

Konrad-Zuse-Zentrum
für Informationstechnik Berlin

Takustraße 7
D-14195 Berlin-Dahlem
Germany

RALF BORNDÖRFER ¹	TORSTEN KLUG ¹
LEONARDO LAMORGESE ²	CARLO MANNINO ²
MARKUS REUTHER ¹	THOMAS SCHLECHTE ¹

Recent Success Stories on Optimization of Railway Systems*

¹ Zuse Institute Berlin (ZIB), Takustr. 7, D-14195 Berlin, Germany {borndoerfer, klug, reuther, schlechte}@zib.de

² SINTEF ICT, Forskningsveien 1, Oslo, Norway

* This work has been developed within the **Forschungscampus Modal** funded by the German Ministry of Education and Research (BMBF).

Herausgegeben vom
Konrad-Zuse-Zentrum für Informationstechnik Berlin
Takustraße 7
D-14195 Berlin-Dahlem

Telefon: 030-84185-0
Telefax: 030-84185-125

e-mail: bibliothek@zib.de
URL: <http://www.zib.de>

ZIB-Report (Print) ISSN 1438-0064
ZIB-Report (Internet) ISSN 2192-7782

Recent Success Stories on Optimization of Railway Systems

Ralf Borndörfer Torsten Klug
Leonardo Lamorgese Carlo Mannino
Markus Reuther Thomas Schlechte

February 18, 2015

Abstract

Planning and operating railway transportation systems is an extremely hard task due to the combinatorial complexity of the underlying discrete optimization problems, the technical intricacies, and the immense size of the problem instances. Because of that, however, mathematical models and optimization techniques can result in large gains for both railway customers and operators, e.g., in terms of cost reductions or service quality improvements. In the last years a large and growing group of researchers in the OR community have devoted their attention to this domain developing mathematical models and optimization approaches to tackle many of the relevant problems in the railway planning process. However, there is still a gap to bridge between theory and practice (e.g. [5] and [3]), with a few notable exceptions. In this paper we address three success stories, namely, long-term freight train routing (part I), mid-term rolling stock rotation planning (part II), and real-time train dispatching (part III). In each case, we describe real-life, successful implementations. We will discuss the individual problem setting, survey the optimization literature, and focus on particular aspects addressed by the mathematical models. We demonstrate on concrete applications how mathematical optimization can support railway planning and operations. This gives proof that mathematical optimization can support the planning of rolling stock resources. Thus, mathematical models and optimization can lead to a greater efficiency of railway operations and will serve as a powerful and innovative tool to meet recent challenges of the railway industry.

1 Introduction

Planning and operating railway transportation systems is extremely hard due to the combinatorial complexity of the underlying discrete optimization problems, the technical intricacies, and the immense sizes of the problem instances. Precisely because of that, however, mathematical models and optimization techniques can result in huge gains for both railway customers and operators, e.g., in terms of cost reductions or service quality improvements. This observation is not new. In fact, railway planning was one of the originating applications for

operations research and mathematical optimization, see the account of Schrijver [30] for a historic overview from the early work of Tolstoï on augmenting cycles to the Ford and Fulkerson theory of network flows, or the article of Charnes and Miller [8] for an early set partitioning approach. Indeed, the development of many mathematical key concepts was motivated by railway applications.

One reason why little of that was put into practice is probably that for a long time railway companies operated in de facto monopolies such that there was not enough incentive to standardize and optimize the planning tasks. The airline and later also the public transportation industry, however, moved ahead in implementing optimization solutions for related problems. Their successes are a strong motivation to investigate the potential of railway optimization from today’s point of view. It is now broadly understood that the development of industry standards for railway planning and the mathematical solution of the associated optimization problems are the key to improve the efficiency of railway systems.

In the last years a large and growing group of researchers in the OR community have devoted their attention to this domain developing mathematical models and optimization approaches to tackle many of the relevant problems in the railway planning process. While there is still a gap to bridge between theory and practice, see, e.g. [5] and [3] for surveys, substantial progress is undeniable.

This paper addresses three success stories, namely:

1. long-term freight train routing (Section 2),
2. mid-term rolling stock rotation planning (Section 3),
3. and, real-time train dispatching (Section 4).

In each case, we describe real-life, successful implementations. We discuss the respective problem setting, survey the optimization literature, and focus on special aspects addressed by the mathematical models. We will demonstrate on concrete applications how mathematical optimization can support railway planning and operations.

The three example problems are at different levels of the planning process of a railway system, which are typically handled by separate companies (in Europe the railway system is segregated into train operating companies, namely, passenger and freight train operators, and railway infrastructure providers).

Figure 1 shows an idealized planning process for such a segregated railway system. In today’s practice, however, each railway company seems to have his own internal and proprietary process to organize their planning. Assumptions on essential problem characteristics as well as on principal purposes can differ strongly, e.g., between a regional passenger railway operator and an international freight train operator. This situation must be taken into account. Our experience from real world railway optimization projects have led us to identify a number of common “pitfalls”. Make sure that:

1. your mathematical model is understood (by users and managers), i.e., degrees of freedom, fixed input, output, hard rules, and soft rules,

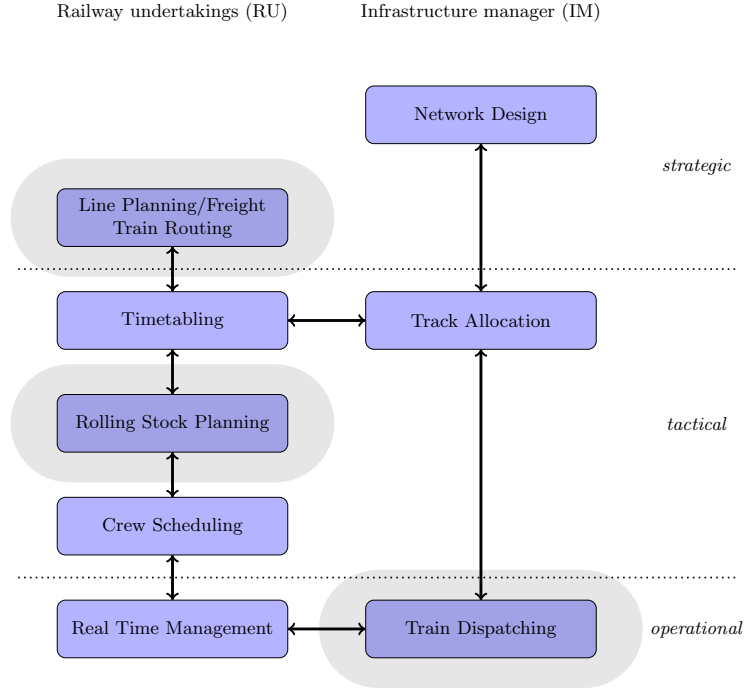


Figure 1: Idealized planning process for railway transportation in Europe based on [29].

2. you have (real and frequent) access to all relevant data of your model (data will always change),
3. the objective function of the optimization model is very flexible (goals will always change),
4. constraints are as generic as possible (rules will always change),
5. your complete interface is precise and abstract because tools around your optimization module may change over time,
6. you can measure improvements, e.g., producing better or same quality solutions as before, planning is faster, the set of considered scenarios is substantially larger, reflecting more aspects and features.

Of course, marking off the points on this list – as good as possible – is not only desirable for optimization projects in the railway industry, but can be seen as good starting point for any applied mathematical optimization project in any interdisciplinary (ad)venture.

2 Freight Train Routing

One of the first steps in the planning process of a railway company, is to find a strategic routing for an estimated demand of a transportation network. We distinguish two types of traffic. On the one hand, there is the passenger traffic with a routing that is mostly determined by political and historical presets. In addition, passenger train routes are limited by several intended intermediate stops with strict time windows, and some planned connections to other trains. It is also widely assumed that passengers expect a stable and frequently reoccurring service. On the other hand, there is the freight traffic with often less strict departure and arrival time requirements and less constrained routes. Furthermore, the freight train demand is much more volatile in comparison to the repetitive amount of passengers.

