

Re-Optimization of Rolling Stock Rotations

Ralf Borndörfer, Julika Mehrgardt, Markus Reuther, Thomas Schlechte and Kerstin Waas

Abstract The Rolling Stock Rotation Problem is to schedule rail vehicles in order to cover timetabled trips by a cost optimal set of vehicle rotations. The problem integrates several facets of railway optimization, such as vehicle composition, maintenance constraints, and regularity aspects. In industrial applications existing vehicle rotations often have to be re-optimized to deal with timetable changes or construction sites. We present an integrated modeling and algorithmic approach to this task as well as computational results for industrial problem instances of DB Fernverkehr AG.

1 Introduction

Rolling stock, i.e., rail vehicles, is the most expensive and limited asset of a railway company and must therefore be used efficiently. The *Rolling Stock Rotation Problem* (RSRP) deals with the cost minimal implementation of a railway timetable by constructing *rolling stock rotations* to operate passenger trips by rail vehicles. The RSRP integrates several operational requirements like *vehicle composition rules*, *maintenance constraints*, *infrastructure capacity constraints*, and *regularity requirements*.

T. Schlechte (✉) · R. Borndörfer · J. Mehrgardt · M. Reuther
Zuse Institute Berlin, Takustr. 7, 14195 Berlin, Germany
e-mail: schlechte@zib.de

R. Borndörfer
e-mail: borndoerfer@zib.de

J. Mehrgardt
e-mail: mehrgardt@zib.de

M. Reuther
e-mail: reuther@zib.de

K. Waas
DB Fernverkehr AG, Frankfurt, Germany

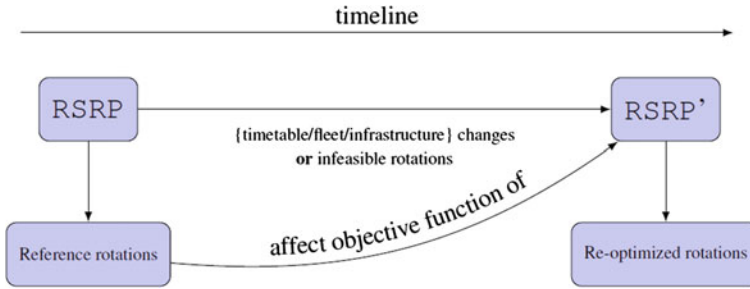


Fig. 1 Concept of re-optimization for the RSRP

A detailed problem description, a mixed integer programming formulation, and an algorithm to solve this problem in an integrated manner is described in detail in [3].

In this paper, we discuss one of the most important and challenging industrial applications of the RSRP, namely, *re-optimization*.

A re-optimization high-level concept for the RSRP is illustrated in Fig. 1. At some point in time a railway undertaking has to tackle an instance of the RSRP and constructs a solution (see boxes RSRP and Reference rotations). At another point in time this problem changes, such that the existing *reference rotation plan* can no longer be operated. Thus, a new problem RSRP has to be solved. The most important difference to the previous planning step is that much of the reference rotation plan was already implemented: Crew was scheduled for vehicle operations and maintenance tasks, capacity consumption of parking areas was reserved, and most important in a segregated railway system, e.g., in Europe and Germany: train paths were already allocated for the deadhead trips. Therefore a major goal is to change as little as possible in comparison to the reference rotation plan.

A literature overview on re-optimization can be found in [1, 2].

Re-optimization problems come up very often at a railway company. There are various causes that can lead to a situation where the implemented rotation plan becomes infeasible in an unexpected manner. Predictable and unpredictable construction sites are main causes. Fleet changes due to disruptions of operations or technical constraints, e.g., different maintenance constraints, modified speed limits for rolling stock vehicles, or changed infrastructure capacity, also ask for a modification of the vehicle rotation plans.

Depending on how large and how long the changes and their consequences are, re-optimization is required either by the dispatchers or in sufficiently lasting cases by the tactical and strategical divisions of the railway companies. In the latter case the problem is considered as a cyclic planning problem as it is introduced in [3].

