Rotation Planning of Locomotive and Carriage Groups with Shared Capacities

Taïeb Mellouli[1] and Leena Suhl[2]

[1] Department of Management Information Systems and Operations Research, Martin-Luther-University Halle-Wittenberg, Universitaetsring 3, 06108 Halle (Saale)
mellouli@wiwi.uni-halle.de

[2] Decision Support & OR Laboratory, University of Paderborn, Warburger Str. 100, 33098 Paderborn, Germany
suhl@uni-paderborn.de

Abstract. In a large railway passenger traffic network, a given set of trips or service blocks are to be serviced by equipment consisting of several groups of locomotives/carriages. The allowed groups per service block are predefined as patterns or multisets of locomotives and carriages. A given type of locomotive/carriage may occur with varying numbers in several groups. We search for a cost-minimal assignment of locomotive/carriage groups to rotations taking special restrictions into account, especially, we shall find the optimal mix of groups obeying given capacities on the level of locomotive and carriage units for each type.

Our solution approach is based on a multi-layer (multi-commodity) network flow model where each layer represents a locomotive/carriage group, and the requirement of servicing each trip exactly once is modeled by cover/partitioning constraints. In this paper, we concentrate on railway specific requirements and present special techniques to model and optimize locomotive and carriage groups with shared capacities. These techniques erable us to solve large-scale practical problem instances of German Railways into optimality.

1 Introduction

In railway passenger traffic, carriages and locomotives have to be assigned to trips in order to carry out a given schedule which has been published for passengers. In a large network this scheduling and routing task may be very complex, and until recently it was not generally possible to compute cost-minimal rotations for a given timetable with hundreds or thousands of trips when considering practical requirements such as maintenance rules or multiple types of carriages and locomotives.

We consider a railway network for passenger traffic, consisting of scheduled trips (service trips), each from a given departure station to a given end station. A trip may be divided into legs, also called *service blocks* during which coupling and uncoupling operations of train equipment are not allowed. Each service block has to be serviced with adequate equipment according to requirements considering

F. Geraets et al. (Eds.): Railway Optimization 2004, LNCS 4359, pp. 276–294, 2007.

technology, number of seats, comfort degree, and so on. The equipment consists of one or several groups of locomotives and/or carriages, which contain a given number of locomotives and/or carriages according to given group types. There may be several alternative groups to be used in putting together the equipment for a service block.

For example, the vehicle group types VG11, VG14 and WG15 may consist of carriage types ABn, ABnrz, BDnf, Bn, Bndf, Bnbdz, and Bnrz in the following way:

VG11: 1*ABnrz + 1*Bnbdz + 2*Bnrz VG14: 2*ABnrz + 2*Bn + 1*Bndf + 1*Bnrz VG15: 2*ABn + 1*BDnf + 3*Bn.

As an example, carriages of type ABnrz are needed for vehicle group types VG11 and VG14, carriages of type Bn for VG14 and VG15, and carriages of type BDnf only for VG15.

Thus, each locomotive and carriage belongs to a given equipment type, and the number of vehicles (locomotives/carriages) of each equipment type is limited.

Individual trains, however, consist of *groups* of locomotives and/or carriages, selected out of a given set of vehicle group types, each given group type being set together as a (multi)set of vehicles with a fixed number of members out of each vehicle type. Thus, we aim to develop an optimization model which minimizes the cost of equipment simultaneously satisfying the requirements on the vehicle group level and the given capacities on the locomotive and/or carriage vehicle type.

The total operational cost is to be minimized so that all requirements considering types of locomotives and carriages and the way they are assembled into train units are fulfilled.

Generally speaking, we mean by *train unit*, *train assembly* or *train consist* a group of compatible units of equipment that travel along some part of the physical rail network. A train assembly may include a given number of first class and second class carriages together with one or two locomotives. In most cases of railway applications, multiple types of locomotives and carriages are in use, and for each service unit a set of compatible types is given. In the following, we use the term *loco/car* or *vehicle* as an abbreviation and abstraction of a unit of locomotive, steering-wheel waggon, or rail carriage/car/waggon. A *vehicle type* or a *loco/car type* is the type of a locomotive or carriage unit.

In this paper, we address the rotation planning problem of the railway application area under these requirements. The task is to generate rotations for locomotives and carriages being of a given equipment type and simultaneously being part of one or more loco/car groups. A loco/car type may be involved in several groups and capacities of equipment types have to be taken into account. Thus, on one hand, we have to consider individual loco/cars in order to meet the capacity requirements, and on the other hand, (types of) loco/car groups, in order to take the type requirements into account. Capacities are shared in the sense that different groups share the same loco/car types. This approach is currently being used at German Railways (Deutsche Bahn), and we will present algorithms tested with their data.

The paper is organized as follows. In section 2, a literature review is provided together with details on our previous research work concerning both the railway application area and solution approaches of rotation planning problems. In section 3, the problem of rotation building for loco/car groups with shared loco/car capacities is formalized. In section 4, a mathematical model based on a multi-layer (multi-commodity) flow network is presented where each network layer represents a loco/car group, and the requirement of servicing each trip exactly once is modeled by cover/partitioning constraints. Especially, a special aggregation scheme of "equivalent" loco/car groups is applied in order to solve large-scale problems of German Railways directly by a standard mathematical optimizer. Finally, we present computational results in section 5 and discuss problems of practical relevance solved by exact optimization together with suitable decision support tools.

2 Previous Work and Solution Approach

2.1 Literature Review

Although the problem of simultaneously assigning locomotives and carriages to trips and building rotations is very important to railways and has to be solved on a regular basis in practice, there are relatively few contributions to it in the scientific literature. One of the first papers was Ramani/Mandal (1992) dealing separately with the assignment of locomotives and carriages, and using a local improvement procedure to improve the overall solution. Ben-Kheder et al. (1997) described a system for the simultaneous assignment of locomotives and carriages for passenger trains at SNCF. The system treats both types simultaneously but uses aggregated modules that are then assigned as a whole, thus not dealing explicitly with compatibility constraints. Zirati et al. (1997) consider the problem of assigning locomotives requiring inspection within a time limit of the considered one-week planning horizon.

