TRANSPORTATION SCIENCE
Vol. 40, No. 3, August 2006, pp. 378–391
ISSN 0041-1655 | EISSN 1526-5447 | 06 | 4003 | 0378

DOI 10.1287/trsc.1060.0155

Efficient Circulation of Railway Rolling Stock

Arianna Alfieri
Politecnico di Torino, Dipt. Sistemi di Produzione ed Economia, Corso Duca degli Abruzzi 24, IT-10129, Torino, Italy, arianna.alfieri@polito.it

Rutger Groot
ORTEC bv Consultants, P.O. Box 490, NL-2800 AL Gouda, The Netherlands, rgroot@ortec.nl

Leo Kroon
NS Reizigers, Department of Logistics, P.O. Box 2025, NL-3500 HA Utrecht, The Netherlands, and Rotterdam School of Management, Erasmus University Rotterdam, P.O. Box 1738, NL-3000 DR Rotterdam, The Netherlands, lkroon@rsm.nl

Alexander Schrijver
Centrum voor Wiskunde en Informatica, P.O. Box 94079, NL-1090 GB Amsterdam, The Netherlands, lex.schrijver@cwi.nl

Railway rolling stock is one of the most significant cost components for operators of passenger trains. The *efficient* circulation of rolling stock is therefore one of the main objectives pursued in practice. This paper focuses on the determination of appropriate *numbers* of train units of different types together with their efficient *circulation* on a single line. To utilize the train units on this line in an efficient way, they are *coupled* to or *uncoupled* from the trains in certain stations according to the passengers' seat demand in peak and off-peak hours. Because coupling and uncoupling train units must respect specific rules related to the shunting possibilities in the stations, it is important to take into account the *order* of the train units in the trains. This aspect strongly increases the complexity of the rolling stock circulation problem. This paper presents a solution approach based on an integer multicommodity flow model with several additional constraints related to the shunting processes at the stations. The approach is applied to a real-life case study based on the timetable of NS Reizigers, the main Dutch operator of passenger trains.

Key words: railway transportation; circulation of rolling stock; multicommodity flow model
History: Received: November 2004; revision received: January 2006; accepted: February 2006.

1. Introduction

The efficient circulation of railway rolling stock is an important consideration for operators of passenger trains, because the rolling stock is one of their most significant cost components. The involved costs are mainly due to acquisition, power supply, and maintenance of the rolling stock. Because these costs are usually substantial, it must be decided carefully how much rolling stock is necessary per scheduled train to provide quality service to passengers: an efficient circulation of rolling stock requires per trip a match between the provided rolling stock capacity and the passengers' demand for transportation.

To achieve this objective, the compositions of the trains may have to be changed during the operations by coupling or uncoupling rolling stock to or from the trains. For example, rolling stock may be uncoupled from the trains after the morning peak hours, and it may be coupled again before the afternoon peak hours. In these *shunting* processes, which are usually carried out in the short time interval between the arrival of a train at a station and its subsequent departure, several practical rules related to the feasibility of the transitions of the train compositions should be considered. For example, train units can only be uncoupled from the rear end of a leaving train.

This paper focuses on the determination of appropriate *numbers* of *train units* of different types together with their efficient *circulation* on a single line of NS Reizigers, the main Dutch operator of passenger trains. Thereby the positions of the train units in the trains are taken into account because these determine the feasibility of the transitions from one train composition into another.

This paper is organized as follows. A detailed problem description is given in §2. Section 3 gives an overview of earlier research in the area of rolling stock circulation. Section 4 describes two models for solving our rolling stock circulation problem. This section also describes the computational complexity of two variants of the problem. Section 5 presents our solution approach based on a combination of an integer multicommodity flow model and finding paths in so-called *transition graphs*. The results of our computational experiments are presented in §6. Finally, conclusions are drawn in §7.

2. Problem Description

2.1. Rolling Stock: Train Units

NS Reizigers has a large variety of rolling stock available for its passenger transportation process. The major part of the rolling stock consists of *train units*. Each train unit consists of a certain number of carriages that cannot be split from each other during the daily operations. A main difference between a train unit and a single *carriage* is that each train unit has its own engines. As a consequence, a train unit can move individually in both directions without a locomotive. This is in contrast with a single carriage, which needs a locomotive for any movement. Figure 1 shows an example of a single-deck train unit consisting of three carriages.

Single-deck train units can be subdivided into train units with three or four carriages. In this paper, we focus on the circulation of such single-deck train units. Each train unit has fixed capacities of first- and second-class seats. These capacities are more or less linear in the number of carriages per train unit. Single-deck train units with three or four carriages may be combined with each other into one longer train. The latter implies that for each type, not only the number of train units in each train is to be determined, but also their order in the train. This is explained later in this section.

An ordered sequence of train units in a train is called a *composition*. For example, if "3" and "4" denote single-deck train units with three and four carriages, respectively, then "344" and "434" indicate different compositions. The notation here is such that both compositions have a train unit of type "4" in front (in other words, the trains move from left to right).

Coupling train units onto a train or uncoupling train units from a train has to be carried out in the short period (typically just a few minutes) between a train's arrival at a station, and its subsequent departure from that station. To minimize the time required for the involved shunting movements, train units are usually coupled onto the *front* end of an incoming train, and uncoupled from the *rear* end of a leaving train. Only if the difference between the departure time and the arrival time of a train is sufficiently large, more complicated shunting movements may be carried out.

From a capacity point of view, the earlier mentioned compositions "344" and "434" are identical. However, these compositions have different transition possibilities: the train unit of type "3" can be uncoupled easily from the composition "344," whereas the latter is not the case for the composition "434." In case of the composition "344," the train unit of type "3" can be uncoupled from the train, and the remaining part of the train (with composition "44") can proceed without being disturbed by the uncoupled train unit. Recall that trains are assumed to move from left to right. In case of the composition "434," uncoupling the train unit of type "3" would require several shunting operations, which usually requires too much time. In this case, uncoupling the rear train unit of type "4" would also be easy, but if one really wants to uncouple the train unit of type "3," then uncoupling the train unit of type "4" may conflict with the train's forecasted passenger demand later on.

Because train units can be subdivided into different types significantly complicates both the planning process and the daily operations. However, the *gain* is that, per train, a better match between the expected number of passengers (demand) and the provided number of seats (supply) can be realized. For example, train units with a length of 4 carriages can compose only trains with 4, 8, or 12 carriages. However, a combination of train units with 3 and 4 carriages may give rise to trains with 3, 4, 6, 7, 8, 9, 10, 11, and 12 carriages. Note that each train should not be longer than the shortest platform of the stations along the train's route, and certainly not longer than 12 carriages.

2.2. Further Details of the Problem

The planning of the rolling stock circulation usually starts after the timetable has been completed, because the timetable is required as input for the rolling stock circulation planning. In this paper, the timetable is assumed to be cyclic.

A line is a direct connection between two end stations that is served with a certain frequency (e.g., once or twice per hour). Each line is served by a fixed number of *trains*. During the day, each train runs up and down between the end stations of the line. The number of trains on a line is determined by the line's circulation time including the return times at the end stations, and the line's frequency. This number equals the number of trains that can be observed on a picture of the line that is taken from above at any moment during the day.

Rolling stock is usually dedicated to its own line to prohibit delays of trains to spread over wider parts

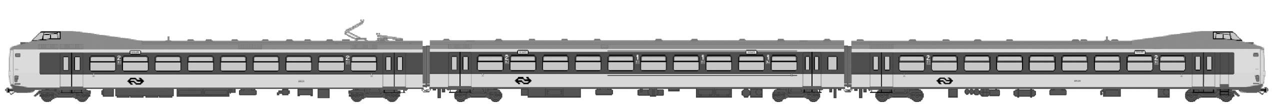

Figure 1 A Single-Deck Train Unit with Three Carriages

of the network. This fact justifies to study the rolling stock circulation problem for a single line, as in the current paper.

