variable are given a higher weighting factor for the

score of that variable. The scores of the variables were used as priorities. But in these initial experiments no real improvements in solution time were obtained.

Chapter 5

Conclusions and Discussion

In this chapter a summary is given of this report and suggestions for future development are made. Section 5.1 describes the various additions to the model and the solution method, and summarizes the results. In section 5.2 suggestions are made for further improvements for the model and how the model could be used.

5.1 Conclusions

In this report some additions have been presented with the main motivation to make an existing model for solving the tactical rolling stock problem, the composition model, capable of planning closer to the execution date of the problem. This means that the model should be able to modify an existing plan instead of planning 'from scratch', and shunting movements become a more important factor to consider.

5.1.1 Additions to the Model

To make the model capable of modifying an existing plan, an addition was introduced that allows **exceptional shunting movements** to be incorporated in the model instead of only allowing standard shunting movements. Here exceptional shunting movements are shunting movements that do not occur frequently but that are sometimes planned. With the addition, exceptional shunting movements are detected in an original plan and are given as input to the model. This way an original plan with exceptional shunting movements becomes a **feasible input** for the model. A concept called the **'shunting index'** was introduced that describes what happens to the units in the train during shunting. This concept was used for the exceptional shunting movements and on various other places in the implementation of the model.

Two important phenomena that were observed when studying the shunting movements at the stations are:

- Often entire trains go to the shunting yard and leave the shunting yard later without composition changes. This observation was made into an addition for the model by extending the inventory of individual units with an **inventory of combined units**.
- Sometimes train units arrive at the station and are reused quickly without departing to the shunting yard first. This type of fast shunting movements was also introduced in the model by specifying **locks** between specific trips. Allowing fast shunting movements is especially useful when the inventory is lower than planned, which often happens close to the execution date of the plans.

Another addition is a new way to implement the **continuity constraint**, the constraint that there should be at least one unit that participates in all trips of a train. A flow model was used to describe this, and the new implementation takes into account concepts such as splitting, combining and exceptional shunting movements. In practice the continuity constraint is often satisfied for most train lines without explicitly demanding it, so the new flow model for the continuity constraint could perhaps be used as a separate tool instead of being implemented in the composition model.

5.1.2 Additions to the Solution Method

An important factor, especially when planning closer to the execution time of the plans, is the solution time. In order to improve the solution time a heuristic was introduced that uses the LP relaxation of the problem to simplify the IP problem. It turns out that a lot of variables in the solution of the LP relaxation are integral, and that the values of these variables are often equal to their corresponding variables in the solution of the IP problem. A subset of the variables in the IP problem is **fixed** to their corresponding values in the LP relaxation, which reduces the size of the IP problem. Even when this fixing is done in an arbitrary way, the resulting fixed IP problem still gives good solutions.

The total solution time observed when testing the fixed IP heuristic was in the order of 5-6 minutes, while the solution time for the original problem was sometimes over an hour. The fixed IP heuristic gives comparable solutions to the IP problem but in a much shorter time. Some possible strategies were described to improve the solution time even more, although more research and a new implementation of the model are needed to determine whether these strategies are a valuable contribution.

The performance of the original composition model was compared with the modified model. Although the original model is significantly faster than the modified model, the computation time for the modified model is not much larger.

5.2 Future Work

In this section some suggestions are made for future research and usages for the composition model. Section 5.2.1 describes some further additions to the model. In section 5.2.2 the future of the model is discussed, how it could be used in practice for example.

5.2.1 Further Improvements to the Model

The fast shunting movements, or locks, are currently incorporated in the model if and only if they occur in an original plan. One could incorporate them in a more general way, as they add some extra flexibility to the model and might especially be useful for planning when the inventory is low. In the latter case locks are useful since they reuse units in a very efficient way. A more general way to include locks could be to allow them for specific times or for specific stations.

When exceptional shunting movements are described in an original plan, one would still like to have an 'original standard shunting code' so that the exceptional shunting movement can be discarded by the model if a better option is available. This could also hold for transitions in the original plan that can be interpreted with a standard shunting code but where another standard shunting code would be used normally. So standard codes for stations could be introduced in the current implementation, this is already done partially. Also including exceptional shunting movements could be done more intelligently by allowing similar exceptional shunting movements, see section 3.7.4.

The current implementation of the model was originally designed as a prototype for testing, and the program used to implement the constraints and variables is more suitable for testing than for a final implementation. Rewriting the entire model would make the model more efficient and the code more transparent.

As described in section 4.4, a different programming language is needed to test some further improvements. In the current implementation it is not possible to analyze the branch and bound process in detail, and Special Ordered Sets can not be used.