Cordeau et al. (1998) give a survey on research until 1998. Since then, a few papers have been published. In Cordeau et al. (2000) an optimization model was developed which assigns both locomotives and carriages simultaneously, solving a tactical periodic problem as an integer programming problem based on a time-space network. The authors propose a multi-commodity network model which they solve using Benders decomposition. A second paper of the same authors extends this model in Cordeau et al. (2001) for the practical case where maintenance and equipment substitution is taken into account.

Brucker et al. (2003) formulate the railway carriage routing problem as an integer multi-commodity network flow problem with nonlinear objective function and present a local search solution approach for it.

A recent publication of Abbink et al. (2004) considers the tactical problem of finding the most efficient schedule of for a set of rolling stock to train series, so that as many people as possible can be transported with a seat, especially when there is little seating capacity available during rush hours.

2.2 Own Research Work and Solution Approach

The Decision Support and Operations Research Laboratory at the University of Paderborn, Germany, has been involved since 1996 in several projects concerning design and development of optimization models and decision support tools in public transport. In the railway domain, we have studied practical tasks within both planning and operations control phases, developed optimization models for maintenance routing problems, and designed dispatcher support tools with embedded simulation capabilities, cf. Suhl and Mellouli (1999), *"computer-aided scheduling of public transport"*, Suhl et al. (2001), and Mellouli (2001) CASPT'2000 Berlin.

Our research work on the development of rotation building models and software date back to the work of the second author (Suhl 1995) where an extension to time windows was developed and applied to airlines.

The first author thoroughly applied and extended time-space networks based on connection-lines to deal with various rotation building problems in public transport. In 1997, he developed a state-expanded time-space flow network to deal with maintenance routing problems for railways and airlines (Mellouli 2001), and in 1999 a new aggregation scheme for potential deadhead trips (empty movements) which is crucial to solve hard practical requirements directly by state-of-the-art optimization software and to derive new complexity results for the rotation building problem (Mellouli 2003). This aggregation scheme for potential deadhead trips is successfully applied in the bus transit domain to solve large-scale multiple-depot, multiple-vehicle-type problems (Kliewer, Mellouli, and Suhl (2002)), as well as in the railway domain.

In 2001, our laboratory developed a prototype for rotation building for German Railways with the best optimization results in a prior study. Based on this, a development project with German Railways was accomplished in 2002. This paper presents parts of research results achieved by our laboratory and tested within this project. We concentrate on railway specific requirements and present special techniques to model and optimize loco/car groups with shared capacities. In order to solve large-scale practical problem instances of German Railways into optimality, the aggregation scheme for arcs representing all possible empty train movements is also used to which we refer to our mentioned works.

3 Problem Formalization and Analysis

In the following, we formalize the problem of rotation building for loco/car groups with shared capacities introduced in section 1. For this problem we are given:

- A set of *service blocks* SB: Each service block is a trip or a maximal trip part in which coupling and uncoupling operations are not performed. Thus, a service trip may consist of one block or of a sequence of blocks with different requirements on used train parts.

- A set VG of (types of) *vehicle groups* (or *loco/car groups*): Each $vg \in VG$ defines a group of locomotive and carriages which can be used as *train part* for some service blocks.
- A set of vehicle types VT (or loco/car types): Thus, VT consists of the different types of locomotives and carriages available.
- A set of home bases HB: Home bases are stations where vehicles may be stationed. For each $vt \in VT$ and $hb \in HB$, let $capacity(vt, hb)$ be the number of vehicles of type vt stationed at homebase hb.

For each $vg \in VG$ and each $vt \in VT$, let $number(vg, vt)$ be the number of vehicles of type vt occurring in the vehicle group vg. For instance, if vehicle group vg_1 consists of 2 vehicles of type vt_6 and one vehicle from the types vt_{12}, vt_{13}, and vt_{14}, respectively, so we have $number(vg_1, vt_6) = 2$, $number(vg_1, vt_{12}) = 1$, $number(vg_1, vt_{13}) = 1$, $number(vg_1, vt_{14}) = 1$, and $number(vg_1, vt) = 0$ for all other vehicle types vt.

Restrictions on assignments of service blocks to vehicle groups are regulated as follows:

- *Assignments of service blocks to vehicle groups:* For each service block $sb \in SB$, there may be several assignments of vehicle groups (not necessarily of different types) for different positions in a train unit. These vehicle groups define the *train assembly* that serves the service block sb. Alternative types of vehicle groups for the same train position are given by means of *global* or *local replacements* of (types of) vehicle groups.
- *Global and local replacements of vehicle groups:* A *global replacement* of the form $vg_i \leftarrow vg_j$ is declared independently of service block assignments. For each service block and train position, if the (type of) vehicle group vg_i can be assigned, then the (type of) vehicle group vg_j can be assigned alternatively.

  A *local replacement* is defined for each specific service block (and train position) by listing the possible (types of) vehicle groups that are allowed for serving this specific service block.

The test data of German Railways that is related to this specific problem comprises 31 different types of vehicle groups, two home bases, and 7,500 assignments of service blocks to vehicle groups. Most of the vehicle groups consist of 6, 5, or 4 vehicles (only one vehicle group contains a single locomotive and two others contain only one carriage vehicle as reinforcement).

The difficulty of the problem is directly related to the possibility of global and local replacement of vehicle groups. In the following, we consider two variants, a simple problem version without, and a complex one, with such replacements:

The simple problem version: Having no replacements of vehicle groups, the vehicle group used for each service block and train position is unique. So the problem can be decomposed according to different vehicle groups, by considering subsets of service blocks uniquely assigned to different vehicle groups, respectively. Minimizing the number of vehicle groups used for each sub-problem is then equivalent to minimizing the number of vehicles (locomotives and carriages) used. For the sub-problems, a polynomial-time procedure for basic rotation building (or vehicle scheduling) problem can be applied.