The paper contributes an adaptation of the generic mixed integer programming approach presented in [3] to re-optimize rolling stock rotations. We show how to incorporate detailed re-optimization requirements into a hypergraph based formulation for rolling stock optimization by simply defining an appropriate objective function.

This paper is organized as follows. Section 2 defines the problem including an overview of our hypergraph based formulation. In Sect. 3 we introduce an objective modification procedure for the re-optimization case. Computational results in Sect. 4 show that our model and algorithm produce high quality and implementable results even for complicated re-optimization settings. Rotation planners of Deutsche Bahn validated the resulting rolling stock rotations from a detailed technical and operational point of view. It turned out that our configuration of the objective function described in Sect. 3 is sufficient and very precise for all re-optimization instances we got.

2 The Rolling Stock Rotation Problem

In this section we provide an overview on the hypergraph based rolling stock optimization model proposed in our previous paper [3]. We focus here on the main modeling ideas. For technical details including the treatment of maintenance and capacity constraints see [3]. The extension of the following problem description and model to include maintenance constraints is straight forward and does not affect the content or the contribution of the paper. Nevertheless, in our computational study we provide results for instances with maintenance constraints.

We consider a cyclic planning horizon of one *standard week*. The set of timetabled passenger trips is denoted by T . Let V be a set of *nodes* representing departures and arrivals of vehicles operating passenger trips of T , let $A \subseteq V \times V$ be a set of directed standard arcs, and $H \subseteq 2^A$ a set of *hyperarcs*. Thus, a hyperarc $h \in H$ is a set of standard arcs. The RSRP *hypergraph* is denoted by $G = (V, A, H)$. The hyperarc $h \in H$ covers $t \in T$, if each standard arc $a \in h$ represents an arc between the departure and arrival of t . We define the set of all hyperarcs that cover $t \in T$ by $H(t) \subseteq H$. By defining hyperarcs appropriately vehicle composition rules and regularity aspects can be directly handled by our model.

The RSRP is to find a cost minimal set of hyperarcs $H_0 \subseteq H$ such that each timetabled trip $t \in T$ is covered by exactly one hyperarc $h \in H_0$ and $\bigcup_{h \in H_0} a$ is a set of *rotations*, i.e., a set packing of cycles (each node is covered at most one time).

We define sets of incoming and outgoing hyperarcs of $v \in V$ in the RSRP hypergraph G as $H(v)^{in} := \{h \in H \mid \exists a \in h : a = (u, v)\}$ and $H(v)^{out} := \{h \in H \mid \exists a \in h : a = (v, w)\}$, respectively. By using a binary decision variable for each hyperarc, the RSRP can be stated as a mixed integer program as follows:

$$\min \sum_{h \in H} \mathbf{c}_h x_h, \tag{MP}$$

$$\sum_{h \in H(t)} x_h = 1 \quad \forall t \in T, \quad (1)$$

$$\sum_{h \in H(v)^{in}} x_h = \sum_{h \in H(v)^{out}} x_h \quad \forall v \in V, \quad (2)$$

$$x \in \{0, 1\}^{|H|} \quad (3)$$

The objective function of model (MP) minimizes the total cost of operating a timetable. For each trip $t \in T$ the covering constraints (1) assign exactly one hyperarc of $H(t)$ to t . The equalities (2) are flow conservation constraints for each node $v \in V$ that imply the set of cycles in the arc set A . Finally, constraints (3) state the integrality constraints for our decision variables.

3 Re-Optimization

The major re-optimization requirement for the RSRP is to change as little as possible in the reference rotation plan. We argue that this requirement can be handled by defining a suitable objective function based on the reference rotation plan.

$$\mathbf{c} : H \mapsto \mathbb{Q}_+ : \mathbf{c}(h) := \left\langle \begin{pmatrix} \mathbf{c}_1(h) \\ \mathbf{c}_2(h) \\ \mathbf{c}_3(h) \\ \mathbf{c}_4(h) \\ \mathbf{c}_5(h) \\ \mathbf{c}_6(h) \\ \mathbf{c}_7(h) \\ \mathbf{c}_8(h) \\ \mathbf{c}_9(h) \end{pmatrix}, \begin{pmatrix} p_1(h) \\ p_2(h) \\ p_3(h) \\ p_4(h) \\ p_5(h) \\ p_6(h) \\ p_7(h) \\ p_8(h) \\ p_9(h) \end{pmatrix} \right\rangle \begin{array}{l} \dots \text{ connection deviations} \\ \dots \text{ composition deviations} \\ \dots \text{ rotation deviations} \\ \dots \text{ service deviations} \\ \dots \text{ vehicles} \\ \dots \text{ services} \\ \dots \text{ deadhead distance} \\ \dots \text{ regularity} \\ \dots \text{ couplings} \end{array} \quad (4)$$