The goal for both types of traffic is to find routes for a set of origin-destination pairs that obey the network capacities and minimize the resulting cost. Typical cost functions for passenger traffic are the operating cost or the experienced traveling times. In the case of public transport this is a well studied problem, for a recent survey see [31].

Railway systems in Europe are operated by mixed traffic. Freight trains and passenger trains share in almost all areas the same infrastructure network and facilities. Thus, these two types of traffic cannot be treated individually and have to be considered in an integrated approach to provide reasonable strategic predictions. A common property of all railway networks is that changes of the infrastructure are always capital-intensive and long term projects. Hence, it is necessary to analyze the existing network in order to estimate and make the best use of the available capacity. Freight traffic has in this respect the basic advantage that upcoming demand could be much more easily distributed to available capacity.

In this paper we focus on *freight train routing* on a strategic planning level in a simplified (macroscopic) transport network. The major aim is to determine macroscopic routes for freight trains by taking the available railway infrastructure and the invariant passenger traffic into account. Literature to related problems can be found in [2].

We start with the problem description of the freight train routing problem and introduce the particular non-linear issues of our optimization model. We report on the integration of our implementation of our algorithm into the IT landscape of the traffic development and model department (GSV) of Deutsche Bahn Mobility Logistics AG.

2.1 The Problem

Given is a railway network that is utilized by a set of passenger trains and a model day that is partitioned into a few intervals. Each time slice represents a special traffic situation of the day and comprises several hours, for instances the morning and afternoon peak of a working day or the night with a rather small amount of passenger traffic. We classify the trains into different standard train

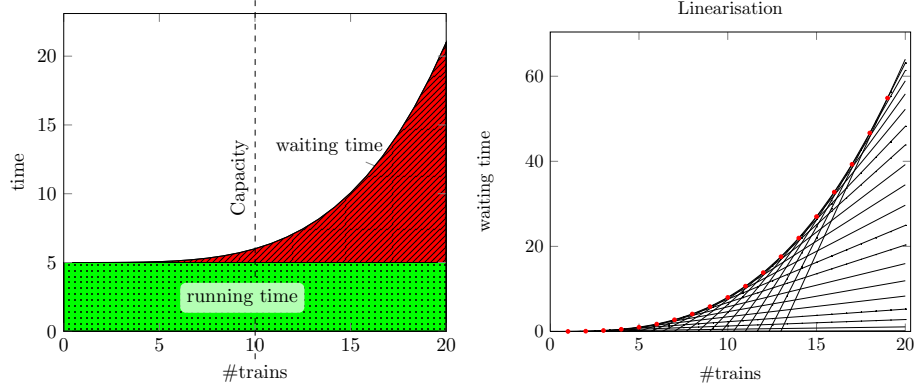


Figure 2: Left: Waiting time in dependence of the track utilization. Right: Linearization of the objective function.

types, which describe track dependent characteristics of the trains, e.g., running times, headway times, or special technical requirements for the track. The headway times define the minimal time differences between two trains entering the same track in the network. One common special requirement of a train type for a track is the electrification of the track. The preset passenger traffic is simply described by the number of trains of a specific train type for each time slice. In particular there is no information about the actual schedule of the trains. We only know that these trains consume track capacity within the corresponding time slice. On the demand side the freight trains are defined by an origin destination pair, the departure time, and a train type. The task is to find a route for each freight train that does not exceed a given distance and running time limit and minimize the expected delays. With the distance and running time limit it is possible to control the deviation from the shortest or fastest route. Since we are only interested in a strategic routing with a rough approximated timing, the minimal expected delays should ensure the existence of a feasible timetable or at least increase the possibility for one.

Still missing in this problem definition is the capacity of the tracks and it is unclear how to measure the expected delay. Modeling railway capacity is technically very complex and hence the prediction of congestion and waiting times is a major challenge. Nevertheless, the crucial relation is that there is almost no waiting time as long as the mixture of allocated trains can be handled by the infrastructure capacity. Once the capacity limit is reached congestion starts and smooth operation is not possible anymore. The closer it comes to the capacity limit, the more delay occurs for trains that are at the end of the schedule. As soon as the number of trains goes further beyond the capacity limit, the possible delay grows even faster and becomes at some point certainty. Since we only know the mixture of trains but not an actual schedule an assignment of expected delays to specific trains is not possible. The typical situation for a single track is depicted in Figure 2. The dotted (green) area is the operating mode without any disturbance where the train needs the usual running time for this track. If the number of trains increases within the time slice the risk of causing delays (waiting time) also increases. The lined (red) area in the figure shows

the additional potential waiting time. The decreasing number of fitting schedules and the reduction of buffer times leads to a nonlinear growth. Apart from that, using non-linear functions has the advantage that they achieve a balanced network utilization. Linear models will tend to route all trains on the shortest path as long as the capacity is not violated. An optimization model utilizing a nonlinear objective function results in an adequately balanced solution that distributes the traffic before the capacity limit is reached or tries to balance the excess of capacity.

There are many different ways to model and compute capacity values for tracks. For a survey on this complex issue we refer to [1] and the references therein. The approach we chose to model the functional relationship between the number of trains passing a certain infrastructure (an arc in the network model) is to introduce a *capacity restraint (CR) function*. These functions are designed to give a reasonable measure of the expected average delay. One of the earliest appearance of CR-function in the literature is due to [12]. [33] uses CR-functions to describe the travel performance or travel time and delay as a function of the flow using properties of the infrastructure and its capacity during the trip distribution and assignment phases of a travel forecasting process. Most applications of CR-functions are tailored to road traffic. Only recently, [16] use CR-functions in railway passenger transport. To the best of our knowledge, our work was the first application of CR-functions to railway freight transport.

2.2 Modeling and solving the Freight Train Routing problem

We formulate the freight train routing problem as mixed-integer non-linear program and adapt the congestion concept from road traffic to rail traffic, see [13, 12, 33, 16] using the mentioned CR-functions. We will shortly discuss some essential modeling and solving aspects. A detailed description could be found in [2].

Let n be the number of trains on a track, then the expected delay for this track is defined as:

$$\tau \left(1 + \alpha \left(\frac{n}{\kappa \gamma} \right)^\beta \right), \quad \alpha, \beta \in [0, \infty[, \gamma \in]0, \infty[, \quad (1)$$

where the running time τ and the capacity κ depends on the track. This function is an undamped variant of the CR-function presented in [16]. In this work a justification for the exponential growth of the CR-function is also given. α, β, γ are parameters to control the shape of the CR-function. α could be interpreted as the multiple of the running time that a train gets if the capacity limit is reached. γ could be used to scale the capacity, i.e., to keep an amount of auxiliary capacity. β controls the rapidness of the penalization. A large value for β results in a larger slope near the capacity. A small one leads to a moderate slope. In our experiment we choose $\alpha = 1$, which means we must pay the running time of a train if we reach the limit; $\gamma = 1$ we do not keep any auxiliary capacity; and $\beta = 4$. The choice of β is guided by the computational experiments.

The nonlinear objective can be linearized by introducing additional variables and constraints, see [10]. Thus, the resulting model is a mixed integer linear program which can be described as follows.

The transportation network is given as a time slice expanded directed graph $G = (N, A)$. A *node* $v \in \mathcal{V}$ represents a station, a junction or some other infrastructure element where train routes can start, branch or end. Additionally, there are a copies of a node for each time slice. There is a directed *arc* $a \in \mathcal{A}$ between two nodes if the corresponding infrastructure element is connected by a track or if the two nodes are copies of the same node for consecutive time slices. The set of trains is denoted by \mathcal{R} . The set of the ingoing and outgoing arcs from node v are denoted by $\delta^-(v)$ and $\delta_+(v)$, respectively.