Let $MFSZ_{vg}$ be the number of vehicle groups used (minimum fleet size) for the subset of service blocks SB_{vg} assigned to vg. Then for each vehicle type vt the total number of used vehicles from type vt is equal to:

$$\sum_{vg \in VG} (MFSZ_{vg} * number(vt, vg))$$

The complex problem version: For each service block and train position, there may be several types of vehicle groups that can be assigned. This results from the local and global replacements of vehicle groups given for this problem setting. Global and local replacements define sets of alternative vehicle groups corresponding in some sense to "groups" of bus types/depots in the multiple-vehicle-type, multiple-depot vehicle scheduling problem (MDVSP, cf. Löbel (1998) and Kliewer, Mellouli, and Suhl (2002)).

Note that the use of the term "group" is different: For the multiple vehicle type problem, a vehicle type group is a set of alternative types of vehicles that can be used to serve a specific trip. For the considered railway application, a vehicle group defines an assembled pattern of vehicles of predefined types and numbers which are required for a certain train position. Because of this, we can say that, for each service block and train position in our problem setting, *a type group of vehicle groups* is given, i.e., a (type of) vehicle group is to be selected out of several feasible alternatives for each service block.

The extra difficulty of the problem is that minimizing vehicle groups of different types does not necessarily use a convenient constellation of locomotive and carriage types according to their availability. In the next sections, we review in short the multi-layer (multi-commodity) flow network for multiple vehicle types problems and apply it to types of vehicle groups for our case study. Then, we extend this model in order to create a "link" between vehicle or loco/car capacities and number of used vehicle groups of different types. Furthermore, we present an aggregation of "equivalent" vehicle groups decreasing the complexity and solution times for large-scale problems.

4 Multi-layer Flow Model and Shared Vehicle Capacity

There are basically two types of networks for basic rotation building and vehicle scheduling problems with one vehicle type: a *trip-as-node network* and a *trip-as-arc network*, where the latter is used in this paper. In both types of network for basic rotation building, vehicles of homogeneous type are modeled by flow units that originate from a depot node and terminate there for bus transit. For railways, there is generally no need of fixed depots and the "vehicle flow" *originates* from a virtual *source node* indicating period start and terminate at a virtual *target node* indicating period end. For the case of building genuine rotations for *periodic* (daily or weekly) timetables, frequently used in railways, this "vehicle flow" *circulates* within the network over *wrap-around period change arcs*. The sum of flow values on these arcs modeling *periodicity* corresponds to the number of used vehicles (of the considered type), since each vehicle used at the end of the period must "flow" back using one of these arcs in order to be used at the beginning of the next period at its space and time of availability (prescribed by its latest served activity, being a timetable trip or a deadhead trip).

In 4.1, we recall basic properties of multi-layer (multi-commodity) flow networks for rotation building with several vehicle types and apply the idea to vehicle group(type)s. Details on the design of this flow network for rotation building with our optimality-preserving deadhead trip aggregation are provided in 4.2 together with the objective function and constraints of the resulting mathematical model in 4.3. This model is extended in 4.3 and 4.4 for some specifics of the railway application studied in this paper.

4.1 The Basic Multi-layer Flow Model for loco/car Groups

To model rotation building problems with several vehicle types, we have to ensure that *vehicles of different types are not merged within the model network*. This can be achieved by constructing a multi-layer network (cf. Figure 1), where different *flow commodities* circulate on different layers, and thus cannot be merged. A network layer or a commodity corresponds to a vehicle type. For our case study, we adapt this multi-layer network where a vehicle type is replaced by a loco/car group (an not by a loco/car type).

The multi-layer network flow model results in a mixed-integer (linear) program (MIP) which is computationally much more difficult to solve than pure minimum cost flow problems. Besides the flow conservation constraints (that are to be formulated separately for each network layer, see 4.2), there are *cover/partitioning constraints* of non-flow type that ensure that each trip is included in the solution flow of only one network layer. These cover/partitioning constraints involve flow variables of value 0 or 1 and make the resulting optimization model of mixed-integer type.

The computational burden for solving these multi-layer network flow problems is due to the fact that the number of variables and constraints are multiplied by the number of network layers. Using the classical trip-as-node flow network model, whose number of variables for possible connection arcs already grows quadratically for basic problems with a single commodity, the resulting mixed-integer models cannot be solved directly by standard optimizers for timetables with thousands of trips. Special solution techniques, such as column generation or branch&price with Lagrangean relaxation, have been applied in order to solve problems of practical size (cf. Löbel (1998)), though in some cases only fleet minimal.

Using our model based on trip-as-arc network, the resulting models are much smaller owing to our powerful aggregation of possible deadhead trips. For large-scale instances of the multiple-depot problem of bus transit, computational results based on direct use of mathematical optimization software are presented in Kliewer, Mellouli, and Suhl (2002).

For each service block $sb \in SB$, let VG_{sb} be the list of possible vehicle groups that can be used to serve this specific service block. This list VG_{sb} is built

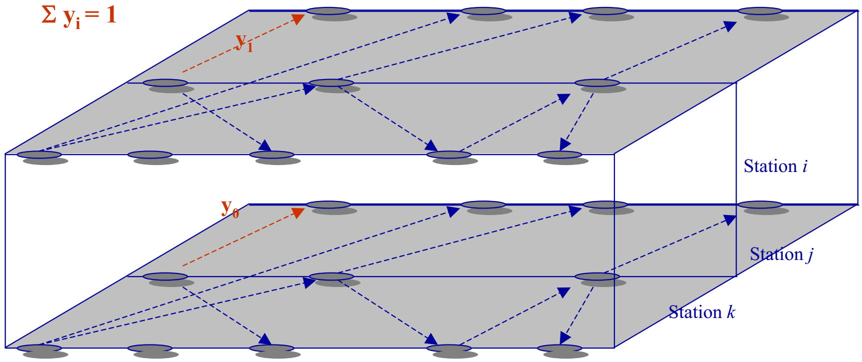

Fig. 1. Multi-layer flow network

by considering all vehicle groups included in local replacements for this specific service block sb and then adding each vg_j into VG_{sb} for each existing global replacement rule $vg_i \leftarrow vg_j$ where $vg_i \in VG_{sb}$ (transitive hull).