Definition (4) illustrates our approach. Our objective function is the sum of the re-optimization cost $\sum_{i=1}^4 \mathbf{c}_i p_i$ and the original objective function $\sum_{i=5}^9 \mathbf{c}_i p_i$. We propose to compute the parts of the re-optimization objective as a sum of costs depending on individual hyperarcs.

Let $h \in H$ be a hyperarc. In a first step we reinterpret h in the reference rotations, i.e., we search the timetabled trips that are connected or covered by h in the reference rotation plan, if they still exist. The reinterpretation procedure is very precise as a node in our hypergraph has the following attributes w.r.t. the vehicle traversing the node: position in a composition, orientation w.r.t. driving direction, fleet type, and rotation (i.e., cycle) of a vehicle.

In a second step we compute a property $p_i(h) \in \mathbb{N}$ for $i = 1, \dots, 4$ for $h \in H$ that states the number of differences of h w.r.t. the reference rotations. Examples for such differences are:

Table 1 Key numbers of re-optimization scenarios

Instance	Trips	Compositions	Fleets	Maintenances	$ V $	$ H $
RSRP_11	104	2	1	0	486	186,130
RSRP_12	104	2	1	1	486	192,612
RSRP_13	104	2	1	2	486	198,758
RSRP_21	805	2	2	0	9,810	15,770,498
RSRP_22	805	2	2	2	9,810	18,768,740
RSRP_31	788	2	2	0	7,776	11,727,856
RSRP_32	788	2	2	2	7,776	14,019,208
RSRP_41	789	10	4	0	16,790	42,764,116
RSRP_42	789	10	4	4	16,790	54,640,466

- Let $h \in H$ a hyperarc connecting the timetabled trips t_1 and t_2 . If t_1 and t_2 exist in the reference rotations and both trips are not connected there, we set $p_1(h) = |h|$. In all other cases we set $p_1(h) = 0$.
- If h covers trip t that exists in the reference rotations and is operated by a different vehicle composition than h , we set $p_2(h) \geq 1$, otherwise $p_2(h) = 0$. The exact numeric number depends on $|h|$, how these vehicles are oriented, which fleets are used etc.
- If h implies that t is operated in a different rotation we set $p_3(h) = |h|$, otherwise $p_3(h) = 0$.
- If h implies a different maintenance service before or after a timetabled trip $p_4(h) = 1$, otherwise $p_4(h) = 0$.

Solutions of re-optimization instances often have the characteristic that major parts of the reference rotations are not changed but some small parts have to be modified. In some cases, however, new timetabled trips have to be incorporated into the reference rotation plans. To handle this case we also have to consider properties of the original objective function $\sum_{i=5}^9 \mathbf{c}_i p_i$ for re-optimization instances, i.e., costs for vehicles consumed by a hyperarc, costs for maintenance services, costs for deadhead distances, cost for irregularities, and costs for coupling activities. Finally all of these individual properties are multiplied by individual cost parameters \mathbf{c}_i , $i = 1, \dots, 9$ that can be adjusted to the requirements of industrial use cases.

In this way we are able to handle a lot of technical re-optimization details simply by changing objective coefficients. As already mentioned, we were able to instantiate all technical re-optimization scenarios we got so far by this simple objective configuration procedure, i.e., by penalizing *local* deviations w.r.t. the reference rotations. This makes it possible to apply the general model and algorithm presented in [3] to solve re-optimization instances.