Based on the time slice expanded graph we model the problem as a multi-commodity arc flow problem. Therefor, we introduce a binary decision variable x_a^r for each arc $a \in \mathcal{A}$ and each train $r \in \mathcal{R}$. The variable is one if and only if train r uses arc a , otherwise the variable is zero. Let $x \in \{0, 1\}^{\mathcal{A} \times \mathcal{R}}$ be the vector of these variables. We define a demand function b_v^r that is, 1 if v is the origin node of train r , -1 if v is a destination node of train r , and 0 otherwise. We introduce for each arc $a \in \mathcal{A}$ an artificial continuous variable y_a which represents the value of the expected delay of arc a .

The objective function (2) contains the expected delay cost for each arc and the sum of all running times and lengths. λ_{wait} , λ_{running} , λ_{length} are the normalized cost factors of each part. For each arc $a \in \mathcal{A}$ we have the running time $\tau_{r,a}$ for each train $r \in \mathcal{R}$, the length l_a and the average running time τ_a considering all train types.

$$\min \quad \underbrace{\lambda_{\text{wait}} \sum_{a \in \mathcal{A}} y_a}_{\text{congestion cost}} + \underbrace{\lambda_{\text{time}} \sum_{r \in \mathcal{R}} \sum_{a \in \mathcal{A}} x_a^r \tau_{r,a}}_{\text{running time}} + \underbrace{\lambda_{\text{length}} \sum_{r \in \mathcal{R}} \sum_{a \in \mathcal{A}} x_a^r l_a}_{\text{length}} \quad (2)$$

The constraints are

$$\sum_{a \in \delta_+(v)} x_a^r - \sum_{a \in \delta^-(v)} x_a^r = b_v^r \quad \forall r \in \mathcal{R} \quad \forall v \in \mathcal{V}, \quad (3)$$

$$Bx \leq l, \quad (4)$$

$$\Gamma_1^a(m) \sum_{r \in \mathcal{R}} x_a^r + \Gamma_2^a(m) \leq y_a \quad \forall a \in \mathcal{A} \quad \forall m \in \{1, \dots, N\} \quad (5)$$

$$x_a^r \in \{0, 1\} \quad \forall a \in \mathcal{A} \quad \forall r \in \mathcal{R} \quad (6)$$

$$y_a \geq 0 \quad \forall a \in \mathcal{A} \quad (7)$$

We have the common flow constraints (3) for each train: the outflow must be one at the origin; the inflow must be one at exactly one of the destination node copies in the time expanded graph; and at the remaining nodes flow conservation is required. The limits have to be fulfilled (4). Therefore a train must change to the succeeding time slice if the running time is larger than the time span of the time slice and the route must be shorter and faster than the length and running time restrictions, respectively. The corresponding coefficient matrix is denoted by B . The vector l contains the limit values for each train. Linearization

constraints of the objective function are described by (5). Let f the CR-function of arc a . $\Gamma_1^a(m)$ is the slope and $\Gamma_2^a(m)$ the y-intersection of the linear function through the points $(m, f(m))$ and $(m-1, f(m-1))$. To get an insight the right hand side of figure 2 shows the linear description of the CR-function for an arc that keeps all important integral points of the possible solution space. N is the maximal number of trains we considered for the cost function and restrict the maximal slope.

It has to be mentioned that the model does not contain explicit capacity bounds for the tracks. This constraint is solely expressed by the objective function. This is reasonable since the capacity of a track could only be estimated and nevertheless the resulting solution directly indicates detailed information where bottlenecks will occur. To reduce the formulation to a size tractable by a standard MIP solver, we restrict the network for each train to its really necessary components. This preprocessing reduced the problem size to 1-5 percent of the original size. For the exact numbers and experimental setting we refer to [2]. The reduced MIP is solved with the commercial state of the art MIP solver CPLEX via a sequential approach. In a first step, we aggregate graph structures of the macroscopic network even further by neglect arcs between different time slices or aggregate crossing areas with short tracks to a smaller structure with fewer nodes and arcs. Thereafter, we provide a heuristic start solution and let the MIP solver run. In the next step, the result of the simplified model is used as a starting point of a model for that we increase the level of detail iteratively, for instance, added arcs between time slices again or restore aggregated crossing areas. Thus, we were able to produce high-quality solutions, sometimes already optimal ones before we reached the full detail model formulation. In this cases the MIP solver needs only to prove the optimality of the solution for the full detailed model.

Based on the MIP formulation of the problem we developed a best insertion heuristic. Here, we insert the trains one by one and update the cumulative cost after each insertion. Each insertion itself finds an optimal routing for the specified train w.r.t the cost of the current train routings.

2.3 Computation and Integration

The developed MINLP approach and the best insertion heuristic was integrated into the IT landscape of the traffic development and model department (GSV) of Deutsche Bahn Mobility Logistics AG. It is used to evaluate different demand forecast scenarios and detect bottlenecks. It also allows to check how the train routings react on network expansions or cutbacks. For a given test scenario of Germany we produce solutions, see Figure 3, for the corresponding model with 1620 nodes, 5162 arcs, and 3350 trains via a best insertion heuristic. It is clear that the overall quality depends on the ordering in which the trains are inserted. In most cases the optimal solution was not found even if we consider all potential orderings. Nevertheless, the computational experiments also demonstrate that the average difference to the optimal solution is only 2 percent. We provide optimal solutions for subgraphs up to 812 nodes, 2612 arcs, and 1300 trains by solving the MINLP to optimality with CPLEX and our sequential approach.

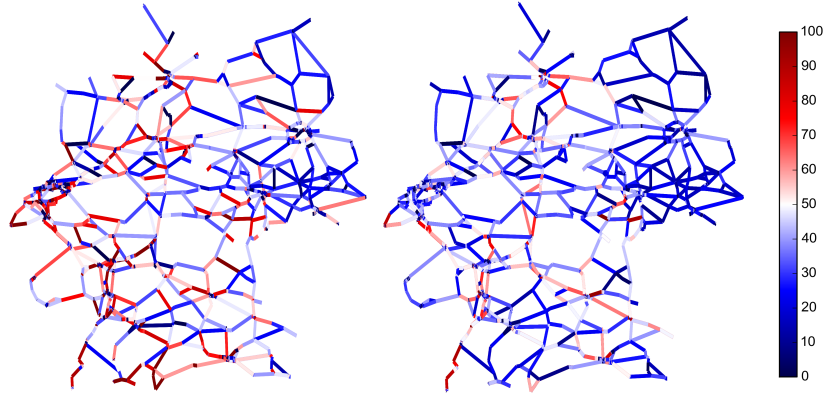


Figure 3: Utilization of the German railway network in percent of the used capacity. Left: afternoon peak between 4 p.m and 8 p.m, Right: night traffic between 8 p.m - 5 a.m.

The combination of best insertion heuristic and an optimal MINLP approach gives us a powerful tool to analyze the freight train routing in large scale railway networks. On the one hand the heuristic provides solutions for large scale networks, on the other hand the MINLP provides high quality solutions and can measure the quality of the best insertion solution.

3 Rolling Stock Rotation Optimization

The rolling stock, i.e., rail vehicles, are among the most expensive and limited assets of a railway operator. The rolling stock is needed to operate a timetable. The implementation of a timetable by a rolling stock fleet must be done in a most efficient way to be in the black.