Let the 0/1-variable $Y_{sb,vg}$ denote the flow value on the arc modeling service block sb in the vg-layer of the network. The cover/partitioning constraints can be formulated as follows:

$$\sum_{vg \in VG_{sb}} Y_{sb,vg} = 1$$

for each service block $sb \in SB$.

As can observed in Figure 1, several Y variables exist for the same trip in different network layers. For basic problems with one commodity, all Y variables for trip arcs are set to 1, as only one network layer is involved. For the multi-layer network, the sum of flow variables $Y_{sb,vg}$ for a fixed service block sb over all network layers is equal to one (in order to guarantee that block sb is carried out by exactly one loco/car group). The optimizer will decide which of the 0/1-variables $Y_{sb,vg}$ (for a fixed service block sb) will be equal to 1. For a computed optimal solution, a value of $Y_{sb,vg} = 1$ means that service block sb is served by a loco/car group (with type) vg.

The overall model is a minimum cost multi-layer flow problem with side cover/partitioning constraints. To further reduce the size of network layer, for each commodity or loco/car group $vg \in VG$, we consider only the subset SB_{vg} of service blocks from SB that can be served by vehicle group vg. Thus, $SB_{vg} = \{sb \in SB \mid vg \in VG_{sb}\}$.

4.2 Trip-as-Arc Flow Network with Deadhead Trip Aggregation

Besides the partitioning/cover constraints, the mathematical model includes usual balance constraints on nodes for incoming and outgoing flow, separately for each vg network layer. Considering figure 1, the nodes in each network layer represent time-space points. These nodes are organized into connection lines

$CL(s, vg)$ to each station s, separately for each network layer vg. Each connection line $CL(s, vg)$ includes a line of nodes $N_i^{s,vg}$ modeling certain points in time at that station s for $i = 0, 1, 2, ..., n_{s,vg}$ (= number of nodes in $CL(s, vg)$), depending on vg and s. The nodes $N_i^{s,vg}$ of a connection $CL(s, vg)$ are connected by *waiting arcs*, where $X_i^{s,vg}$ denotes the flow of vehicle groups of type vg waiting from $N_{i-1}^{s,vg}$ to $N_i^{s,vg}$. Whereas the variables $X_i^{s,vg}$ connect nodes of the same connection line, the variables $Y_{sb,vg}$ for service blocks connect nodes of different connection lines – from a certain node of $CL(\textit{start-station}(sb), vg)$ to a node of $CL(\textit{end-station}(sb), vg)$ depending on start time and end time of service block sb, respectively.

Compatible service blocks sb_1 and sb_2 with $s = \textit{end-station}(sb_1) = \textit{start-station}(sb_2)$ and $\textit{end-time}(sb_1) \leq \textit{start-time}\ (sb_2)$ can be linked in the model through one (connection) node in $CL(s, vg)$ or through several consecutive (connection) nodes of this connection line over one or several waiting arcs. Thus connections of trips are not modeled explicitly but aggregated for several connections over the use of connection lines.

This aggregation is extended in Mellouli (2003) to the efficient representation of the quadratic number of deadhead connections needed to potentially connect all service blocks for the case where $s_1 = \textit{end-station}(sb_1) \neq s_2 = \textit{start-station}(sb_2)$. Our idea illustrated in Figure 2 is based on a two-stage aggregation for potential deadhead trips (matches).

The first stage aggregation is based on the matter of fact that each match is implicitly represented by taking the *first match* arc to the destination of the deadhead trip and then eventually going through waiting arcs of that destination station. The second stage aggregation is based on the observation that for a bundle of first match arcs connected to the same target service block *f*, only the *latest* one is needed, because we can go through waiting arcs of start station of deadhead trip until the *latest first match* in order to implicitly represent the connection of the omitted first matches.

Thus using waiting arcs of a trip-as-arc network only *latest first matches* (see Figure 2) are needed within the used connection line based network model in order to implicitly model all potential matches. The number of these latest first matches are considerably smaller than the number of service blocks multiplied by the number of stations. Since the number of stations is practically of factor 100 smaller for large models than the number of trips, a considerable reduction of arcs (and thus of model variables) is achieved. (In comparison, trip-as-node networks needs quadratic number of arcs relative to the number of service blocks). To get a figure on the impact of this aggregation, we could reduce in our railway case study with 7,500 service block arcs of 14 network layers, 134,643 matches to 5,861 first matches, and further to 1,661 latest first matches. This makes hard extensions of this network model solvable by direct use of mathematical optimization software for the large-scale railway application (cp. computational results in the next section). Results of the application of our model for multi-depot bus scheduling are presented in Kliewer, Mellouli, and Suhl (2002).

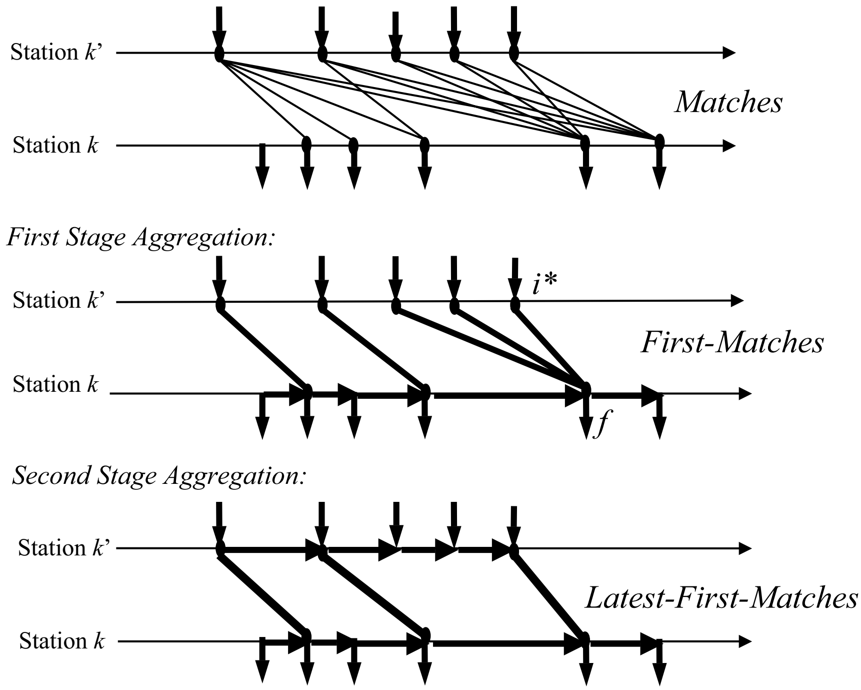

Fig. 2. Two-stage Aggregation for potential deadhead trips

4.3 Mathematical Model

The overall mathematical multi-layer flow model contains the partition/cover constraints given in 4.1 and flow balance constraints for the network described in the previous subsection based on connection lines and deadhead trip aggregation. For each connection node $N_i^{s,vg}$ in each connection line $CL(s, vg)$ (to each vehicle group vg and station s), let the set of:

- service block arcs incoming into this node be denoted by $E_i^{s,vg}$
- service block arcs outgoing from this node be denoted by $S_i^{s,vg}$
- latest first match arcs incoming into this node be denoted by $LE_i^{s,vg}$
- latest first match arcs outgoing from this node be denoted by $LS_i^{s,vg}$

Using these sets the flow balance constraints on connection line nodes $N_i^{s,vg}$ can be formulated as follows:

$\forall vg$, $\forall s$, and $\forall i = 0, 1, 2, ..., n_{s,vg}$ (number of nodes in $CL(s, vg)$):

$$X_i^{s,vg} + \sum_{sb \in E_i^{s,vg}} Y_{sb,vg} + \sum_{lfm \in LE_i^{s,vg}} Z_{lfm,vg} = X_{i+1}^{s,vg} + \sum_{sb \in S_i^{s,vg}} Y_{sb,vg} + \sum_{lfm \in LS_i^{s,vg}} Z_{lfm,vg}$$

Here the Y- and X-variables represent the flow on service block arcs and waiting arcs respectively (as described in 4.1 and 4.2), the Z-variables the flow on latest first match (deadhead) arcs. Whereas the flow value for Y-variables is of 0/1-type, the X- and Z-variables are general integers, since several aggregated matches can use the same waiting or latest first match arc. One can show that the X-variables can be declared continuous, since they must take integer values anyway.

The partitioning/cover constraints for arcs of the same service block sb described in 4.1 are added into the model:

$$\forall sb \in SB \qquad \sum_{vg \in VG_{sb}} Y_{sb,vg} = 1$$

As described at the beginning of this section, periodicity of timetables is modeled by *wrap-around back arcs*. Especially. for the case of waiting arcs, the X-variable with last i-index in each connection line $CL(s, vg)$) is set equal to (or replaced by) that with 0-index of the same connection line.

$$\forall vg \forall s : \quad X^{s,vg}_{n_{s,vg}} = X^{s,vg}_0$$

Let the set of period change arcs for service blocks (e.g., starting on Sunday 23:00 and ending on Monday 2:00 for a weekly timetable) on vg-network layer be denoted by $PC(vg)$ and the set of period change arcs for latest first matches on vg-network layer be denoted by $PCL(vg)$, then the **fleet size constraints** (to each vg network layer) can be formulated as follows:

$$\forall vg : \quad FSZ(vg) = X^{s,vg}_0 + \sum_{sb \in PC(vg)} Y_{sb,vg} + \sum_{lfm \in PCL(vg)} Z_{lfm,vg}$$

Here, $FSZ(vg)$, the fleet size (= number of units) for vehicle group vg is set equal (as remarked at the beginning of this section) to the sum of flow values over wrap-around period change arcs for periodical timetables within the network layer for vg. It is important to see that not only waiting arcs can be period change arcs, but also arcs for service blocks and for latest first matches can be of this type.

The objective function of the overall model minimizes the overall fixed and variable costs of the rotation building problem. Fixed costs F_{vg} are those for used units of vehicle groups of type vg and the variable costs C_{lfm} for empty movements of vehicles incurred when using one of the latest first match arcs lfm of the network layer vg (set of all latest first matches being denoted by $LFM(vg)$. The objective function can now be stated easily:

$$minimize \quad \sum_{vg \in VG} FixCost_{vg} * FSZ(vg) + \sum_{lfm \in LFM(vg)} Cost_{lfm} * Z_{lfm,vg}$$

Note that $Cost_{lfm}$ can be made dependent on vehicle group vg proceeded empty from one station to another by setting $Cost_{lfm,vg}$ and that we can set small costs on waiting X-variables on connection lines of non-maintenance stations to favor standing times of vehicle groups at maintenance stations.

4.4 Modeling Shared loco/car Capacities Within the Flow Network

Recall that the problem is to build rotations for vehicle groups while assigning each service block to a vehicle group (type) and regarding the shared capacities on vehicle level, i.e., on loco/car level. To model the problem, we first take a network layer for each vehicle group and construct the multi-layer aggregated flow network as discussed in the previous subsections. The link between the number of used vehicle groups in the network layers and the available capacities of individual vehicles or loco/cars is reached by the following additional constraints (using the terminology in section 3):

Vehicle capacity constraint

$$\forall\, vt \in VT : \sum_{vg \in VG} FSZ(vg) * number(vg, vt) \leq capacity(vt)$$

where $capacity(vt)$ is the available number of vehicles or loco/cars of type vt over all homebases (sum of $capacity(vt, hb)$ over all $hb \in HB$) and $FSZ(vg)$ denotes the fleet size on the network layer for vehicle group vg.

Here, we have a direct relation to the **fleet size constraints** (to each vg network layer) formulated in the previous subsection.

If it is desired to consider different homebases for vehicles separately, we can write the vehicle capacity constraint as follows:

$$\forall\, vt \in VT\ \ \forall\, hb \in HB : \sum_{vg \in VG} FSZ(vg, hb) * number(vg, vt) \leq capacity(vt, hb)$$

As we have introduced a network layer to each (type of) vehicle group, the resulting model for the data of German Railways (see Section 3), including 32 network layers and 7,500 service blocks, risks to become computationally difficult and perhaps not directly solvable by optimization software. Therefore, we developed a technique to reduce the number of network layers in order to considerably reduce the model sizes. This refinement is discussed in the following subsection.