The timetable is given in the form of a set of trips. A trip represents the movement of a train between two stations in which the composition of the train can be changed. Each trip has a start time, end time, origin station, and destination station. For each trip, the corresponding train is also known. Conversely, each train corresponds to an ordered sequence of trips carried out by this train. In particular, for each trip, the next trip of the corresponding train is also known a priori.

Other input consists of the forecasts of the required seat capacities per trip. For collecting this input, the passengers are counted by the conductors. The translation of these counts into the minimally required first- and second-class capacities per trip is based on a statistical procedure, which falls outside the scope of this paper. In the operations, the number of passengers on a trip has a *stochastic* character, and NS Reizigers does not use a *seat reservation* system, therefore, it is impossible to *guarantee* a seat for all passengers, especially during peak hours. However, during nonpeak hours, the rolling stock capacity is usually sufficient to provide all passengers with a seat.

Fixed rolling stock costs are related to acquisition and depreciation. Variable rolling stock costs are related to power supply and maintenance: after a certain number of kilometers, each train unit is routed to a maintenance station for a preventive check-up and possibly for a repair. Note that, in the Netherlands, routing train units to a maintenance facility is carried out in the *operations* (Maróti and Kroon 2005). Therefore, maintenance routing can be ignored in our rolling stock circulation *planning* problem.

Because of the timetable for a single line and the corresponding passengers' demand forecasts, the problem is to find appropriate numbers of train units of the different types, together with such a circulation of these train units that (i) all forecasted passengers can be seated, (ii) all practical constraints concerning the maximum lengths of the trains and the transitions of the train compositions are respected, and (iii) the relevant objective consisting of fixed and/or variable costs is minimized. Note that in some trains, the allocated capacity may be larger than the capacity required by the demand. This is due to the demand on subsequent trips of the train, or for relocating train units.

2.3. Case Study: The Line 3000

The model and solution approach described later in this paper are illustrated with a case study based on the line 3000, one of the intercity lines of NS Reizigers. This line provides twice per hour an intercity connection from Den Helder (Hdr) to Nijmegen (Nm) and vice versa (see Figure 2 and www.ns.nl).

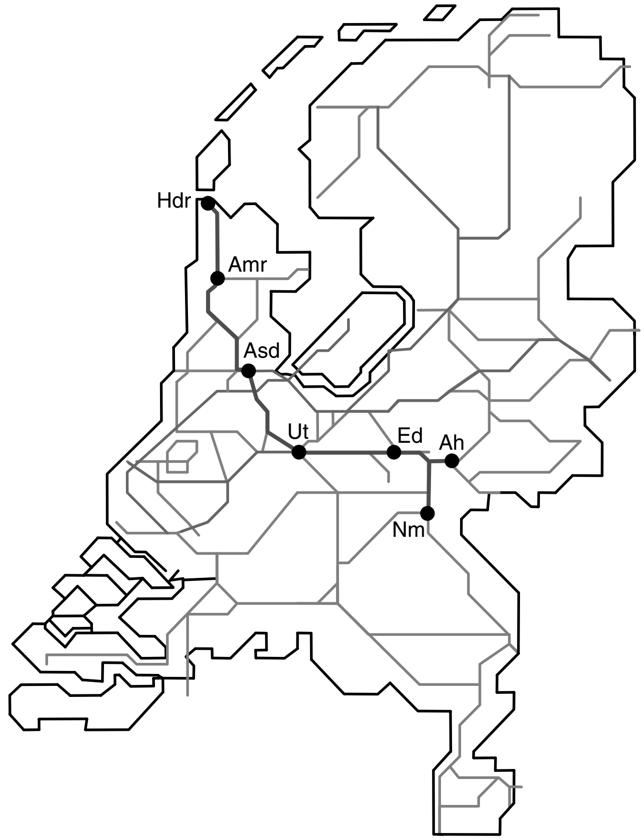

Figure 2 The Line 3000 Den Helder (Hdr)–Nijmegen (Nm)

The timetable of the line 3000 is cyclic with a cycle length of 30 minutes. During early morning and late evening, some exceptions occur in the cyclic timetable. For example, in the early morning, some trains start in Alkmaar (Amr), Amsterdam (Asd), Utrecht (Ut), and Arnhem (Ah). Similar exceptions exist in the late evening. In principle, the compositions of the trains can be changed only in Alkmaar and Nijmegen. In late evening, there are again some exceptions from this rule.

The line 3000 is operated by 12 trains. This occurs because the circulation time on the line between Den Helder and Nijmegen and vice versa is six hours and because two trips occur per hour in each direction. Hence, every twelfth departure from, say, Nijmegen, can be covered by the same train.

Figure 3 shows a time-space diagram for part of the trips of this line. The numbers at the top and bottom of the figure indicate the time axis. The grey diagonal lines indicate the trips, and the adjacent numbers are the corresponding train numbers. For simplicity, the compositions of the trains have been indicated for only some of the trains. Train units of type "3" are represented by dashed lines. Train units of type "4" are represented by solid lines. The train unit at the front of a train is represented by the rightmost line of a composition.

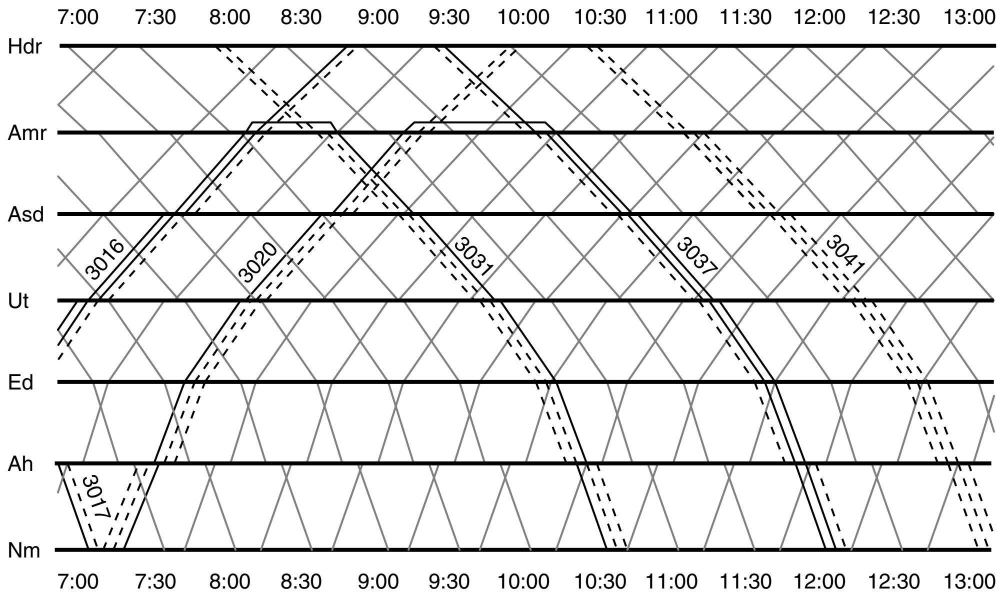

Figure 3 **Part of Rolling Stock Circulation for the Line 3000**

For example, the train on trip 3017 is operated with composition "43," where the train unit of type "3" is the front unit. This train arrives in Nijmegen at 7:11, where an additional train unit of type "3" is coupled. Nijmegen is an end station, where the front end and the rear end of the train are exchanged. The coupled train unit becomes the front end of the incoming train, which is the rear end of the outgoing train. The train leaves at 7:20 from Nijmegen on trip 3020 with composition "334," and arrives in Arnhem at 7:33.