In order to have a master plan for a medium term period – say six weeks to one year – *rolling stock rotations* are developed in a certain point of preparation. We consider a timetable in a *standard week*, i.e., the trips of the timetable repeat from week to week. Figure 4 shows a cyclic timetable that is valid on seven operating days. For each day of operation all given passenger trips are plotted as time expanded paths arranged in a torus, in which time proceeds counterclockwise. A profile at a specific time of this torus represents the current location of all vehicles operating the timetable. Furthermore, Figure 4 demonstrates that a railway timetable is almost periodic, because only a few of the given passenger trips differ from day to day. Hence, special attention has to be paid to achieve regularity of the rotations.

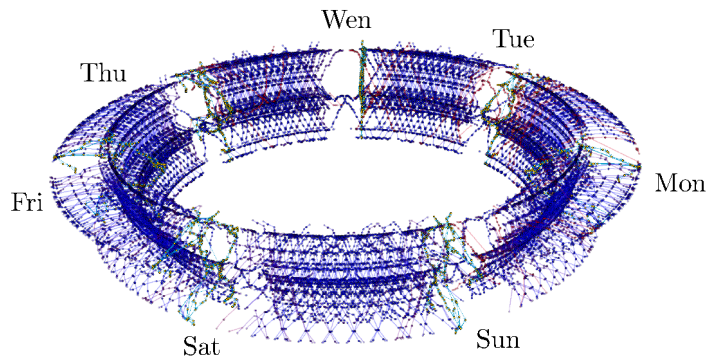


Figure 4: An almost periodic timetable for a cyclic standard week.

The rolling stock rotations are cycles that cover timetabled trips for the purpose of deciding what happens to a dedicated rail vehicle after the operation of a timetabled trip. Each of these decisions is crucial for the operational efficiency and must absolutely agree several intricate conditions:

- vehicle composition rules,
- maintenance constraints,
- infrastructure capacities, and
- regularity requirements.

This variety of requirements gives rise to a very challenging competition on rolling stock rotation planning. Our *productive optimization software Rotor* participates in this competition for one of the leading railway operators in Europe: DB Fernverkehr AG. *Rotor* is able to compute cost minimal rolling stock rotations under a large variety of requirements. In this section we survey the

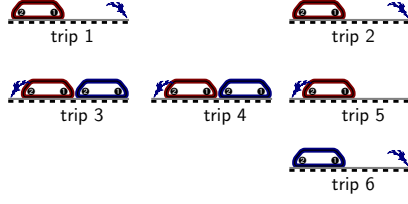


Figure 5: Timetable

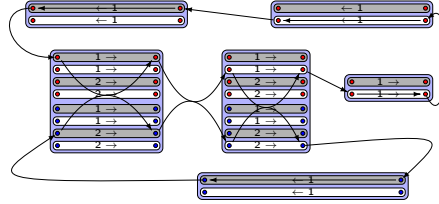


Figure 6: Hypergraph model

new technologies that have been built into **Rotor** and show the potential benefit of **Rotor** by means of a rich case study, namely “The Price of Regularity”. A literature review on rolling stock rotation planning can be found in [25].

3.1 Rolling stock rotation optimization in a nutshell

In this section we introduce the *Rolling Stock Rotation Problem* (RSRP). Particular attention is devoted to the modeling idea: Using a hypergraph as the basic structure for the optimization model. This concept “kills two birds with one stone”: The hypergraph can be easily constructed in a way such that it completely expresses all requirements for vehicle composition and also for regularity which is needed in the industrial application at DB Fernverkehr AG. Therefore, it is our *major modeling idea* which we explain in the following w.r.t. vehicle composition. How to apply this idea to regularity is proposed in Section 3.3 within the scope of a case study. We refer the reader to our paper [25] for technical details including the treatment of maintenance and capacity constraints.

Given a set of timetabled trips denoted by T we introduce the set of *nodes* V . The nodes represent timetabled departures and arrivals of *single* vehicles that operate trips of T . Further, we consider the set of *arcs* $A \subseteq V \times V$ that models connections *traversed by single vehicles* between the nodes, e.g., a connection between the departure and arrival of a timetabled trip or a connection between the arrival and departure of two different timetabled trips. Rolling stock vehicles can be composed to form *vehicle compositions* that consist of multiple vehicles. In order to handle this, we define a set $H \subseteq 2^A$ of *hyperarcs*. A hyperarc $h \in H$ is a set of standard arcs that models a connection *traversed by multiple vehicles* that form a vehicle composition. The hyperarc $h \in H$ *covers* $t \in T$ if each standard arc $a \in h$ represents an arc connecting the departure and arrival of t .

We introduce an objective function $c : H \mapsto \mathbb{Q}_+$ and define the Rolling Stock Rotation Problem (RSRP) as:

$$\min \left\{ \sum_{h \in S} c(h) \mid \underbrace{S \subseteq H}_{\text{chose hyperarcs}} \wedge \underbrace{\bigcup_{h \in S} h \text{ is a set of cycles}}_{\text{that form rotations}} \wedge \underbrace{t \text{ is covered once } \forall t \in T}_{\text{and cover the timetable}} \right\}.$$

A solution to an instance of the RSRP is a set of hyperarcs that form a set of cycles, called *(vehicle) rotations*, and cover the timetable. The RSRP is \mathcal{NP} -hard, see [11]. The proposed interpretation of hyperarcs is essential. Figures 5 and 6 provide an example for this. In Figure 5 a sketchy illustration of the input

data for the RSRP, namely the timetable and the feasible vehicle compositions to operate the trips is given. The timetable is indicated by the six railway tracks with a tree that defines the driving direction, while the feasible vehicle compositions are implied by the red and blue rail cars on the tracks. For the trips 1, 2, 5, and 6 it is only allowed to use a single vehicle of a dedicated fleet, while for trips 3 and 4 we have to operate two vehicles. In industrial use cases the individual positions of vehicles in compositions must be decided. In addition one of the two possibilities how vehicles can be placed on a railway track, namely the orientation of vehicles must be decided. The orientation of a vehicle is distinguished by the location of the first class w.r.t. the driving direction (in case of our cooperation partner DB Fernverkehr AG).

Figure 6 demonstrates how all degrees of freedom w.r.t. to the input data of Figure 5 are modeled. Each red or blue circle is a node for the departure or arrival of a red or blue rail vehicle. Each gray box that fits a pair of departure and arrival nodes indicates the operation of a timetabled trip. The numbers in these boxes denote the position at departure. The colors – white and gray – distinguish if the vehicle departs with first or second class in front.

The hyperarcs in Figure 6 define a solution. In this solution we decided to operate trip 1 with a single vehicle that departs with the second class in front. Trip 3 will be operated with a vehicle composition of a red and a blue vehicle. The red vehicle departs at position one with the second class in front and the blue vehicle departs at position two with the same orientation. By applying this interpretation to all trips the two formed cycles, namely the vehicle rotations, for the red and blue vehicles appear.

The hypergraph can easily be used to express several situations that only arise in railway applications compared to the operation of buses, airplanes, cars, trucks, and ships. In the following we give a set of examples that occur in industrial use cases:

- If the passenger platforms along a trip allow different vehicle compositions we can easily express this by introducing appropriate hyperarcs.
- If the time between the arrival of trip 3 and trip 4 is very short, it is not possible to perform any shunting or coupling activities such that the connection can only be made by both vehicles in the composition. We can easily model this by excluding all arcs that imply a coupling after trip 3 or before trip 4.
- Suppose we have a large distance, e.g., 100 km, and enough time, e.g., three hours, between the arrival of trip 3 and the departure of trip 4. Assume that the cost for allocating a deadhead trip per kilometer is 100 Euros for a vehicle composition of one vehicle and 120 Euros for a vehicle composition of two vehicles. The ratio $100/120$ is approximately the one in the industry. We can directly express this cost structure by the cost function of our hypergraph. Two individual deadhead trips would increase the objective function by 20.000 Euros, while one coupled deadhead trip is only about 12.000 Euros.
- If trip 2 arrives at a terminus station (see left of Figure 7) and trip 1 departs at the same station, any vehicle composition turns around. This



Figure 7: Left: Topology of a terminus station; Right: Tracks for an additional turn around trip

can be easily modeled by only allowing the one hyperarc that correctly reflects the implications of the topology of the railway infrastructure.