4.5 Model Refinement by Aggregating Vehicle Groups

As a motivation for this model refinement, we consider the characteristics of the test data of German Railways for this rotation problem for several vehicle group types. We have 32 vehicle groups denoted by $VG0, VG1, VG2, ..., VG31$. Most of the vehicle groups consist of 6, 5, or 4 vehicles. There is a vehicle group containing a locomotive (VG0) and two other vehicle groups that contain only one carriage vehicle as reinforcement.

Inspecting the 7500 assignments of service blocks (and train positions) to sets of feasible vehicle groups, we observed that relatively few different sets of feasible vehicle groups occur. These are the following 13 group sets:

$\{VG0\}$	$\{VG1, VG2, VG3\}$
$\{VG4, VG5, VG6, VG7\}$	$\{VG8, VG9\}$
$\{VG10, VG11\}$	$\{VG13, ..., VG20\}$
$\{VG21, ..., VG28\}$	$\{VG29, VG30\}$
$\{VG8\}$	$\{VG4\}$
$\{VG12\}$	$\{VG1\}$
$\{VG31\}$	

Now, we introduce the notion of *equivalent vehicle groups*. Two (types of) vehicle groups VG_i and VG_j are equivalent, if and only if they occur in the same sets of feasible vehicle groups (over the whole timetable). For example, $VG2$ and $VG3$ are equivalent, as both appear only once in the occurring set of alternative groups {VG1,VG2,VG3}. However, both $VG2$ and $VG3$ are not equivalent with $VG1$, since $VG1$ additionally appears in {VG1}.

Having an equivalence relation, we can build *equivalence classes* for vehicle groups, where each class includes a maximal set of equivalent vehicle groups. As vehicle groups are combined in classes, we call such an equivalence class a *combined vehicle group*. The following classes or combined vehicle groups are built for the above case:

$[VG0]$	$[VG1]$
$[VG2, VG3]$	$[VG4]$
$[VG5, VG6, VG7]$	$[VG8]$
$[VG9]$	$[VG10, VG11]$
$[VG12]$	$[VG13, ..., VG20]$
$[VG21, ..., VG28]$	$[VG29, VG30]$
$[VG31]$	

Vehicle groups within an equivalence class are interchangeable over the whole timetable. Having two equivalent vehicle groups VG_i and VG_j, each rotation that can be served by VG_i can be served by VG_j and vice versa. Now the idea is to generate a network layer to each combined vehicle group and not to each vehicle group. For the test data of German Rail, we get 14 instead of 32 network layers, as 14 combined vehicle groups are generated out of 32 vehicle groups. This makes large-scale instances of this problem type directly solvable by optimization software and enables integrating other requirements such as adding special constraints ensuring sufficient slots for servicing at maintenance bases with a even distribution in time.

Within a solution of the resulting flow model, the fleet size on a network layer for a combined vehicle group $CombVG$ specifies the number of vehicle groups required that can be freely chosen from the vehicle groups included in $CombVG$. For instance, if the fleet size for CombVG $[VG10, VG11]$ is 9, so several solutions are equivalent, namely $(7 * VG10$ and $2 * VG11)$ or alternatively $(5 * VG10$ and $4 * VG11)$, etc.

Observing this we can let the optimizer *split* the fleet size over all vehicle groups included in a $CombVG$ by including the following constraints:

Fleet size split constraints: For each combined vehicle group $CombVG$:

$$FSZ(CombVG) = \sum_{vg \in CombVG} FSZ(vg)$$

As above, additional fleet size constraints (to each layer) sets $FSZ(CombVG)$ equal to the sum of flow values on the wrap-around period change arcs within the network layer for $CombVG$.

If it is desired to consider different homebases for vehicles separately, we can write the fleet size split constraints as follows:

$$FSZ(CombVG) = \sum_{vg \in CombVG} \sum_{hb \in HB} FSZ(vg, hb)$$

Now, the vehicle capacity constraints of the last subsection connect the fleet size variables $FSZ(vg, hb)$ (or $FSZ(vg)$) for vehicle groups to the capacities of vehicles or loco/cars. Therefore, with both types of constraints, the optimizer will split the fleet size required for a $CombVG$ (network layer) among vehicle groups included in $CombVG$ while satisfying the capacities of vehicles used by the different vehicle groups.

5 Optimization Results and Decision Support Aspects

5.1 Computational Results

Applying the network flow approach and techniques presented in the last section, the resulting mathematical models for rotation building with the above requirements could be solved efficiently for the test data of German Railways (31 different types of vehicle groups, two home bases, and 7,500 assignments of service blocks to vehicle groups). The problem instances handle periodicity of the weekly timetable and empty train movements are allowed in order to reduce the fleet sizes used. Using our aggregation of empty movements as described in 4.2, 134,643 deadhead possibilities (matches) could be reduced in a first aggregation stage to 5,861 first matches and further in a second phase to 1,661 latest first matches.

The constructed 14-layer multi-commodity network flow model with special constraints for handling several vehicle groups with shared capacities results in a mathematical model with 206,000 variables and 46,000 constraints. Depending on the level of maintenance handling, the model is solved within 3 to 10 min by ILOG CPLEX on a 1 GHz Pentium III processor.