In section 4.2.3 some more suggestions were made for improving the solution time, building on the method of fixing a part of the IP problem. Possibilities include searching for better methods to determine which variables to fix and taking into account the values of the fractional variables in the solution of the LP relaxation.

5.2.2 Future of the Model

As explained before, a rolling stock plan needs to be checked by local planners, who can determine if the shunting plans are feasible. As it takes a lot of time to push the plan back

and forth between global planners and local planners, it would be nice to be able to determine with more certainty if a shunting plan would be feasible. So more information could be taken into account in order to accomplish this. The model could be extended to optimize over more train lines with more different types of train units, so conflicts between these could be avoided. Also crew scheduling is considered by local planners, so a further step would be to include this as well. As crew scheduling and rolling stock planning are quite related this might be an important next step in developing planning tools.

A big question is how the model described in this report can be used in practice by planners. For this, a practical tool would have to be created based on the model. But it would also be important to make the model more robust, to make sure that the model works for many different scenarios. For example, it was described earlier that the fixed IP method could fix too much variables and generate an infeasible problem in some cases – this would be unacceptable in a practical tool. Furthermore, the data needed as input for the model needs to be readily available, so this process needs to be automated.

The additions described in this report aim to make the model suitable for the operational planning phase. A next step would be to make the model useful in the real time planning phase. Here speed is even more essential, and a very limited amount of changes can be made. For a lot of different problems that emerge during real time planning, for example delays that occur due to blockades on the tracks, standard strategies exist that describe how these situations need to be dealt with. These strategies might need to be incorporated into the model as well. When problems occur in the real time phase the demands of the crew usually become more important than the 'demands' of the rolling stock. So it would be important to integrate crew scheduling and rolling stock planning in one model.

A different development direction for the model is to use it as a basis for a stochastic model: if there are different scenarios possible, one would like to find a solution that works well for all those scenarios.

Appendix A

Notation

A.1 Basic Composition Model

Basic notation

Starter	trip with no predecessor trip.
Finisher	trip with no successor trip.
$\mathcal{M}$	set of unit types.
n_m	# units of type m available.
c_m	# carriages in units of type m.
$\mathcal{C}$	set of service classes.
$k_{m,c}$	number of seats available of class c in units of type m.
$\mathcal{S}$	set of stations.
$i^0_{s,m}$	preferred initial inventory of units of type m at station s.
$i^\infty_{s,m}$	preferred final inventory of units of type m at station s.
$\mathcal{T}$	set of trips.
$\rho(t)$	Reallocation time after a unit is sent to the shunting yard.
$s_d(t)$	departure station of trip t.
$s_a(t)$	arrival station of trip t.
$\tau_d(t)$	departure time of trip t.
$\tau_a(t)$	arrival time of trip t.
d_t	length of trip t in kilometers.
$\delta_{t,c}$	passenger demand for class c on trip t.
$\sigma(t)$	successor trip of trip t if there is exactly one successor trip.
$\sigma^1(t)$	the first departing successor trip of trip t if t has two successors.
$\sigma^2 t)$	the last departing successor trip of trip t if t has two successors.

$\mathcal{T}_0$	the set of trips with no predecessor trip, the starters.
$\mathcal{T}_\infty$	the set of trips with no successor trip, the finishers.
composition	an ordered set of units from $\mathcal{M}$.
$\|p\|$	the number of units in composition p.
$\nu(p)_m$	the number of units of type $m \in \mathcal{M}$ in composition p.
$\nu(p)$	a vector describing the number of units of all types in composition p.
$\mathcal{P}_t$	the set of compositions allowed on trip t.
$\mathcal{G}_t$	set of possible transitions between trip t and its successor $\sigma(t)$.
$c_m(p, p')$	number of units of type m coupled to the train during composition change from p to p'.
$u_m(p, p')$	number of units of type m uncoupled from the train during composition change from p to p'.
$s_{t,p,c}$	seat shortages for passenger class c on trip t with composition p.
Transition Graph	graph representing possible transitions between trips.

Variables

$X_{t,p}$	whether trip t has composition p.
$Z_{t,p,p'}$	transition between t and $\sigma(t)$: whether trip t has composition p and trip $\sigma(t)$ has composition p'.
$N_{t,m}$	the number of units of type m in trip t.
$C_{t,m}$	the number of units of type m that are coupled to the train right before it starts trip t.
$U_{t,m}$	the number of units of type m that are uncoupled from the train right after it has completed trip t.
$I_{t,m}$	the inventory of units of type m at station $s_d(t)$ right after the departure of trip t.
$I^0_{s,m}$	the number of units of type m stored at station s at the start of the day.
$I^\infty_{s,m}$	the number of units of type m stored at station s at the end of the day.
CKM	the number of carriage kilometers.
SKM	seat shortage kilometers.
CCH	total number of shunting movements.