- In some situations it is necessary to perform an *additional turn around trip* (see right of Figure 7), e.g., if two trips must be connected and operated with the same orientation, but the railway infrastructure implies different orientations as in the previous example. Such additional turn around trips are only constrained by a minimum amount of time to perform the turn around, but they are expensive and undesirable and play a central role in our case study. This too can be easily modeled by expensive hyperarcs that implement an additional turn around trip.

3.2 An integrated algorithm for the RSRP

We solve the RSRP by an integrated algorithm that is based on integer programming (IP). In this section we present our IP model and survey the main algorithmic components to solve the RSRP. We define sets of hyperarcs coming into and going out of $v \in V$ as $H(v)^{\text{in}} := \{h \in H \mid \exists a \in h : a = (u, v)\}$ and $H(v)^{\text{out}} := \{h \in H \mid \exists a \in h : a = (v, w)\}$, respectively. Using a binary decision variable for each hyperarc, the RSRP can be stated as an IP as follows:

$$\min \sum_{h \in H} c_h x_h, \quad (\text{MP})$$

$$\sum_{h \in H : h \text{ covers } t} x_h = 1 \quad \forall t \in T, \quad (8)$$

$$\sum_{h \in H(v)^{\text{in}}} x_h = \sum_{h \in H(v)^{\text{out}}} x_h \quad \forall v \in V, \quad (9)$$

$$x_h \in \{0, 1\} \quad \forall h \in H. \quad (10)$$

The objective function of model (MP) minimizes the total cost of the chosen hyperarcs. For each trip $t \in T$ the covering constraints (8) assign one hyperarc to each timetabled trip. The equations (9) are flow conservation constraints for each node $v \in V$ that define a set of cycles – the vehicle rotations – of arcs of A . Finally, (10) states the integrality constraints for our decision variables. This model was introduced in [26] for the first time. In the application at DB Fernverkehr AG it is necessary to also consider maintenance and capacity constraints. A direct extension of the introduced model for these requirements is given in [27]. Another important feature – if not the most important – is to re-optimize an already implemented but not completely feasible rotation plan, i.e., the re-optimization case. **Rotor** is also able to handle those scenarios. The



Figure 8: Most irregular operation of train 66

procedure for that purpose that simply adjusts the objective function of the proposed model is described in [23].

We solve this model by a novel integer and linear programming approach that combines several algorithmic “workhorses”. The linear relaxation of the hypergraph based model is solved by a Coarse-To-Fine approach [24] that aims to restrict the search space as much as possible and as large as necessary. This is done by utilizing problem specific *layers* that are associated with different levels of detail of the RSRP. To find integer feasible solutions we use the Rapid-Branching scheme [4] as well as problem specific local search heuristics [25, 22]. All of the introduced models and algorithms are implemented in our [ROT]ation [O]ptimizer for [R]ailways **Rotor**.

3.3 Case study: The price of regularity

As already mentioned we use the hypergraph to model vehicle compositions, but also for a requirement that is called *regularity*. In this section we take a closer look at regularity and provide a case study to underline the benefit of using **Rotor** in the railway industry.

In our context a *train* is a set of at most seven timetabled trips which are associated with the seven days of operation of the standard week. In instances of the RSRP provided by DB Fernverkehr AG, in most of all cases the trips of a single train only differ w.r.t. the day of operation. This means that the departure and arrival locations and corresponding times are equal. This is made to provide a timetable that is periodic (or regular) to the passengers. It also yields a quality measure for a timetable, i.e., to operate trains that do not differ over the operational days and operate on all seven days is a desired result in timetabling. This also transfers to rotation planning. In Figure 8 we consider a train with train number 66. In this example train 66 is operated by seven different vehicle compositions on each day of operation and shows the most irregular case. In rolling stock rotation planning it is desired to operate train 66 with the same vehicle composition at each day of operation. We handle this by introducing appropriate hyperarcs. Let π be one of the eight possible vehicle compositions to operate a trip of train 66 with a composition of a red and a blue vehicle. Thus, π defines the position and orientation of the red as well as the blue vehicle. Suppose that $h_{\text{Mon}}^\pi, \dots, h_{\text{Sun}}^\pi$ are those hyperarcs that model the operation of the seven timetabled trips w.r.t. π . We penalize the choice of $h_{\text{Mon}}^\pi, \dots, h_{\text{Sun}}^\pi$ by a constant factor $R \in \mathbb{Q}_+$ in the objective function and introduce:

$$h_{66}^\pi := \bigcup_{\text{day}=\text{Mon}}^{\text{Sun}} h_{\text{day}}^\pi \quad \text{with} \quad c(h_{66}^\pi) := \sum_{\text{day}=\text{Mon}}^{\text{Sun}} (c(h_{\text{day}}^\pi) - R).$$

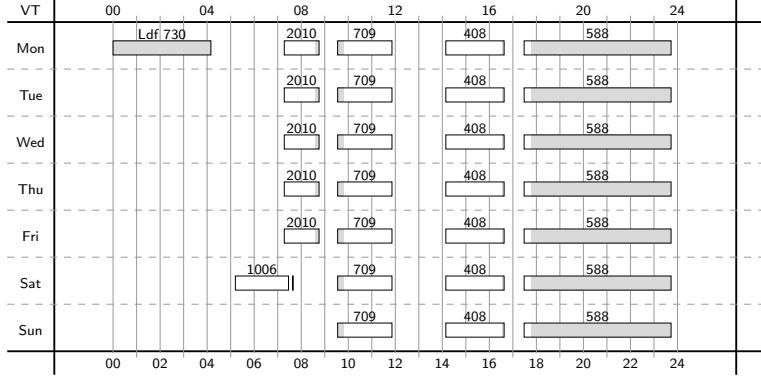


Figure 9: A sophisticated regularity pattern: rotation day

By this definition it is cheaper to choose the *regular hyperarc* h_{66}^π for the operation of train 66 compared to the other seven individual hyperarcs that may differ in their vehicle composition in a solution. These regular hyperarcs are introduced for all possible vehicle compositions of a train.

Figure 9 shows an example of a regularity pattern for this purpose, called *rotation day*. It consists of seven paths that appear in rolling stock rotations. Each path is associated with one day of operation. The figure illustrates the two aspects that are desired in a rotation day: Almost all of the seven paths contain the same train numbers and almost all connections between trains are equal. This builds an assembly of the rolling stock rotations that can be easier processed in further planning steps in comparison to seven completely different paths.

In this case study we investigate the price of regularity by a multi-criteria optimization [9]. This means, that we compute all Pareto-optimal solutions of the following two objective functions:

1. Minimize the operational cost that includes:
 - cost for rolling stock vehicles
 - cost for deadhead trips
 - cost for additional turn around trips
 - cost for violating planned turn times [25]
2. Maximize regularity (i.e., minimize irregularity)

We use the weighted-sum [9] algorithm for this multi-criteria optimization. In a nutshell, this algorithm starts with two solutions that only minimize one of the two objective functions and refines the Pareto-front in a recursion tree. In each step of computation, the value of R is chosen by a gradient argument in the weighted-sum method. A solution of the RSRP w.r.t. the dedicated choice of R is then computed by **Rotor**. Therefore, this multi-criteria approach runs on top of **Rotor**.