In Arnhem, the physical composition of a train cannot be changed, but the front and rear ends of a train are exchanged, as is shown in Figure 3. The latter is because each train leaves the station Arnhem from the same side that it entered ("kopmaken" in Dutch). The train leaves Arnhem at 7:38 with composition "433."

The train arrives in Alkmaar at 9:19. Here, the train unit of type "4" at the rear end of the train is uncoupled and remains in Alkmaar, because of the lower expected passengers' demand north of Alkmaar. The remaining train continues at 9:21 to Den Helder with composition "33." As a consequence, the train unit of type "4" should be uncoupled within two minutes, which prohibits a complex shunting process. The remaining train arrives at 9:56 in Den Helder. At 10:33, it returns to Nijmegen on trip 3041, again with composition "33." This process continues until the end of the day.

Note that in Den Helder, there is also a departure already at 10:03 (see Figure 3). Thus, the train that arrived in Den Helder at 9:56 might have returned already at that time. However, because of the robustness of the circulation, all trains have a standstill at Den Helder of at least 30 minutes. Obviously, the price that has to be paid for this robustness is the additional train required for the rolling stock circulation.

Train units uncoupled from a certain train can be coupled onto another train later on. For example, Figure 3 shows how the train unit uncoupled from the rear end of the train on trip 3020 at 9:19 in Alkmaar is coupled onto the front end of another train on trip 3037 at 10:12. Note that in Alkmaar, it is undesirable to couple a train unit onto a northbound train, nor is it desirable to uncouple a train unit from a southbound train.

3. Related Literature

In the literature, several rolling stock circulation problems have been described, which shows the seriousness of these problems. Most of the literature focuses on the circulation of *locomotive hauled carriages* and not on the circulation of *train units*. However, the following three papers deal with the circulation of train units for passenger transportation.

Schrijver (1993) considers the problem of minimizing the number of train units of different types for an hourly train line in the Netherlands, given that the passengers' seat demand must be satisfied. The only restriction on the transition between two compositions on two consecutive trips is that the required train units must be available at the right time and station. (Un)coupling restrictions related to the feasibility of the shunting movements are ignored.

Ben-Khedher et al. (1998) study the problem of allocating train units to the French high-speed trains. Their rolling stock allocation system is based on a capacity adjustment model linked to the seat reservation system. This system aims at maximizing the expected profit. Because these train units are identical, shunting restrictions are less important in this paper.

Abbink et al. (2004) present a model to allocate different rolling stock families and types to different train lines. They present an integer programming model, that minimizes the seat shortages during the morning peak hours by allocating rolling stock families and types with different capacities to all the trains running simultaneously at 8:00 A.M., the busiest moment of the day. Their approach is applied to several scenarios of NS Reizigers that differ in the numbers of rolling stock families and types that can be allocated to a line.

Papers dealing with the efficient circulation of *locomotive hauled carriages*, possibly in combination with the required locomotives, are the following. Brucker, Hurink, and Rolfes (2003) study the problem of finding a circulation of railway carriages through a rail network, given a timetable. Because the required train compositions have been specified a priori, this paper focuses on finding appropriate repositioning trips of carriages from one station to another. Their solution approach is based on local search techniques such as simulated annealing.

Van Montfort (1997) also focuses on the efficient circulation of railway carriages. He studies the assignment of carriages to trains, given a cyclic timetable and a *core standard structure* for train compositions on a combination of lines in the Netherlands. Van Montfort uses an integer programming model improved by the application of several types of valid inequalities.

Cordeau, Soumis, and Desrosiers (2000) present a Benders decomposition approach for the locomotive and car assignment problem. Their approach is based on the concept of a *train consists*, i.e., a group of compatible units of rolling stock (locomotive(s), first- and second-class carriages) that travel along some part of a rail network. Computational experiments show that optimal solutions can be found in short computation times by applying column-generation techniques. In a subsequent paper, Cordeau, Soumis, and Desrosiers (2001) extend their model with various practical constraints, for example, dealing with maintenance of the rolling stock.

Lingaya et al. (2002) study the problem of assigning carriages to trains at VIA Rail in Canada. They present a complex model to adapt a master plan to additional information concerning the expected numbers of passengers. They allow for coupling and uncoupling of carriages at various locations in the network and explicitly consider the order of the carriages in the trains. Several other real-life constraints, such as maintenance requirements, are considered as well. The solution approach is based on a Dantzig-Wolfe reformulation solved by column-generation techniques. Next, a branch-and-bound procedure is applied heuristically to obtain good integer solutions.

The problem described in this paper is different from the problems in the above-mentioned papers. First, this paper deals with train units instead of locomotive-hauled carriages. Shunting restrictions for train units are different from those for locomotive-hauled carriages: Due to their shorter shunting time, train units are far more flexible than locomotive-hauled carriages. Second, the current paper deals with train units of different types, which implies that the detailed orders of the train units in the trains are relevant. Further details of our problem and solution approach can be found in Groot (1996).

Historically, there was not much cooperation and coordination between different railway operators. Therefore, most operators have their own types of rolling stock with their own peculiarities. Also, the rules used in setting up a rolling stock circulation differ from operator to operator. As a result, there is a low probability that research in this area carried out for one operator is directly applicable for another one. Obviously, there is a need for a generic approach in solving rolling stock circulation problems. However, to get there, a number of specific cases first have to be studied and described.

4. Model Formulation

This section describes the models developed for solving the rolling stock circulation problem. In §4.1, we give some definitions and notations. In §4.2, we describe a model that neglects the compositions of the trains. Thereafter, in §4.3, we present a model that takes into account the compositions of the trains. In §4.4, we describe the computational complexity of two variants of the rolling stock circulation problem.

4.1. Definitions and Notation

We assume to have a single line connecting two end stations with a given frequency. The timetable on this line is cyclic. The fixed number of trains operated on this line depends on the line's circulation time and frequency. All trains are assumed to be operated by train units of different possible types. The latter implies that the problem is a complex variant of an integer multicommodity flow problem. In particular, the problem could be called an *ordered* integer multicommodity flow problem, because the order of the train units in the trains is as important as their number.

In this paper, we use the following notation. First, we have a set T of trips, a set S of stations, and a set J of types of train units. Each trip $t \in T$ is represented by an origin station O_t, a destination station D_t, a start time S_t, and an end time E_t. For each train τ, the set T_τ denotes the ordered set of trips that is operated by train τ. In particular, for each trip t, the next trip $n(t)$ carried out by the same train also is known. The parameter L_t represents the length of trip t. Furthermore, the parameter $d_{c,t}$ $(c = 1, 2)$ denotes the expected number of passengers in class c on trip t. The set T_t^a is the subset of trips t' arriving in station O_t before the departure of trip t from O_t. That is, $T_t^a = \{t' \in T \mid D_{t'} = O_t, E_{t'} < S_t\}$. Similarly, T_t^d is the subset of trips t' departing from station O_t before the departure of trip t. Thus, $T_t^d = \{t' \in T \mid O_{t'} = O_t, S_{t'} < S_t\}$. The length of the shortest platform along trip t is denoted by P_t. The length and number of carriages of each train unit of type j are denoted by l_j and N_j, respectively. Finally, $C_{j,c}$ represents the capacity in class c $(c = 1, 2)$ of each train unit of type j, and F_j and V_j denote the fixed and variable costs of each train unit of type j, respectively.

4.2. Model 1: Neglecting the Compositions

If the order of the train units in the trains is neglected, then the rolling stock circulation problem can be represented by an integer multicommodity flow model with several additional constraints. This model is represented by a *flow graph* (also called a *time-space graph*, see Figure 4), whose nodes correspond to events (an arrival or departure of a train in a station) and whose arcs are connections between two different events.