Additional notation

$\mathcal{B}_t$	a set of vectors $\nu(p)$ describing the number of units per type for all possible compositions.
$Y_{t,b}$	whether the number of units specified in $b \in \mathcal{B}_t$ is used in trip t.

$\mathcal{T}^s$	subset of trips where the train is split into two parts.
$\mathcal{T}^c$	subset of trips where the train is combined with another train.
$\sigma^1(t)$	the first departing successor trip of $t \in \mathcal{T}^s$.
$\sigma^2(t)$	the second departing successor trip of $t \in \mathcal{T}^s$.
$\mathcal{G}_t^s$	a triple of compositions describing splitting.
$\mathcal{G}_t^c$	a triple of compositions describing combining.
Z^s_{t,p,p_1,p_2}	describes whether composition change $(p, p_1, p_2) \in \mathcal{G}_t^s$ is used.
Z^c_{t,p,p_1,p_2}	describes whether composition change $(p, p_1, p_2) \in \mathcal{G}_t^c$ is used.

α_i^L	the number of units uncoupled from the left after trip t_i.
α_i^R	the number of units uncoupled from the right after trip t_i.
β_i^L	the number of units coupled to the left after trip t_i.
β_i^R	the number of units coupled to the right after trip t_i.
γ_i	the number of units in trip t_i.

A.2 Modified Composition Model

Generalized Composition Model

$\mathcal{P}_t$	set of extended compositions.
$p(e)$	composition corresponding to extended composition $b \in \mathcal{P}_t$.
$\mathcal{G}_t$	set of extended transitions.
$p(a)$	composition of trip t where $a \in \mathcal{G}_t$.
$p'(a)$	composition of trip $\sigma(t)$ where $a \in \mathcal{G}_t$.
$X_{t,e}$	whether trip t has extended composition $e \in \mathcal{P}_t$.
$Z_{t,a}$	whether extended transition $a \in \mathcal{G}_t$ is used between trips t and $\sigma(t)$.
$c_m(a)$	the number of units of type m coupled to the train during extended transition a.
$u_m(a)$	the number of units of type m uncoupled from the train during extended transition a.
$c_m^S(e)$	the number of units of type m in composition $p(e)$, used for starters.
$u_m^F(e)$	the number of units of type m in composition $p(e)$, used for finishers.

Extended Shunting Movements

Shunting Index	A code describing what happens to the train units of a trip during a transition.

Combined Inventory

$\mathcal{M}^C$	set of compositions of units allowed in the combined inventory.
$\mathcal{P}_t^{C[S]}$	extended composition indicating that starter t comes from the combined inventory.
$\mathcal{P}_t^{C[F]}$	extended composition indicating that finisher t goes to the combined inventory.
$\mathcal{P}_t^{C[SF]}$	extended composition indicating that trip t comes from the combined inventory and returns to the combined inventory.
$\mathcal{P}_t^{N}$	extended composition indicating a 'normal' composition.
$\mathcal{G}_t^{C[u]}$	extended transition indicating that units uncoupled from trip t go to the combined inventory.
$\mathcal{G}_t^{C[c]}$	extended transition indicating that units coupled to trip t come from the combined inventory.
$\mathcal{G}_t^{C[cu]}$	extended transition indicating that units uncoupled from trip t go to the combined inventory and units coupled to t come from the combined inventory.
$\mathcal{G}_t^{N}$	extended transition indicating a 'normal' transition.
$z^{C[c]}(t)$	decision variable indicating whether units coupled to trip t come from the combined inventory.
$z^{C[u]}(t)$	decision variable indicating whether units uncoupled from trip t go to the combined inventory.
$\mathcal{P}_t^{C[S]'}$	extended compositions for starters that need to be penalized.
$\mathcal{P}_t^{C[F]'}$	extended compositions for finishers that need to be penalized.
$\mathcal{G}_t^{C[u]'}$	extended transitions that need to be penalized.
$\mathcal{G}_t^{C[c]'}$	extended transitions that need to be penalized.

Fast Shunting Movements

$\mathcal{L}$	set of pairs of trips between which a lock can occur.
$\mathcal{P}_t^{L}$	extended compositions for starters or finishers where a lock can occur.
$\mathcal{G}_t^{L}$	extended transitions where a lock can occur.
$c_p^L(a)$	whether ordered composition p is coupled to the train during extended transition a.
$u_p^L(a)$	whether ordered composition p is uncoupled from the train during extended transition a.
$c_p^{L[S]}(e)$	whether ordered composition p is equal to extended composition e.

$u_p^{L[F]}(e)$	whether ordered composition p is equal to extended composition e.
$z^L(t_1, t_2)$	whether a lock is used for $(t_1, t_2) \in \mathcal{L}$.