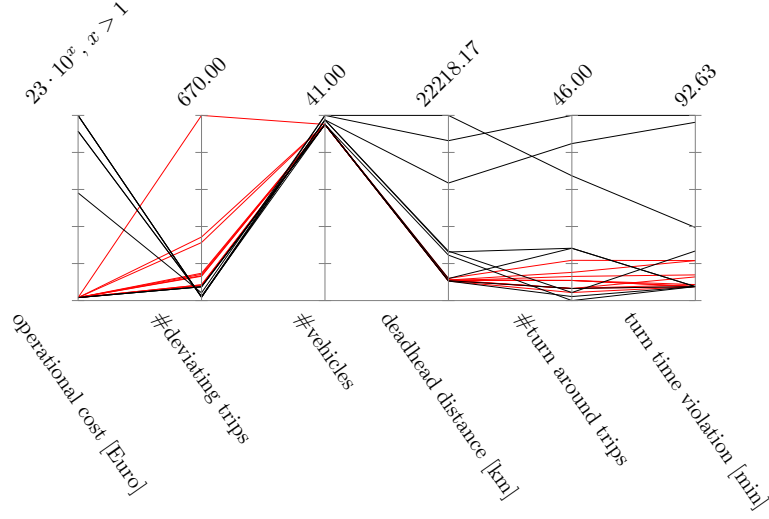


Figure 10: The price of regularity

We consider a dedicated instance provided by our industrial cooperation partner DB Fernverkehr AG. The instance consists of 670 timetabled trips that are widely spread over the German intercity network. At 52 locations passengers can get on a train or disembark from a train. The instance contains 127 trains overall. Possible vehicle compositions are composed of at most two vehicles of the same fleet. The number of hyperarcs in the resulting hypergraph is 4.946.356.

Figure 10 shows the results of the multi-criteria optimization. We obtained eight Pareto-optimal solutions which are indicated by the black paths in Figure 10. Each Pareto-optimal solution has either less operational cost (which are confidential and therefore obfuscated) or less deviating trips compared to all other solutions, i.e., it is non-dominated. The solutions indicated by the red paths are dominated in one objective function by another solution. Those solutions appeared during the computation of the set of Pareto-optimal solutions. We precisely observe:

- The number of vehicles needed to operate the timetable is almost constant if we vary the amount of regularity in the rolling stock rotations.
- The number of additional turn around trips varies – from zero to 46. This is directly related to the deadhead distance. Using 46 turn around trips for this instance is practically not implementable, it is much too much.
- Also the turn time violation [25] varies – from 10 minutes overall to almost 93 minutes. From an industrial point of view we claim that a turn time violation of 93 minutes is not much for the considered instance.

What is the price of regularity? The higher the amount of regularity in the rolling stock rotations, the higher the number of additional turn around trips to appropriate equal vehicle compositions for trains. Turn around trips are expensive and very unattractive. Irregularities are also to be avoided as much as possible in operation. Therefore, finding a good compromise between the two

objectives is a very challenging task for railway operators. Furthermore, this offers the important insight, that an optimization software can be very beneficial to handle the complexity in the railway competition in order to simulate "what if" real-world scenarios.

4 Real-Time Train Dispatching

Trains follow predetermined routes in the railway network, temporarily occupying a sequence of railway resources. Train companies and network operators produce and deploy an official timetable which drivers are obliged to follow. Designing a timetable is a complex and extensive process and an entire branch of railway optimization is dedicated to this ([7, 17, 29, 6, 32]). In principle, such timetable should ensure that no two trains will occupy simultaneously the same railway resource or different but incompatible resources. However, due to unpredictable events such as train delays, network failures, cancellations and so forth, actual train schedules may deviate from the official timetable. When these deviations occur, conflicts in the use of such resources may arise. To prevent conflicts from occurring, re-routing and re-scheduling decisions must be taken in real-time so as to minimize some objective, typically a function of the delays. We refer to this optimization problem as the Train Dispatching (TD) problem ([28]). In spite of its relevance and computational complexity, until recently train dispatching has been basically entirely in the hands of human operators (*dispatchers*), with some support by software tools and remote equipments, so called *Train Management Systems* (TMSs). Dispatchers have to take decisions in a few seconds which affect the entire network, in some cases for many hours to come. The main source of complexity are decisions of type "who goes first". In any approach to TD, these must be tackled to avoid potential conflicts in the use of unsharable resources. In the corresponding optimization problem, such decisions give rise to so-called "disjunctive" constraints, well-known to be very challenging to handle in practice. A further source of complexity stems from the often very large number of potential routes for each train. Recent papers and applications show how this problem can be effectively tackled with the use of suitable optimization techniques, leading, in most cases, to improved solutions w.r.t. those currently carried out, both in terms of quality and computation time. Although many TMSs actually include estimates on train movements and conflict identification, few are able to take re-scheduling and re-routing decisions, let alone incorporate advanced optimization algorithms. An overview of related literature is out of the scope of this paper, for a recent survey see [5].

4.1 Modeling and solving the Train Dispatching problem

A railway network is generally a complex, interconnected system that can contain large stations and several parallel tracks between stations, plus other installations such as sidings and cross-overs. A typical way to represent the railway is by a graph, where nodes correspond to resources where trains can stop and perform different activities, and arcs correspond to tracks connecting such resources. With this notation, stations correspond to sub-graphs of the overall network. What is important to note is that the railway infrastructure can be decomposed in *atomic resources*, namely resources that are occupiable by at most one train at the time. A train's route is representable as a sequence of atomic resources. Atomic resources can in turn be labeled as *station resources*, and *line resources* (which connect different stations). Stations and lines share a small, specific subset of atomic resources, namely the entry and exit points

to/from stations.

We now sketch our approach to tackle TD (for deeper insight, we refer to [14, 15]). With every train i and every atomic resource r on its route, we associate a continuous variable t_{ir} which represents the time at which i enters r . The vector t is called *schedule*. The sub-vector t^T of t , corresponding to the entry and exit points of stations, is the *real-time timetable*.

Informally, we can state TD as follows:

The Train Dispatching problem. *Given a railway network G and its status, and a set of trains with their current position, find, in real-time, a route and a conflict-free schedule such that the cost function $c(t)$ is minimized.*

Note that in general the objective function $c(t)$ may vary. However, conformity to the official timetable is generally identified as the main factor in determining the quality of the real-time schedule t . Next, we define two sets of binary variables x and y . Vector x is the *routing vector*, and represents the choice among alternative atomic resources for train routes. Vector y determines "who goes first" between pairs of trains accessing the same atomic resource.

TD can be formulated in terms of such variables as the following MIP program:

$$\begin{aligned}
& \min c(t^T) \\
& s.t. \\
& (i) \quad A^l t^l + B^l x^l + C^l y^l + P^l t^T \geq d^l, \\
& (ii) \quad Q^l t^T + A^s t^s + B^s x^s + C^s y^s \geq d^s, \\
& (iii) \quad t \text{ real, } x, y \text{ binary}
\end{aligned} \tag{11}$$

where t^T, t^l, x^l, y^l are the variables associated with line resources and t^T, t^s, x^s, y^s are the variables associated with the station resources. These two sets of variables only share the subset of scheduling variables t^T , namely the real-time timetable. This block structure can be exploited in a decomposition approach. To this end, let the *Line Dispatching* problem (LD) be obtained from (11) by dropping constraints (11.ii), and let $\bar{t}^T, \bar{t}^l, \bar{x}^l, \bar{y}^l$ be the optimal solution to LD. If there exists a feasible solution $\tilde{t}^T, \tilde{t}^s, \tilde{x}^s, \tilde{y}^s$ to constraints (11.ii) such that $\tilde{t}^T = \bar{t}^T$, then $(t^*, x^*, y^*) = (\bar{t}^T, \bar{t}^l, \tilde{t}^s, \bar{x}^l, \tilde{x}^s, \bar{y}^l, \tilde{y}^s)$ is an optimal solution for (11).