5.2 Decision Support Tools

Since railway operations are carried out in a complex dynamic environment with rapidly changing requirements, it is often necessary that human planners adjust the plans obtained by mathematical optimization techniques. In complex

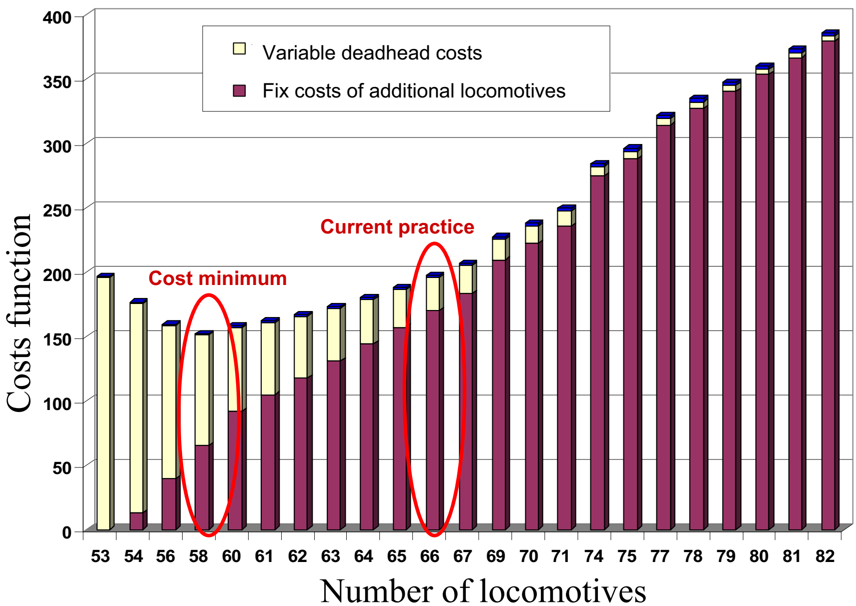

Fig. 3. Tradeoff fleet size versus empty movement costs

environments, this is only possible if optimization methods are embedded in a decision support system, providing a graphically interactive user interface which makes it easy to change the data and edit the results.

For example the following questions can be tackled within a what-if analysis: What is the practically best fleet size for a given timetable? What are the consequences of adding or deleting some trips of the timetable in terms of fleet size and operational costs? How sensitive are these costs against small changes in departure time of trips, in duration of scheduled or empty trips, or in minimum turn times required between consecutive trips? The first question assumes fixed input data (timetable, minimum turn time, empty trip duration), all others analyze changes in these input data.

Is it not sufficient, in order to answer the first question, to solve one problem instance, since the sum of fixed costs for vehicles and empty movement costs are minimized? Often, fixed costs of vehicles are set to a large value and not necessarily well and precisely scaled relatively to empty movement costs. A nice what-if analysis here is to analyze the trade-off between operative empty movement costs and the number of needed vehicles. Figure 3 shows the result of such a what-if-analysis for a timetable of German Railways with 1,098 (compound) trips and 77 terminal stations.

The analysis shows that the solution with minimum total costs is reached by a fleet of 58 locomotives. According to the used costs function, a considerable potential reduction of total fixed costs and empty movement costs is possible. An interesting aspect shown by this analysis is that empty movement costs increases non-linearly according to the number of saved vehicles. Nice to see is that these empty movement costs surmount the fixed locomotive costs for solutions with less than 58 locomotives. Thus, the minimum fleet size solution with 53 locomotives is not a cost-minimum solution.

Presenting a series of results with their properties, the decision maker chooses the best solution. Besides number of vehicles used and costs of empty movement trips incurred, other indices can be relevant such as the robustness of constructed rotations against delays or the number of induced opportunities for maintenance operations. Considering these aspects, a solution with 62-63 locomotives may be the best one from a practical point of view.

The questions related to a change in input data emphasizes the central position of timetable design in the production planning process and its interaction with rotation building. In fact, small changes in the input data may save considerable amounts in fixed costs of vehicles and empty movement costs. A useful what-if analysis here is to make some experiments changing the given minimum turn times (MTT). An analysis of this type for three fleets of German Railways is shown in Table 1:

Table 1. What-if analysis: Changing minimum turn times (MTT)

MTT (in minutes)	0	5	10	15	20	25	30	35	40
Fleet 1	34	34	34	35	35	37	37	37	38
Fleet 2	48	49	51	52	52	52	53	54	56
Fleet 3	53	54	57	61	66	68	73	75	76

Since the minimum turn time is handled as a “hard restriction” in rotation building (minimum duration between end time of one trip and start time of the next within a rotation), small changes may have considerable effect on the fleet size. Take two trips T1 from A to B and T2 from B to C. If T2 starts at 12:00, T1 arrives at 11:42 and the minimum turn-time is set to 20 minutes, then T2 is not a connection trip for T1 unless the minimum turn time is reduced to 18 minutes (or changing the duration or start times of T1 or T2). This situation is critical if no other suitable connection for T2 exists. In this case, a local change of turn-time and/or start time of T2 may save a vehicle or considerable amount of empty movements.

How to find critical locations where these savings are possible?

Finding critical locations can be supported by a chart plotting standing times (or availability) of vehicles at terminal stations. We realized a graphically interactive user-interface where these *vehicle availability charts* can be shown for all terminal stations (cf. Figure 4). An up-arrow indicates an arriving vehicle that becomes available at that station. A down-arrow indicates a departing vehicle from available ones at that point in time.

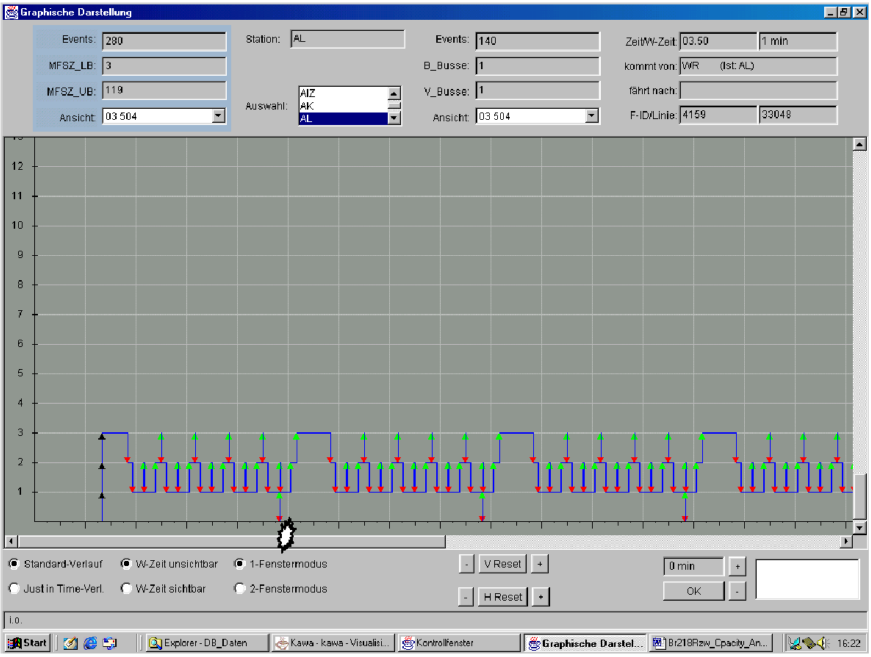

Fig. 4. Analysis of standing times

Generally long standing times within rotations which must appear in these vehicle availability charts (for some stations) constitute a strong argument that vehicles may not be utilized in an optimal way. This may appear within rotations of an optimal solution of rotation building and the cause of *practical non-optimality* may then lie in the input data themselves, and, thus concerns other planning phases, such as timetable design or trip scheduling.