An arc is a *trip arc* if the two connected nodes belong to different stations (represented by dashed lines in Figure 4), and it is a *station arc* if the two events are consecutive events of the same station (represented by solid lines). A station arc represents a connection between an arrival (or departure) in a station and the next arrival (or departure) in the same station.

In each node of the flow graph we have to guarantee the flow balance. In Figure 4, for example, the number of train units in Alkmaar (Amr) before the arrival of the train on trip 3031 plus the number of train units arriving in the train on trip 3031 equals the number of train units leaving from Alkmaar in the train on trip 3031 plus the number of train units in Alkmaar after the departure of the train on trip 3031. This relation holds separately for each train unit type.

The model is expressed in terms of the decision variables $x_{t,j}$ indicating the number of train units of type j that is allocated to trip t. Additional decision variables are the variables $y^0_{s,j}$ denoting the number of train units of type j stored in station s during the night, and y_j^{tot} denoting the total number of available train units of type j.

Now Model 1 (neglecting the compositions) can be stated as follows:

$$\min F(x, y)$$

subject to

$$y_j^{\text{tot}} = \sum_s y^0_{s,j} \quad \forall j \in J \tag{1}$$

$$x_{t,j} \le y^0_{s,j} + \sum_{t' \in T_t^a} x_{t',j} - \sum_{t' \in T_t^d} x_{t',j} \quad \forall j \in J;\ t \in T,\ s = O_t \tag{2}$$

$$\sum_j l_j x_{t,j} \le P_t \quad \forall t \in T \tag{3}$$

$$\sum_j C_{j,c} x_{t,j} \ge d_{c,t} \quad \forall t \in T;\ c = 1, 2 \tag{4}$$

$$x_{n(t),j} \le x_{t,j} \quad \forall t\colon O_t = \text{Ah},\ D_t = \text{Amr};\ j \in J \tag{5}$$

$$x_{n(t),j} \ge x_{t,j} \quad \forall t\colon O_t = \text{Hdr},\ D_t = \text{Amr};\ j \in J \tag{6}$$

$$x_{t,j} \in \mathcal{Z}^+ \quad \forall t \in T,\ j \in J \tag{7}$$

$$y_j^{\text{tot}}, y^0_{s,j} \in \mathcal{Z}^+ \quad \forall s \in S,\ j \in J. \tag{8}$$

In this model, $F(x, y)$ represents one of the following objective functions:

1. PB1: minimize the fixed costs of the train units ($\min \sum_j F_j y_j^{\text{tot}}$).
2. PB2: minimize the variable costs of the train units ($\min \sum_t L_t(\sum_j V_j x_{t,j})$).

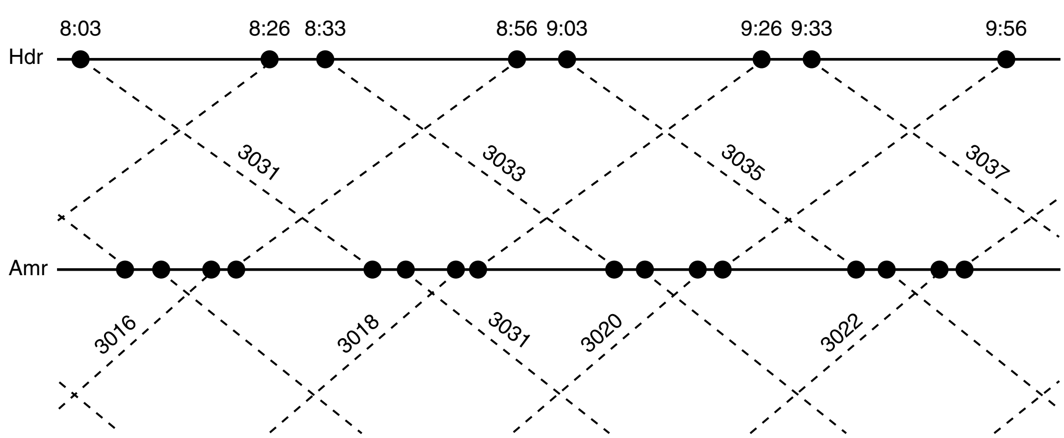

Figure 4 Part of the Flow Graph for the Line 3000

3. PB3: minimize the variable costs of the train units taking into account an upper bound on the number of carriages ($\min \sum_t L_t(\sum_j V_j x_{t,j})$ subject to $\sum_j N_j y_j^{\text{tot}} \le UB$).

The total number of required train units of each type equals the total number of train units $y_{s,j}^0$ that stay in the various stations during the night, as is represented by constraints (1). Constraints (2) describe that the number of train units of type j that is allocated to trip t should not exceed the number of train units of this type available in station O_t just before the start time of trip t. The latter equals the number of such train units by the start of the day plus the number of train units that have arrived in this station until this time instant, minus the number of train units that have departed from there until then. Note that it is not difficult to consider a certain minimum reassignment time between the uncoupling of a train unit from a train and the subsequent coupling of this train unit onto another train. However, such a reassignment time has been omitted here. On each trip t, a train should not be longer than the length of the shortest platform P_t along the trip. The latter is guaranteed by constraints (3). Constraints (4) are the demand satisfaction constraints for first- and second-class seat demands on each trip. Constraints (5) and (6) describe that in Alkmaar, train units cannot be coupled onto a northbound train nor uncoupled from a southbound train. Finally, constraints (7) and (8) specify the integer character of the decision variables. The domains of these decision variables can be reduced a priori by taking into account, e.g., the maximum train length per trip or the maximum storage capacity per station.

4.2.1. Valid Inequalities. This section describes how the constraints on the maximum train length (3) and demand satisfaction (4) can be made more tight by adding certain *valid inequalities*. Note that these valid inequalities were described already by Schrijver (1993). For example, the model may contain the following constraints (9) and (10) for a certain trip t:

$$3x_{t,3} + 4x_{t,4} \le 12 \tag{9}$$

$$166x_{t,3} + 224x_{t,4} \ge 510. \tag{10}$$

These constraints represent that a train should have a length of at most 12 carriages, and that the second-class seat demand is to be satisfied. Here, the number 510 equals the required number of second-class seats on trip t, and the numbers 166 and 224 represent the second-class capacities of train units with three and four carriages, respectively. Due to the integrality of the variables $x_{t,3}$ and $x_{t,4}$, constraint (10) can be sharpened as follows:

$$x_{t,3} + 2x_{t,4} \ge 4 \tag{11}$$

$$x_{t,3} + x_{t,4} \ge 3. \tag{12}$$

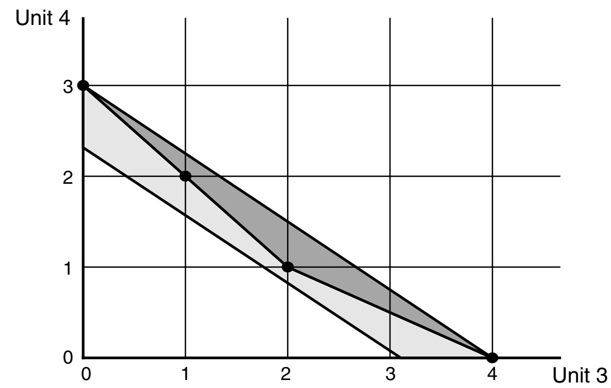

Figure 5 **Reduced Feasible Region**

The above example is shown in Figure 5. Here, the grey and dark areas correspond to the continuous feasible region for the train composition on trip t obtained by considering constraints (9) and (10). The black dots represent the feasible combinations of the two types of train units (Units 3 and 4, respectively). In this example, the feasible combinations are $(0,3)$, $(1,2)$, $(2,1)$, and $(4,0)$. The dark area represents the convex hull of the locally feasible region.