A general scheme for tackling TD is as follows. Find a feasible, possibly optimal solution to LD. If none is found, declare the problem infeasible. If one is found, extend it to a feasible solution for system (11.ii). If, in turn, no feasible solution to (11.ii) is found, then "slightly modify" the previous LD problem and iterate. Depending on how the steps of this iterative scheme are applied, the overall approach may be exact or approximate. In either case, it is important to observe that feasibility problem (11.ii) further decomposes into independent sub-problems, one for each station. We refer to any such sub-problem as a *Station Dispatching* problem (SD), which amounts to finding routes and schedules

for trains in a station, given their arrival and departure times. An exact implementation of the above scheme is presented in [14, 15], where the modification of LD is carried out by adding suitable (valid) combinatorial cuts. Heuristic implementations of such scheme have been proposed by various authors (for an extensive survey see [5]). Remarkably, the dispatching process typically carried out "manually" by dispatchers can also somehow be seen as an implementation of such iterative scheme, where conflicts are solved sequentially in a "greedy" fashion.

4.2 Applications

Although for many years real-time train dispatching has been considered an impossible task for automated systems to carry out, this trend has recently changed. A first example notable example is [20], where an automated dispatching system with limited routing options was put in operation in . More recently, TMSs embedding optimization are being (or are soon to be) deployed on several lines in Europe. More in general, a growing awareness towards the potential of optimization-based TMSs is tangible. Infrastructure managers and operators around Europe are starting to explicitly request the use of optimization modules within TMSs, as seen in recent tenders (e.g. Denmark). This in itself can already be seen as an important result and an example of how the use of OR techniques can have an impact on real-life applications.

We now give an outline of our "success stories", those innovative applications which deploy optimization to tackle real-time train dispatching. For an in-depth description of the underlying optimization models and algorithms, we refer to [14, 15, 18].

Before giving details of the single applications, let us sketch the interaction between TMS, optimization modules and dispatchers, which is the common feature among such applications: first of all, the TMS acquires real-time information regarding the status of the network (e.g. train positions, speeds, resource availability etc). This information is fed into the optimization modules, which, combining it with the required "static" information (e.g. network layout, train connections), return one or more solutions to the current dispatching problem. Dispatchers may also "manually" interact with the systems providing further information (e.g. train delays or cancellations, network disruptions, fixed meeting points).

Aside from this common trait, each of such optimization-based TMS has a set of distinct features that characterizes it. A first, natural distinction is between *Mass Transit* and *Main Line* applications. We refer to Mass Transit as a railway system generally devoted to urban public transport services (such as subways or rapid transit), as opposed to Main Line, which forms the principal railway arteries connecting different cities and urban areas. A second, important feature, particularly from an OR perspective, is the optimization technique implemented in each application. Although the iterative decomposition scheme introduced in the previous subsection is valid for all applications, its various steps were, or often had to be, implemented differently case by case. The reasons behind this vary, although generally boil down to local operative rules which restrict

the nominal space of solutions to a subset of these that can be carried out in practice. A third and final feature concerns the level of automation of the whole *dispatching loop* (1. conflict identification 2. solution 3. validation), which, again, is generally dependent on local operative rules. Solutions returned by the optimization module may either have to be validated by dispatchers (*semi-automation*) or forwarded directly to the signalling system (*full-automation*). This distinction however is in form rather than substance: statistics from our applications show that even where semi-automated systems are deployed, dispatchers still follow optimized suggestions in almost all cases (94% of times on average [15]).

Main Line The first Main Line TMS embedded with our optimization algorithms was put in operation in Italy in 2011, on a single-track regional line (Trento - Bassano del Grappa). The use of the tool was later extended to other lines in Northern, Southern and Central Italy, such as Milano - Mortara, Piraineto - Trapani - Alcamo, Orte - Terontola - Falconara and others. Table 1 reports some data on these lines.

Line	Abbr.	Stops	Stations	Length (m)	Tracks
Trento - Bassano	T-BG	22	14	95711	Single
Piraineto - Trapani	P-T	12	12	93532	Single
Alcamo - Trapani	A-T	14	13	116119	Single
Orte - Terontola - Falconara	O-T-F	54	34	283839	Mixed

Table 1: Infrastructure details. Single stands for Single-Track, Mixed stands for Single and Double-Track stands for Double-Track

This Main Line TMS was developed by Bombardier Transportation and deploys a heuristic algorithm as optimization module. The dispatching loop is semi-automated: the algorithm finds alternative solutions each time it is called and presents them to the dispatcher(s) ranked by cost. Statistics show that in most cases the first solution proposed by the algorithm is accepted. Although validation is left to dispatchers, the tool embeds an exact procedure for detecting potential deadlock situations. The choice of deploying a heuristic algorithm and semi-automated dispatching loop is a consequence of the infrastructure manager RFI's (Rete Ferroviaria Italiana) operative rules. In a recent work [15] we tried to quantify the possible impact of relaxing such rules by comparing the performance (in terms of average punctuality) of the heuristic and the exact algorithm on a set of real-life instances from the O-T-F line in Central Italy. Results showed a significant average increase of trains on time (+10%), indicating how, even though the current system represents an important innovation in railway traffic management, results could be further improved allowing a change in regulations.

In a project involving the Norwegian infrastructure manager (JBV) and train operating companies (NSB, FlyToget, CargoNet), we developed a new Main Line dispatching system which was initially tested (Trondheim-Dombås, Stavanger - Moi) then put in operation (Stavanger - Moi, Figure 11) in February 2014.

This is the only case of real-life Main Line dispatching being supported by an exact optimization algorithm ([21]), which allows the possibility of exploring the

Eastern Latvia: Daugavpils-Eglaine, Daugavpils-Krustpils, Rezekne-Krustpils, Zilupe-Krustpils, Karzava-Rezekne, a total of 52 stations, with 10 communication points and 8 station gates. These lines are mainly used for freight transportation and run around 100 trains every 20 hours. As in Italy, the optimization module is a heuristic based on local operative rules. In this case however, dispatching decisions regarding freight trains are automatically implemented by the TMS via the signaling system, without requiring any human validation (full automation). Validation is on the other hand required for solving conflicts involving passenger trains (few on these lines).

So far we have discussed dispatching on railways connecting different nodes or cities. However, optimization algorithms have proven to be very useful also for TMSs dedicated solely to large stations. The TMS in Roma Tiburtina station has been equipped with our algorithms which re-schedule and re-route trains inside the station according to the real-time timetable and current resource occupation. The optimization module includes both heuristic and exact algorithms, with the former proving to work very well (average relative gap 0.6%). This system was first released in February 2014 and is scheduled to be fully operative at the beginning of 2015. With its 12 line points, 30 stopping points and 62 interlocking routes, Tiburtina station is considered one of the most important and complex stations in Italy. Aside from its sheer size, this is also due to geographic reasons and to its role as interface between a number of different rail services, both passenger (local, regional, high speed) and freight. A similar TMS is scheduled to be released in Monfalcone, a *multi-station*¹ in Northern Italy in November 2014.

Mass Transit An optimization-based TMS, developed in cooperation with Bombardier Transportation, operated in some terminal stations of the Milano Underground between 2007 and 2009 ([19]). Such TMS embedded an exact optimization algorithm based on branch and bound and was fully automated. An extensive on-field test campaign was carried out before deploying the system, with the purpose of comparing the system's performance with that of the dispatchers operating in the terminal stations at the time. Although the size of these stations was relatively small, results showed how the system performed, on average, 8% better than dispatchers in terms of the two relevant objectives (deviations from timetable and regularity). These tests present a rare example in the literature of a direct "man vs machine" performance comparison. Despite its proven effectiveness, the system was dismantled in 2009 when Bombardier lost the tender for the renovation of Milano Underground's signaling system.