Table 2 Key numbers of re-optimization results with ROTOR 2.0 and CPLEX 12.5

Instance	Vehicles	Dev. heads	Dev. configurations	Dev. fleets	Dev. orientations	Gap (%)	hh:mm:ss
RSRP_11	9	0	0	0	0	0.00	00:03:13
RSRP_12	9	0	0	0	0	0.09	00:04:05
RSRP_13	9	1	0	0	0	1.65	00:05:35
RSRP_21	55	3	1	1	0	0.00	00:09:27
RSRP_22	55	1	0	0	0	0.15	03:34:00
RSRP_31	55	29	2	2	0	0.00	00:07:59
RSRP_32	55	30	2	2	0	0.28	01:17:17
RSRP_41	61	39	7	45	23	0.32	00:49:36
RSRP_42	59	40	7	42	17	0.91	02:48:20

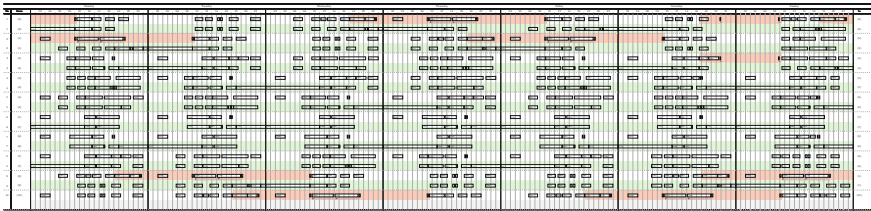


Fig. 2 Comparison of reference and re-optimized rolling stock rotations

4 Computational Results

We implemented our re-optimization model and algorithm in a computer program, called ROTOR 2.0. This implementation makes use of the commercial mixed integer programming solver CPLEX 12.5. ROTOR 2.0 is integrated in the IT environment of Deutsche Bahn. All our computations were performed on computers with an Intel(R) Xeon(R) CPU X5672 with 3.20 GHz, 12 MB cache, and 48 GB of RAM in multi thread mode with eight cores.

Table 1 lists the sizes of the instances, i.e., the number of *trips*, *compositions*, *fleets*, and *maintenance constraints*. In addition the total number of nodes ($|V|$) and hyperarcs ($|H|$) report about the size of the hypergraphs for the considered instances.

Furthermore, Table 2 provides re-optimization results. The second column reports on the number of used vehicles. The next four columns denote the number of deviations w.r.t. the reference solution introduced in Sect. 3. Finally, the last two columns show the proven worst case optimality gap and the total computation time.

The considered instances include scenarios where vehicles got broken, where the timetable was changed due to track sharing with other railway operators. And we also tackle instances where the fleet size increases, i.e., for the case when new vehicles are available and have to be integrated in the current operations. All scenarios were given by Deutsche Bahn Fernverkehr AG. Figure 2 shows a difference view of the

reference solution and the solution re-optimized with ROTOR 2.0 in green. The rows alternate between the reference solution and the re-optimized solution. The red parts of the reference solution can never be reproduced because of timetable changes.

We conclude that re-optimization instances of Deutsche Bahn Fernverkehr AG for the RSRP can be handled in great detail. On the other hand huge parts of the reference rotation plans must not be changed: See column trips w.r.t. column $\sum p_1$ (sum of connection deviations) in Table 2 and Fig. 2. This combination directly results in short computation times, high quality solutions, and therefore a powerful tool for re-optimization of rolling stock rotations at Deutsche Bahn Fernverkehr AG.

References

1. Jespersen-Groth, J., Potthoff, D., Clausen, J., Huisman, D., Kroon, L., Maróti, G., & Nielsen, M. (2009). Disruption management in passenger railway transportation. In A. RavindraK, M. RolfH, & Z. ChristosD (Eds.), *Robust and Online Large-Scale Optimization* (vol. 5868, pp. 399–421). Berlin: Springer.
2. Nielsen, L. K. (2011). Rolling stock rescheduling in passenger railways: Applications in short-term planning and in disruption management. *PhD Thesis*.
3. Reuther, M., Borndörfer, R., Schlechte, T., Waas, K., & Weider, S. (2013). Integrated optimization of rolling stock rotations for intercity railways. In *Proceedings of the 5th ISROR (Rail Copenhagen 2013)*, Copenhagen, Denmark.