We certainly do not claim that valid inequalities such as (11) and (12) give a complete description of the convex hull of the integer feasible region of the complete problem. Nevertheless, the improved local description of the convex hull turned out to give an improved performance of our solution approach in many cases (see the computational results in §6).

4.3. Model 2: Taking into Account the Compositions

To also take into account the compositions of the trains, the following definitions are used. First, the set of all feasible compositions is denoted by K. Recall that a *composition* is an ordered sequence of train units in a train. The set of compositions feasible for trip t is denoted by K_t. This set is determined based on the required first- and second-class capacities and on the maximum train length for trip t. The parameter $g_{k,j}$ denotes the number of train units of type j in composition k. The feasible transitions from one composition of a train to another are described in the sets $A_{t,k}$ for each trip t and composition k. That is, the set $A_{t,k}$ contains the compositions feasible on trip $n(t)$ if the train has composition k on trip t.

For each train, the feasible compositions per trip and the feasible transitions from one composition to another are represented in a so-called *transition graph* (see Figure 6). The set of nodes of the transition graph of train τ is the set $\bigcup_{t\in T_\tau}\{(t,k) \mid k \in K_t\}$. Here, the union is taken over all trips t that are carried out by train τ. The set of arcs of this graph is the set $\bigcup_{t\in T_\tau}\bigcup_{k\in K_t} A_{t,k}$. Here, the union is again taken over

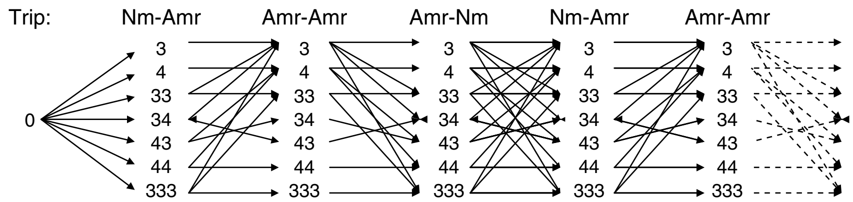

Figure 6 Part of the Transition Graph for One Train of the Line 3000

all trips t that are carried out by train τ and over all feasible compositions $k \in K_t$. Furthermore, each transition graph has a source node, connected to all nodes of the first trip of the corresponding train, and a sink node, connected to all nodes of the last trip of this train.

An example of (part of) a transition graph is shown in Figure 6. To keep the figure as clear as possible, we assume that the maximum train length is nine carriages on all trips. The involved train starts with a trip from Nijmegen to Alkmaar. On this first trip, each of the seven compositions with at most nine carriages can be chosen. The feasible compositions on the second trip depend on the composition chosen on the first trip. The feasible transitions in Alkmaar are denoted by the arcs between the compositions on the trip from Nijmegen to Alkmaar and the trip from Alkmaar to Alkmaar. These transitions represent that the order of the train units in the train was reversed in Arnhem (e.g., composition "34" changes to "43"), and that in Alkmaar, train units can only be *uncoupled* from a northbound train. Note that the second trip runs from Alkmaar to Alkmaar because it is assumed that in Den Helder, no train units are (un)coupled. Therefore, the two train movements from Alkmaar to Den Helder and vice versa have been aggregated into a single trip from Alkmaar to Alkmaar. The feasible transitions between the second and third trip represent that the order of the train units in the train was changed in Den Helder and that in Alkmaar, train units can only be *coupled* onto a southbound train. The transition graph shows that in Nijmegen, train units can be coupled or uncoupled (but not both). The rest of the transition graph can be explained similarly.

In each transition graph, we have to find a path from its source node to its sink node, which altogether minimize the objective function $F(x, y)$. The objective functions used for Model 2 are the same as for Model 1.

Because a transition graph exists for each train, this seems to give a decomposition of the problem across the trains at first sight. However, this is not the case, because the paths through the transition graphs interact with each other via the inventories of the train units in the stations. Train units uncoupled from a train can be coupled onto another one later on, as was shown in Figure 3.

In the model, binary decision variables are associated with the *nodes* in the transition graphs. That is, a decision variable $a_{t,k}$ assumes the value 1 if and only if composition k is selected for trip t. Now Model 2 (taking into account the compositions) can be represented as follows:

$$\min F(x, y) \tag{13}$$

subject to

$$(1),\ (2),\ (5)\text{–}(8) \quad \text{and}$$

$$\sum_{k \in K_t} a_{t,k} = 1 \quad \forall\, t \in T \tag{14}$$

$$a_{t,k} \le \sum_{k' \in A_{t,k}} a_{n(t),k'} \quad \forall\, t \in T,\ k \in K_t \tag{15}$$

$$x_{t,j} = \sum_{k \in K_t} g_{k,j} a_{t,k} \quad \forall\, t \in T,\ j \in J \tag{16}$$

$$a_{t,k} \in \{0, 1\} \quad \forall\, t \in T,\ \forall\, k \in K_t. \tag{17}$$

Note that constraints (3) and (4) of Model 1 do not appear in Model 2, because these are handled now by constraints (14). These constraints represent that for each trip, exactly one feasible composition is to be selected. As stated earlier, the term feasible refers to the required first- and second-class capacities and to the maximum train length. Constraints (14) guarantee that the composition selected for trip t is compatible with the composition selected for the next trip $n(t)$ of the same train. Next, constraints (16) link the flow graph and the transition graphs with each other: For each type, the number of train units on a trip follows directly from the composition that is used on that trip. In fact, these constraints make the flow variables $x_{t,j}$ superfluous because in all occurrences of these variables, they can be removed by substituting (16). If this substitution is applied consequently, then (16) itself becomes superfluous as well. Constraints (17) specify the binary character of the variables $a_{t,k}$.

Note that the model formulation can be tightened by also including the reverse variant of constraints (15):

$$a_{n(t),k} \le \sum_{k' \in K_t:\, k \in A_{t,k'}} a_{t,k'} \quad \forall\, t \in T,\ k \in K_{n(t)}. \tag{18}$$

These constraints guarantee that the composition selected for the successor $n(t)$ of trip t is compatible with the composition selected for trip t itself.

4.4. Computational Complexity

It is well known that integer multicommodity flow problems and integer generalized flow problems are NP-hard in their most general form (Ahuja, Magnanti, and Orlin 1993). However, these results do not yet imply that the rolling stock circulation problem is NP-hard, because in this case, the underlying network has a special structure, and the rolling stock circulation problem involves additional constraints on the coupling and uncoupling of train units.

Therefore, we analyze the computational complexity of the rolling stock circulation problem in this section. Theorem 1 shows that the rolling stock circulation problem is NP-hard. This result is proved for the case that the total fixed costs are to be minimized. For other objectives, a similar result holds. The proof of Theorem 1 applies to the rolling stock circulation problem both if the compositions are neglected and if these are taken into account. Furthermore, Theorem 2 shows that the rolling stock circulation problem can be solved in polynomial time if the numbers of trains, stations, train unit types, as well as the maximum train length are fixed.

Theorem 1 is proved by a reduction from the problem numerical three-dimensional matching (N3DM), which is defined as follows:

Instance: $3n$ different positive integers a_i, b_i, and c_i, and a positive integer d satisfying $d/4 < a_i, b_i, c_i < d/2$ and $\sum_{i=1}^{n}(a_i + b_i + c_i) = nd$.

Question: Do there exist permutations σ and τ of the set $\{1, \ldots, n\}$ such that $a_{\sigma(i)} + b_{\tau(i)} + c_i = d$ for $i = 1, \ldots, n$?

Theorem 1. *Finding a feasible rolling stock circulation with minimum total fixed costs for the train units is NP-hard.*

Proof. Let I be an instance of N3DM as described above. Then the following instance I' of the rolling stock circulation problem is constructed.