Continuity Constraint

$G_t(p_1, p_2)$	the set of transitions where the unit in position p_1 in trip t has position p_2 in trip $\sigma(t)$.
D_{t,p_1,p_2}	the amount of flow between the unit in position p_1 in trip t and the unit in position p_2 in trip $\sigma(t)$.

Appendix B

Programming Issues

In this section several issues are described that are not directly related to the composition model, but that are relevant in the implementation of the model.

Section B.1 describes how to redefine data structures in order to reduce the number of iterations needed in loops. This greatly reduces the preprocessing time of the model. In section B.2 is described how the usages of a sort of pointer can save several hundreds of MB's of memory. Finally, section B.3 describes some input errors that were observed in original plans, and that need to be taken into account.

B.1 Simplifying Loops

One time consuming factor in the preprocessing time is big loops, some loops iterate hundreds of millions of times. By using convenient ways to split up big arrays with data, often these big loops can be reduced in size dramatically leading a to much faster preprocessing time. In this section an example is given of how such a big loop could be dealt with.

One big array used in the model is the `transition` array. This array contains all possible transitions for all trips, in total several tens of thousands of possibilities. Now suppose one wants to calculate the number of units uncoupled from a train at a specific station s before trip t. In pseudo code this could be calculated by:

```
sum (trans in transition:
                trip1 in trans arrives at s &
                arrival time of trip1 in trans < arrival time of t)
        Z[trans]
```

Basically this code checks every possible transition, 30.000 transitions for the koploper units, and sums over a subset. A way to improve this code is to 'split' `transition` into `transition_trip`

and `transition_composition`. Here `transition_trip` consists of a set of two trips between which a transition takes place, and `transition_composition` describes all possible transitions between two compositions, not directly related to specific trips. In order to keep the link between trips and composition changes, for every pair of trips between which a transition occurs a list of possible composition changes is created, `allowed_change`. Now the code above can be modified to:

```
sum (trans in transition_trip:
        trip1 in trans arrives at s &
        arrival time of trip1 in trans < arrival time t)
                sum (transcomp in transition_composition:
                        transcomp is allowed for trans)
                        Z[transcomp]
```

Now only 500 transitions are checked, and only for a few of those all possible composition changes are considered. Suppose that there are 10 relevant transitions and that there are 300 possible composition changes, then in total 3.000 instead of 30.000 iterations are done. When considering a more complex loop than this simple example, which also occurs in the implementation of the composition model, much bigger improvements can be made.

The example presented above was already implemented in the version of the model that was used as the basic model in this report, several similar additions were made. For a part of the implementation an improvement was made that reduced the time needed to run a loop from 10 minutes to only a few seconds.

B.2 Reducing Memory Usage

During the preprocessing time where the constraints and the variables are created, a lot of memory was used. After some investigation, it turned out that a major part of this memory usage was due to a lot of information about trips being saved for every object that used those trips. For example, in the array that contains a description of the transitions, two copies of the trips were created for every instance.

In order to save memory, an ID was introduced for the trips, so that instead of saving the entire trip a lot of times only a reference to the trip in the form of an ID is stored, similar to the usage of pointers. Using this technique several hundreds of MB were saved. Also less time is needed to build the model this way, since the data structures are smaller.

B.3 Input Errors

The original plan that serves as input for the model is the result of an even earlier created plan that was changed over time and by different people. Although not frequently, some errors can occur in the original plan which need to be dealt with before the plan is used in the model. In this section some examples are given of errors that were observed.

- During the detection of exceptional shunting movements sometimes errors in the input file were observed. For example, sometimes the order of the units in a train is changed at a station while the train continues in the same direction. It is difficult to detect automatically whether an abnormal shunting movement is an exceptional shunting movement or an error, therefore some human intervention might be needed here.
- In the input file of train duties an error was observed where a train unit arrives at one station and then departs from another station.
- The expected number of passengers is sometimes not available.

When real errors are observed in an original plan, feedback from the designers of the original plan could be needed to determine how the errors need to be dealt with. In any case, it is important to take the possibility for errors in the input into account.

Bibliography

[1] G. Maróti, *Operations Research Models for Railway Rolling Stock Planning*, PhD thesis, Technische Universiteit Eindhoven, 2006.

[2] D. Ryan and M. Osborne, *On the solution of highly degenerate linear programmes*, Mathematical Programming, 41 (1988), pp. 385–392.

[3] P. Wolfe, *A technique for resolving degeneracy in linear programming*, Journal of the Society for Industrial and Applied Mathematics, 11 (1963), pp. 205–211.

[4] L. A. Wolsey, *Integer Programming*, Wiley-Interscience, 1998.