¹The Monfalcone node consists of Monfalcone station, Ronchi Nord station and Ronchi Sud station and a communication point, all of which have to be controlled simultaneously.

5 Conclusion

In this paper we addressed three different success stories – one on long-term freight train routing, another one on mid-term rolling stock rotation planning, and one on real-time train dispatching, where mathematical optimization is the key technology of a decision support systems. In each case, we describe real-life requirements, modeling aspects, and the basic algorithmic solution approaches of our successful productive implementations. This demonstrates that mathematical optimization can support the planning and operation of railway systems. Thus, mathematical models and optimization can lead to a greater efficiency of railway operations and will serve as a powerful and innovative tool to meet recent challenges of the railway industry.

References

- [1] M. Abril, F. Barber, L. Ingolotti, M. A. Salido, P. Tormos, and A. Lova. An assessment of railway capacity. *Transportation Research E*, 44:774 – 806, 2008.
- [2] Ralf Borndörfer, Armin Fügenschuh, Torsten Klug, Thilo Schang, Thomas Schlechte, and Hanno Schülldorf. The freight train routing problem. Technical Report 13-36, ZIB, Takustr.7, 14195 Berlin, 2013. in review for Transportation Science.
- [3] Ralf Borndörfer, Martin Grötschel, and Ulrich Jäger. Planning problems in public transit. In Martin Grötschel, Klaus Lucas, and Volker Mehrmann, editors, *Production Factor Mathematics*, pages 95–122. acatech/Springer, Berlin Heidelberg, 2010. ZIB Report 09-13.
- [4] Ralf Borndörfer, Andreas Löbel, Markus Reuther, Thomas Schlechte, and Steffen Weider. Rapid branching. *Public Transport*, pages 1–21, 2013.
- [5] V. Cacchiani, D. Huisman, M. Kidd, L. Kroon, P. Toth, L. Veelenturf, and J. Wagenaar. An overview of recovery models and algorithms for real-time railway rescheduling. *Transportation Research Part B*, 63:15–37, 2014.
- [6] V. Cacchiani and P. Toth. Nominal and robust train timetabling problems. *European Journal of Operational Research*, 219(3):727–737, 2012.
- [7] Gabrio Caimi. *Algorithmic decision support for train scheduling in a large and highly utilised railway network*. PhD thesis, ETH Zürich, 2009.
- [8] A. Charnes and M.H. Miller. A model for the optimal programming of railway freight train movements. *Management Science*, 3(1):74–92, 1956.
- [9] M. Ehrgott. *Multicriteria Optimization*. Lecture notes in economics and mathematical systems. Springer, 2005.
- [10] Armin Fügenschuh, Henning Homfeld, Hanno Schülldorf, and Stefan Vigerske. Mixed-integer nonlinear problems in transportation applications. In H. Rodrigues et al., editor, *Proceedings of the 2nd International Conference on Engineering Optimization (+CD-rom)*, 2010.

- [11] Olga Heismann. *The Hypergraph Assignment Problem*. PhD thesis, Technische Universität Berlin, 2014.
- [12] N.A. Irwin and H.G. Von Cube. Capacity restraint in multi-travel mode assignment programs. *Highway Research Board Bulletin*, 347:258 – 287, 1962.
- [13] Ekkehard Köhler, Rolf H. Möhring, and Martin Skutella. Traffic networks and flows over time. In Jürgen Lerner, Dorothea Wagner, and Katharina A. Zweig, editors, *Algorithmics of Large and Complex Networks: Design, Analysis, and Simulation*, volume 5515 of *Lecture Notes in Computer Science*, pages 166–196. Springer, 2009.
- [14] L. Lamorgese and C. Mannino. An exact decomposition approach for the real-time train dispatching problem. *Operations Research*, published online, 2015.
- [15] Leonardo Lamorgese, Carlo Mannino, and Mauro Piacentini. Optimal train dispatching by benders’-like reformulation. *Transportation Science*, to appear, 2015.
- [16] J. Lieberherr and E. Pritscher. Capacity-restraint railway transport assignment at SBB-Passenger. In *Proceedings of the 12th Swiss Transport Research Conference*, 2012.
- [17] Richard Lusby, Jesper Larsen, Matthias Ehrgott, and David Ryan. Railway track allocation: models and methods. *OR Spectrum*, December 2009.
- [18] C. Mannino. Real-time traffic control in railway systems. In Alberto Caprara and Spyros Kontogiannis, editors, *11th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems*, volume 20 of *OpenAccess Series in Informatics (OASICS)*, pages 1–14, Dagstuhl, Germany, 2011. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.
- [19] C. Mannino and A. Mascis. Optimal real-time traffic control in metro stations. *Operations Research*, 02/2009:1026–1039, 2009.
- [20] M. Montigel. Operations control system in the lötschberg base tunnel. *RTR - European RailTechnology Review*, page 32, 2009.
- [21] P. Pellegrini, J. Marlière, and D. Rodriguez. Optimal train routing and scheduling for managing traffic perturbations in complex junctions. *Transportation Research Part B*, 59:58–80, 2014.
- [22] Markus Reuther. Local Search for the Resource Constrained Assignment Problem. In Stefan Funke and Matúš Mihalák, editors, *14th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems*, volume 42 of *OpenAccess Series in Informatics (OASICS)*, pages 62–78, Dagstuhl, Germany, 2014. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.
- [23] Markus Reuther, Ralf Borndörfer, Julika Mehrgardt, Thomas Schlechte, and Kerstin Waas. Re-optimization of rolling stock rotations. In *Operations Research Proceedings 2013*, pages 49 – 56, 2014.

- [24] Markus Reuther, Ralf Borndörfer, and Thomas Schlechte. A coarse-to-fine approach to the railway rolling stock rotation problem. In *14th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems, ATMOS 2014, September 11, 2014, Wroclaw, Poland*, pages 79–91, 2014.
- [25] Markus Reuther, Ralf Borndörfer, Thomas Schlechte, Kerstin Waas, and Steffen Weider. Integrated optimization of rolling stock rotations for intercity railways. In *Proceedings of the 5th ISROR (RailCopenhagen2013)*, Copenhagen, Denmark, May 2013.
- [26] Markus Reuther, Ralf Borndörfer, Thomas Schlechte, and Steffen Weider. A Hypergraph Model for Railway Vehicle Rotation Planning. In Alberto Caprara and Spyros Kontogiannis, editors, *11th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems*, volume 20 of *OpenAccess Series in Informatics (OASICS)*, pages 146–155, Dagstuhl, Germany, 2011. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.
- [27] Markus Reuther, Ralf Borndörfer, Thomas Schlechte, and Steffen Weider. Vehicle rotation planning for intercity railways. In J.C. Muñoz and S. Voš, editors, *Proceedings of Conference on Advanced Systems for Public Transport 2012 (CASPT12)*, 2012.
- [28] G Sahin, R.K. Ahuja, and C.B. Cunha. Integer programming based approached for the train dispatching problem. Technical report, Sabanci University, 2010.
- [29] Thomas Schlechte. *Railway Track Allocation - Models and Algorithms*. PhD thesis, Technische Universität Berlin, 2012.
- [30] Alexander Schrijver. On the history of the transportation and maximum flow problems. *Mathematical Programming*, 91(3):437–445, 2002.
- [31] Anita Schöbel. Line planning in public transportation: models and methods. *OR Spectrum*, 34(3):491–510, 2012.
- [32] A. Caprara V. Cacchiani and P. Toth. Scheduling extra freight trains on railway networks. *Transportation Research Part B*, 44(2):215–231, 2010.
- [33] M. Wohl. Notes on transient queuing behavior, capacity restraint functions, and their relationship to travel forecasting. *Papers in Regional Science*, 21(1):191 – 202, 1968.