In analyzing a German Railways sub-fleet, we encountered the situation given in Figure 4 at a station called AL. Among the three available vehicles at the start of the week (vertical axis), one vehicle (level 0 to level 1) is standing during the day. This vehicle is utilized at 23:01 and shortly thereafter (one minute later), another vehicle becomes available (after arrival).

Analyzing the situation at this station, we found out that two trips, say T1 and T2, arrive in AL at 22:55 and 22:58, respectively, and other two trips T3 and T4 start from AL at 23:01 and 23:10, respectively. The minimum turn time at station AL is set to 7 minutes. The vehicle serving T2 cannot serve T3 and must serve T4. The problem lies that only 6 minutes are available between arrival time of T1 and start time of T3. Since the minimum turn time is handled as a hard restriction by rotation building no connection from T1 to T3 is possible and the computed solution requires 3 instead of two vehicles. Discussing this with experts, they affirmed that local violations of the given minimum turn times are allowed in situations like this.

6 Conclusions

In this paper, we presented an exact optimization approach for rotation building for railways. In this domain, there are specific requirements related to the assembly of locomotives and carriages into train units. We discussed the case study appearing in some railways' fleet where service blocks are to be assigned to predefined groups of locomotives and carriages.

For this type of requirements, we developed new mathematical models based on time-space trip-as-arc network with aggregation schemes both for the basic and special problem. For the basic problem, an aggregation of all potential empty movements is applied which was published in former works. In this paper, we provided an extension of the multi-layer network flow model together with special constraints relating the fleet sizes expressed in number of used vehicle groups to the given capacities at locomotive and carriage level. For solving large-scale models, a special aggregation of "equivalent" vehicle groups is applied in order to reduce the number of involved network layers.

After applying the techniques described above, the resulting mixed-integer mathematical models showed a very small LP/IP gap. To our understanding, this behavior is due to the fact that the cover/partitioning constraints involved in the network flow model have much less non-zero elements than in standard set-partitioning and set-covering models. Large-scale instances of our models are solved directly using state-of-the-art mathematical optimization software.

Furthermore, we discussed some practically relevant questions related to the process of planning railway fleet and provided ways of integrating the optimization components with suitable decision support tools.

References

1. Abbink, E., van den Berg, B., Kroon, L., Salomon, M.: Allocation of railway rolling stock for passenger trains. Transportation Science 38(1), 33–41 (2004)
2. Ben-Kheder, N., Kintanar, J., Queille, C., Strainling, W.: Schedule optimization at SNCF: From conception to day of departure. Interfaces 28, 6–23 (1998)
3. Brucker, P., Hurink, J., Rolfes, Th.: Routing of Railway Carriages. Journal of global optimization 27, 313–332 (2003)
4. Cordeau, J.-F., Thoth, P., Vigo, D.: A survey of optimization models for train routing and scheduling. Transportation Science 32, 380–404 (1998)
5. Cordeau, J.-F., Soumis, F., Desrosiers, J.: A Benders decomposition approach for the locomotive and car assignment problem. Transportation Science 34(2), 133–149 (2000)
6. Cordeau, J.-F., Soumis, F., Desrosiers, J.: Simultaneous assignment of locomotives and cars to passenger trains. Operations Research 49(4), 531–548 (2001)
7. Kliewer, N., Mellouli, T., Suhl, L.: Multi-depot vehicle scheduling: a time-space network based exact optimization approach. In: Presented at the 9-th Meeting of the EURO Working Group on transportation. Bari. Italy (2002)
8. Löbel, A.: Optimal Vehicle Scheduling in Public Transit. PhD thesis, ZIB, Berlin. Shaker, Aachen (1998)

9. Mellouli, T.: A Network Flow Approach to Crew Scheduling based on an Analogy to a Train/Aircraft Maintenance Routing Problem. In: Voß, et al. (eds.) Computer-Aided Scheduling of Public Transport. LNEMS, vol. 505, pp. 91–120. Springer, Berlin (2001)
10. Mellouli, T.: Scheduling and Routing Processes in Public Transport Systems. Habilitation Thesis. University of Paderborn, Germany (2003)
11. Ramani, K.V., Mandal, B.K.: Operational planning of passenger trains in Indian Railways. Interfaces 22(2), 39–51 (1992)
12. Suhl, L.: Computer-Aided Scheduling: An Airline Perspective. Gabler–DUV, Wiesbaden (1995)
13. Suhl, L., Mellouli, T.: Requirements for, and Design of, an Operations Control System for Railways. In: Wilson, N. (ed.) Computer-Aided Transit Scheduling. LNEMS, vol. 471, pp. 371–390. Springer, Heidelberg (1999)
14. Suhl, L., Mellouli, T., Biederbick, C., Goecke, J.: Managing and preventing delays in railway traffic by simulation and optimization. In: Pursula, M., Niittymäki, J. (eds.) Mathematical methods on optimization in transportation systems. Applied Optimization, Ch. 1, vol. 48, pp. 3–16. Kluwer, Dordrecht (2001)
15. Zirati, K., Soumis, F., Desrosiers, J., Gélinas, S., Saintonge, A.: Locomotive assignment with heterogeneous consists at CN North America. European Journal of OR 97, 281–292 (1997)