The instance I' contains $3n$ trains and $6n$ trips. For $i = 1, \ldots, 3n$, train i carries out a trip from station A to station B in the time interval $(i, i + 3n)$ and a trip back from station B to station A in the time interval $(i + 7n, i + 10n)$.

For $i = 1, \ldots, n$, the first trip of train i has a second-class demand of d and the second trip of this train has a second-class demand of $2d - c_i$. The first-class demand of the latter trips equals 1. For *all* other trips the first-class demand equals 0. The maximum train length of the latter trips equals 3. For *all* other trips, the maximum train length equals 2.

For $i = 1, \ldots, n$, the first trip of train $i + n$ has a second-class demand of $d + a_i$ and the second trip of this train has a second-class demand of d. For $i = 1, \ldots, n$, the first trip of train $i + 2n$ has a second-class demand of $d + b_i$, and the second trip of this train has a second-class demand of d.

There are $2n + 1$ different train unit types. For $i = 1, \ldots, n$, train unit type i has first-class capacity 1 and second-class capacity a_i. For $i = 1, \ldots, n$, train unit type $i + n$ has first-class capacity 0 and second-class capacity b_i. The last train unit type has first-class capacity 0 and second-class capacity d. For each train unit type, the fixed costs are equal to the train unit's second-class capacity and the length of each train unit equals 1.

Now we claim the following: There exists a feasible solution for I if and only if there exists a feasible solution for I' with total fixed costs $4nd - \sum_{i=1}^{n} c_i$.

First, suppose there exists a feasible solution for I' with total fixed costs $4nd - \sum_i c_i$. Note that both in the time interval $(3n, 3n + 1)$ and in the time interval $(10n, 10n + 1)$, the total second-class demand exactly equals the total fixed costs. Due to the fixed cost structure of the train units, it follows that in both time intervals there is an exact match of the second-class demand and the provided second-class capacity.

Due to the maximum train length of 2 in the time interval $(0, 6n)$, each trip with second-class demand $d + a_i$ is covered by a train consisting of a train unit with second-class capacity d and a train unit with second-class capacity a_i. Similarly, each trip with second-class demand $d + b_i$ is covered by a train consisting of a train unit with second-class capacity d and a train unit with second-class capacity b_i. Finally, each trip with second-class demand d is covered by a train consisting of a single train unit with second-class capacity d.

In the time interval $(7n, 13n)$, the same train units are assigned to the trips running in this interval. Each of the $2n$ trips with second-class demand d is covered by a single train unit with second-class capacity d, due to the maximum train length of 2 on these trips.

The remaining train units are assigned to the trips with second-class demand $2d - c_i$ in such a way that on each trip there is an exact match of capacity and demand. Each of these trips has a first-class demand of 1. Therefore, it is covered by *at least* one train unit with first-class demand 1 (and second-class capacity in $\{a_i \mid i = 1, \ldots, n\}$), and, since there are only n of these train units, it is covered by *exactly* one of them. It follows that each of these trips is covered by exactly one train unit with second-class capacity in $\{a_i \mid i = 1, \ldots, n\}$, one train unit with second-class capacity in $\{b_i \mid i = 1, \ldots, n\}$, and one train unit with second-class capacity d. Hence, if $\sigma(i)$ and $\tau(i)$ are defined as the train unit types that are assigned to the trip with second-class demand $2d - c_i$, then it follows that $a_{\sigma(i)} + b_{\tau(i)} + d = 2d - c_i$ for $i = 1, \ldots, n$. Thus, σ and τ are the requested permutations, and it follows that I has a feasible solution.

Conversely, if I has a feasible solution, then the construction can be reversed to find a feasible solution for I' with total fixed costs $4nd - \sum_i c_i$. Since N3DM is NP-complete (Garey and Johnson 1979), and the reduction is polynomial, it follows that the rolling stock circulation problem is NP-hard. □

The proof of Theorem 1 strongly uses the fact that the number of trains and the number of train unit types are not fixed a priori. However, if these numbers are fixed a priori, then the problem can be solved in polynomial time by a dynamic programming approach, as is shown in Theorem 2. Note that this result is hardly relevant from a computational point of view, due to the huge size of the involved dynamic programming network.

THEOREM 2. *If the number of trains, number of stations, number of train unit types, and maximum train length are fixed, then a feasible rolling stock circulation with minimum total fixed costs can be found in an amount of time that is polynomial in the number of trips.*

PROOF. This theorem is proved by solving the problem as a shortest path problem in a network with numbers of nodes and arcs polynomial in the number of trips. Here, we only give a rough sketch of the proof. For similar proofs, see Arkin and Silverberg (1987) or Kroon, Romeijn, and Zwaneveld et al. (1999).

Without loss of generality, all arrivals and departures occur at different time instants. An event is defined as the arrival or departure of a train at a station, and each node in the network represents a feasible state of the system between two consecutive events. These nodes represent both the compositions of *all* trains running in the corresponding time interval and the inventories of the train units at *all* stations. Each arc in the network represents a feasible transition between two successive states of the system. If the numbers of trains, stations, train unit types, and the maximum train length are fixed, then it is not difficult to see that the numbers of nodes and arcs in the network are polynomial in the number of trips. The costs of the arcs can be defined in such a way that a shortest path in the network corresponds with a rolling stock circulation with minimum total fixed costs. □

5. Solution Approach

We first tried to solve Models 1 and 2 with the commercially available integer programming solver CPLEX 8.0. For instances of the line 3000 (described in §2.3), Model 1 could be solved in acceptable running times. However, for Model 2, this approach did not lead to acceptable running times. Therefore, more dedicated methods were needed for solving Model 2.

For single commodity flow problems, many algorithms are available (Ahuja, Magnanti, and Orlin 1993), but for multicommodity flow problems, this is not the case. Furthermore, the available algorithms usually assume that variables may have fractional values (Ahuja, Magnanti, and Orlin 1993; McBride 1998). Solution approaches for integer multicommodity flow problems were studied by Barnhart, Hane, and Vance (1998) based on branch-and-price techniques. However, this paper does not deal with the *ordered* integer multicommodity flow problem. The solution approach that we propose in the current paper for solving the latter problem in the context of our rolling stock circulation problem can be summarized as follows.

1. Solve Model 1. This provides a usually strong lower-bound L for the solution of Model 2.
2. Reduce the sizes of the transition graphs by successively applying:
   —Node elimination, and
   —Disconnection elimination.
3. If there exists at least one trip whose nodes have all been eliminated, then go back to Step 2 with relaxed elimination conditions.
4. If each trip still has at least one node, then solve Model 2 with the additional constraint $F(x, y) \leq L$ on the reduced transition graphs. If this model has a feasible solution, then STOP. Otherwise go back to Step 2 with relaxed elimination conditions.

The lower-bound L obtained in Step 1 by solving Model 1 turns out to be quite strong usually. Indeed, for many instances of the rolling stock circulation problem, we observed that the optimal objective function values of Models 1 and 2 were the same, although the solutions themselves were not.

Therefore, after Model 1 has been solved in Step 1 with optimal objective function value L, we check Steps 2 to 4 if there exists a solution for Model 2 with the *same* optimal objective function value L. The latter is done by first reducing the numbers of nodes and arcs of the transition graphs by the application of *node elimination* and *disconnection elimination*. If an optimal solution for Model 2 with objective function value L does not exist, then Steps 2 to 4 are repeated with a relaxed objective function constraint.

5.1. Node Elimination in Step 2

The node elimination process consists of solving a series of subproblems. Each subproblem corresponds to solving the linear programming relaxation of Model 1, where (i) the constraint $F(x, y) \leq L$ has been added, and (ii) a certain variable $x_{t,j}$ has been fixed to one of its a priori feasible values V. These a priori feasible values for a variable $x_{t,j}$ are determined based on the passenger demand for trip t and on the maximum train length for trip t.

If a linear programming instance in which a variable $x_{t,j}$ has been fixed to the value V does *not* have a feasible solution, then obviously the corresponding integer programming instance is infeasible as well.

Therefore, all nodes and arcs in the transition graph corresponding to trip t and each composition $k \in K_t$ with $g_{k,j} = V$ can be eliminated from this transition graph. This node elimination process iterates over all variables $x_{t,j}$ and over all a priori feasible values V.

For example, if a linear programming instance in which a certain variable $x_{t,4}$ has been set to the value $V = 2$ turns out to be infeasible, then all nodes and arcs corresponding to this trip and a composition with two train units of type "4" (that is, "44," "344," "434," and "443") are eliminated from the involved transition graph.

5.2. Disconnection Elimination in Step 2

The node elimination process may have the effect that some nodes in a transition graph become disconnected, because all their neighbors have been removed. Then, obviously, no path in this transition graph from the source node to the sink node may pass through these nodes. We then perform *disconnection elimination* by eliminating all nodes that have become disconnected in this way. During the disconnection elimination, other nodes may become disconnected. Thus, this process may be carried out until no more disconnected nodes are eliminated.

Once the disconnection elimination process terminates, another round of node elimination could be started. This process could be carried out until no more nodes are eliminated at all. However, each iteration has a certain cost in terms of the required CPU time. Because this cost turns out to be high, in particular in comparison with the number of additionally eliminated nodes, Step 2 is carried out only once per major iteration of the algorithm.

5.3. Iteration in Step 3

If, at the end of Step 2, there exists at least one trip whose nodes have all been eliminated, then obviously a feasible solution for Model 2 on the reduced transition graphs with objective function value L does not exist: Too many nodes have been eliminated to allow for a feasible solution. In this case, we go back to Step 2, where the elimination process is relaxed. That is, in Step 2, we set $L := L + \varepsilon$, where the actual value of ε depends on the objective function $F(x, y)$. By relaxing the elimination conditions, the number of eliminated nodes and arcs decreases, which increases the probability of obtaining a feasible solution for Model 2. Note that, when going back from Step 3 to Step 2, the search process of looking for a feasible solution for Model 2 is *completely* restarted. The same holds for going back from Step 4 to Step 2.

5.4. Iteration and Termination in Step 4

If, at the end of Step 2, there still exists at least one node for each trip, then each reduced transition graph still contains a path from source to sink. In that case, the remaining transition graphs are used as input for Model 2 with the additional restriction $F(x, y) \le L$.

If this model has an optimal solution, then this solution is an optimal solution for Model 2. Otherwise, if a feasible solution for this model does not exist, then we also go back to Step 2, where the relaxed elimination process is carried out after setting $L := L + \varepsilon$.

Note that, if we would not use the additional constraint $F(x, y) \le L$ in this step, then we may find a feasible solution that may not be an optimal solution for Model 2. Indeed, many nodes of the transition graphs have been eliminated based on the assumption that $F(x, y) \le L$.

6. Computational Results

6.1. Experiments

This section presents the computational results obtained by applying the algorithm described in §5 to the case of the intercity line 3000 of NS Reizigers that is described in §2.3. The data that were used for our experiments correspond to a single generic workday. For the solution of the integer programming models and their linear programming relaxations, we used CPLEX (version 8.0) on a RISC 6000 workstation.

In our experiments, we consider single-deck train units for which two types are available: Train units with three carriages and train units with four carriages (indicated as type "3" and type "4," respectively). On all trips, the maximum train length is 12 carriages. To evaluate the influence of the valid inequalities of the capacity constraints instead of the capacity constraints themselves, we experimented with both model formulations.

We solved the instances for a complete day, but also for just the trips that start before 10:30 AM (called the morning peak hours, or MPH). In fact, the total required capacity of the train units on a certain line is mainly determined by the passengers' seat demand during the morning peak hours. Indeed, this period is usually the busiest period of the day, because the peak in the morning is higher than the peak in the afternoon, which lasts longer.

6.2. Results

In Table 1, the dimensions of the case study are reported in terms of the numbers of columns, rows and nonzeros in the constraint matrix. The labels "Compl." and "MPH" indicate the instances with all trips in the timetable and the instances reduced to trips leaving during the morning peak hours.

Obviously, in Model 1, the number of decision variables is less than the number of constraints, while in Model 2, the situation is the opposite. The latter is due to the large number of potential compositions per trip.

Table 1 Dimensions of the Instances of the Line 3000

	Compl.	MPH
Model 1		
Columns	1,544	494
Rows	1,755	544
Nonzeros	4,132	1,290
Model 2		
Columns before elimination	10,148	2,992
Columns after elimination	9,062	2,331
Rows	6,380	1,894
Nonzeros	66,017	16,182

For Model 2, the numbers of decision variables and constraints representing the compositions depend on the number of nodes eliminated from the transition graphs and, thus, also on the objective function used. The values in Table 1 are average values observed in our computational experiments.

Tables 2 to 4 show the results of our computational experiments. The labels "No VI" and "VI" in the columns refer to the instances where the valid inequalities described in §4.2 have not been included and have been included, respectively.

The rows "Tot. nodes" and "Elim. nodes" contain the total number of nodes in the transition graphs and the number of nodes eliminated in the elimination phase. The difference between these two numbers gives the total number of nodes in the reduced transition graphs. The row "Iterations" indicates the number of major iterations required in the solution process.

The row "CPU Total" indicates the total CPU time in seconds required for the complete solution process. This total CPU time is split up in the next rows into the CPU time required for solving Model 1 (row "CPU Model 1"), the time required for the elimination phase (row "CPU Elim."), and the time required for solving Model 2 (row "CPU Model 2").

The row "Obj. Model 1" and "Obj. Model 2" denote the obtained objective function values for Models 1 and 2. Note that the obtained objective function values for Models 1 and 2 are very close indeed. The obtained numbers of train units of type "3" and "4" in the solution of Model 2 are represented in the rows "y_3^{tot}" and "y_4^{tot}," respectively.

Table 2 Results for the Objective Function PB1

	Compl, No VI	Compl, VI	MPH, No VI	MPH, VI
Tot. nodes	8,604	8,604	2,498	2,498
Elim. nodes	515	1,950	298	1,597
Iterations	4	4	3	3
CPU Total	2,829.2	1,771.2	119.3	62.9
CPU Model 1	338.3	128.1	13.2	4.7
CPU Elim.	2,193.8	1,472.7	88.5	53.1
CPU Model 2	397.1	170.4	17.6	5.1
Obj. Model 1	109	109	110	110
Obj. Model 2	112	112	112	112
y_3^{tot}	20	20	20	20
y_4^{tot}	13	13	13	13

Table 3 Results for the Objective Function PB2

	Compl, No VI	Compl, VI	MPH, No VI	MPH, VI
Tot. nodes	8,604	8,604	2,498	2,498
Elim. nodes	416	416	144	144
Iterations	1	1	1	1
CPU Total	5,202.0	308.9	41.1	19.2
CPU Model 1	1,524.1	88.7	10.5	3.7
CPU Elim.	2,077.5	124.9	19.1	11.6
CPU Model 2	1,600.4	95.3	11.5	3.9
Obj. Model 1	1,735	1,735	617	617
Obj. Model 2	1,735	1,735	617	617
y_3^{tot}	23	23	23	23
y_4^{tot}	14	14	11	11

6.3. Comments

Tables 2 to 4 show that, after the elimination phase, solving Model 2 to optimality takes roughly the same amount of computation time as solving Model 1. The latter is mainly due to the excellent lower bound provided by the solution of Model 1 and to the results of the elimination phase. Indeed, we also experimented with the solution process without this lower bound or the results of the elimination phase, but these experiments resulted in strongly increased running times.

Due to the excellent lower bound provided by the solution of Model 1, often only one major iteration of the solution process is required. The latter implies that there exists an optimal solution for Model 2 with the same objective function value as the initial optimal solution for Model 1. Note that this does *not* imply that this initial optimal solution for Model 1 is already feasible for Model 2. Because we are also interested in the details of the solution (the rolling stock circulations themselves), this justifies the extension of Model 1 to Model 2.

Table 4 Results for the Objective Function PB3

	Compl, No VI	Compl, VI	MPH, No VI	MPH, VI
Tot. nodes	8,604	8,604	2,498	2,498
Elim. nodes	449	2,766	298	1,597
Iterations	1	1	1	1
CPU Total	5,819.8	326.4	42.8	20.8
CPU Model 1	1,611.5	91.3	11.1	4.3
CPU Elim.	2,200.2	132.6	20.3	12.3
CPU Model 2	2,008.1	102.4	11.4	4.2
Obj. Model 1	1,769	1,769	621	621
Obj. Model 2	1,769	1,769	621	621
y_3^{tot}	20	20	24	24
y_4^{tot}	13	13	10	10

Furthermore, Tables 2 to 4 show that for all objective functions, a large part of the total CPU time is spent on the elimination phase: during the node elimination phase, the solution of each linear programming subproblem takes little time, but the large number of such subproblems to be solved here makes the node elimination process time consuming. However, as was noted above, skipping the elimination phase resulted in increased computation times.

Obviously, the additional valid inequalities (11) and (12) speed up the solution process of Model 1 significantly, due to the improvement of the linear programming lower bound. Note that the valid inequalities also reduce the time required for the elimination phase significantly. The latter is because the valid inequalities are also applied when solving the linear programming subproblems in the elimination phase, which results in stronger lower bounds.

In all cases, the results obtained by solving the MPH instances are rather similar to the results obtained by solving the complete instances in terms of the total required capacity of the train units, as was expected. Anyway, the fixed costs of a certain MPH instance provide a strong lower bound for the fixed costs of the corresponding instance for a complete day.

7. Conclusions and Further Research

In this paper, we studied the problem of determining optimal numbers of train units together with their efficient circulation on a single line, thereby considering that trains can be composed of train units of different types. The latter implies that not only the *number* of train units of the different types in the trains, but also their *order* in the trains is to be modeled.

We proved that the rolling stock circulation problem is NP-hard in its most general form. However, if the numbers of trains, stations, train unit types, and the maximum train length are fixed, then an optimal rolling stock circulation can be found in an amount of time polynomial in the number of trips.

We described a model and an algorithm for solving the rolling stock circulation problem. The model uses the concept of *transition graphs*. The algorithm starts with reducing the sizes of the transition graphs as much as possible by applying node elimination and disconnection elimination. Then the remaining problem is solved by CPLEX.

We applied the algorithm to the Intercity line 3000 of NS Reizigers. Based on our results, it can be concluded that the proposed solution approach is a sufficiently powerful scheme, which succeeds to find optimal solutions within an acceptable amount of time.

In our further research, we will study cases that have a more complex structure than the one presented in this paper. Such cases may involve several train lines at the same time, more than one family of train units (for example, both single-deck and double-deck train units), more than two types of train units, or trains splitting and combining underway. First results on this subject can be found in Fioole et al. (2005).

In addition to the *tactical* problem of determining appropriate numbers of train units to be operated on a certain set of lines, we will also focus on the *operational* problem of optimally circulating a *given* number of train units along these lines. In that case, the problem is to find a balance between the conflicting objectives of minimizing (i) the shortages of seats (service), (ii) the number of train unit or carriage kilometers (efficiency), and (iii) the number of shunting movements (robustness). The latter is relevant because shunting movements are potential sources of disruption of the railway process. Therefore, avoiding these shunting movements may be beneficial for the punctuality.

In our further research, we will also focus on alternative algorithmic approaches for solving the rolling stock circulation problem. In particular, we will experiment with a solution approach based on column generation. Here the column generation mechanism generates appropriate paths through the transition graphs based on shortest path algorithms. This column-generation mechanism takes into account *dual cost* information obtained from the master problem. The latter handles the coordination between the paths in the different transition graphs, mainly by taking into account the inventories of the train units in the different stations. First results on this subject can be found in Peeters and Kroon (2006).

References

Abbink, E. J. W., B. W. V. van den Berg, L. G. Kroon, M. Salomon. 2004. Allocation of train units to passenger trains. *Transportation Sci.* **38**(1) 33–41.

Ahuja, R. K., T. L. Magnanti, J. B. Orlin. 1993. *Network Flow—Theory, Algorithms, and Applications*. Prentice Hall, Englewood Cliffs, NJ.

Arkin, E. M., E. L. Silverberg. 1987. Scheduling jobs with fixed start and end times. *Discrete Appl. Math.* **18** 1–8.

Barnhart, C., A. Hane, P. H. Vance. 1998. Using branch-and-price to solve origin-destination integer multi-commodity flow problems. *Oper. Res.* **32** 208–220.

Ben-Khedher, N., J. Kintanar, C. Queille, W. Stripling. 1998. Schedule optimization at SNCF: From conception to day of departure. *Interfaces* **28** 6–23.

Brucker, P., J. Hurink, T. Rolfes. 2003. Routing of railway carriages: A case study. *J. Global Optimiz.* **27** 313–332.

Cordeau, J. F., F. Soumis, J. Desrosiers. 2000. A Benders decomposition approach for the locomotive and car assignment problem. *Transportation Sci.* **34** 133–149.

Cordeau, J. F., F. Soumis, J. Desrosiers. 2001. Simultaneous assignment of locomotives and cars to passenger trains. *Oper. Res.* **49** 531–548.

Fioole, P.-J., L. G. Kroon, G. Maróti, A. Schrijver. 2005. Rolling stock circulation of passenger train units: Splitting and combining of passenger trains. CWI Report PNA-E0420, ISSN 1386-3711. CWI, Amsterdam, The Netherlands. *Eur. J. Oper. Res.* Forthcoming.

Garey, M. R., D. S. Johnson. 1979. *Computers and Intractability: A Guide into the Theory of NP-Completeness*. Freeman, San Francisco, CA.

Groot, R. 1996. Minimum circulation of railway stock: An integer programming algorithm. Master's thesis, University of Amsterdam, The Netherlands.

Kroon, L. G., H. E. Romeijn, P. J. Zwaneveld. 1999. Routing trains through railway stations: Complexity issues. *Eur. J. Oper. Res.* **98** 485–498.

Lingaya, N., J. F. Cordeau, G. Desaulniers, J. Desrosiers, F. Soumis. 2002. Operational car assignment at VIA Rail Canada. *Transportation Res. Part B* **36** 755–778.

Maróti, G., L. G. Kroon. 2005. Maintenance routing of passenger train units: The transition model. *Transportation Sci.* **39**(4) 518–525.

McBride, R. D. 1998. Advances in solving the multi-commodity-flow problem. *Interfaces* **28** 32–41.

NS homepage: www.ns.nl.

Peeters, M., L. G. Kroon. 2006. Circulation of railway rolling stock: A branch-and-price approach. *Comput. Oper. Res.* Forthcoming.

Schrijver, A. 1993. Minimum circulation of railway stock. *CWI Quart.* **6** 205–217.

van Monfort, J. 1997. Optimizing Railway Carriage Circulation with Integer Linear Programming. Master's thesis, University of Amsterdam, The Netherlands.