Konrad-Zuse-Zentrum
für Informationstechnik Berlin

Takustraße 7
D-14195 Berlin-Dahlem
Germany

Ralf Borndörfer Berkan Erol Thomas Graffagnino
Thomas Schlechte Elmar Swarat

Optimizing the Simplon Railway Corridor

This work was funded by the Federal Ministry of Economics and Technology (BMWi), project *Trassenbörse*, grant 19M7015B.

ZIB-Report 10-24 (November 2010)

Optimizing the Simplon Railway Corridor

Ralf Borndörfer[*] Berkan Erol[*] Thomas Graffagnino[**]
Thomas Schlechte[*] Elmar Swarat[*]

01.11.2010

Abstract

This paper presents a case study of a *railway timetable optimization* for the very dense *Simplon corridor*, a major railway connection in the Alps between Switzerland and Italy. Starting from a detailed *microscopic network* as it is used in railway simulation, the data is transformed by an automatic procedure into a less detailed *macroscopic network*, that is sufficient for the purpose of capacity planning and amenable to state-of-the-art integer programming optimization methods. In this way, the macroscopic railway network is saturated with trains. Finally, the corresponding timetable is retransformed to the microscopic level in such a way that it can be operated without any conflicts among the slots. Using this micro-macro aggregation-disaggregation approach in combination with integer programming methods, it becomes for the first time possible to generate a profit maximal and conflict free timetable for the complete Simplon corridor over an entire day by a simultaneous optimization of all trains requests. This also allows to to undertake a sensitivity analysis of various problem parameters.

Keywords: railway track allocation, network aggregation, case study, Simplon corridor

1 Introduction

Timetabling and, intimately related, train *slot allocation* are two of the most challenging planning problems for every railway infrastructure provider. Due to the

[**]Schweizerische Bundesbahnen SBB AG, Mittelstrasse 43, 300 Bern 65, Switzerland, Infrastructure / Timetable and Network Design, Email thomas.graffagnino@sbb.ch

[*]Zuse-Institute Berlin (ZIB), Takustrasse 7, 14195 Berlin-Dahlem, Germany, Email borndoerfer, erol, schlechte, swarat@zib.de

ongoing deregulation of the transportation market in Europe, new railway undertakings are entering the market. This leads to an increase in slot requests and thus to a higher number of conflicts among them. The goal of timetabling and slot allocation is to resolve these problems as much as possible by producing a feasible, i.e., conflict free, timetable that achieves a maximal utilization of the railway infrastructure.

From a mathematical point of view, the *optimal slot allocation problem* can in principle be easily stated as a *multi-commodity flow problem* in an extremely large space-time network with certain additional constraints, see, e.g., Borndörfer & Schlechte [2007]. The problem is well investigated in the literature, but it is only a very recent development that problems of sizes that are, say, somewhat relevant in practice started to become tractable. The literature is therefore currently divided into two branches. One branch is devoted to the study of simplified macroscopic models, considering mainly corridor scenarios, see Cai & Goh [1994]; Brännlund et al. [1998]; Caprara et al. [2002]; Borndörfer et al. [2006]; Cacchiani [2007]; Borndörfer & Schlechte [2007]; Cacchiani et al. [2008]; Fischer et al. [2008]. The other branch investigates routing problems through individual stations on a more detailed level, see Zwaneveld et al. [1996]; Caimi et al. [2004]; Lusby et al. [2006]; Caprara et al. [2007]. The as far as we know only work that discusses the interplay between both approaches is the thesis of Caimi et al. [2009], who develops a top-down planning strategy that is based on empirical knowledge of experienced planners.

We demonstrate in this paper the application of a bottom-up planning approach that is based on an automatic simplification of complex *microscopic* railway infrastructure data. This is done at the example of the Simplon railway corridor. The term "microscopic" means that the starting data describes the infrastructure and the train driving dynamics on a very detailed level that is suitable to simulate the railway traffic accurately, i.e., with exact track and platform assignments and with a precise treatment of switches, signals, and vehicle dynamics. This data is condensed into much fewer constituents, namely, only those that are relevant for planning by means of a "macrotization" that makes the problem amenable to exact integer programming optimization. It is, of course, crucial to retain as many degrees of freedom as possible in the construction of the *macroscopic* network. The micro-macro network aggregation algorithm is implemented in a tool called NETCAST, see Borndörfer et al. [2010], the optimization algorithm in a tool called TS-OPT, see Borndörfer et al. [2009].

The transition between microscopic and macroscopic planning in our approach may seem complicated and difficult, but it is well justified. In fact, testing the operational feasibility of timetables and slot allocations is one of the prime motivations behind the development of railway *simulation* systems. In the last 20 years several sophisticated programs of this type were developed, see Wendler [1999]; Gröger [2002]; Hürlimann [2001]; Siefer & Radtke [2005]. Almost all railway companies use them to support their operations and their planning processes by experimentally evaluating the interactions between large numbers of trains in a timetable, the stability of a timetable with respect to delays, and the effects of changes in the infrastructure or the rolling stock pool. Industrial simulation tools are therefore widely accepted for their accurate reproduction of the operation of a railway system. If a timetable or slot allocation "works" in one of them, it can be considered to be operable. We take this point of view in this study, and use the synchronous simulation system `OpenTrack`, that was developed at the ETH Zurich, see Hürlimann [2001], as the basis of our approach; see also Hansen & Pachl [2008] for a wide comparison of synchronous and asynchronous railway simulation methods. Our *microscopic network* consists of exported `OpenTrack` data, and `OpenTrack` was also used to verify the feasibility of our timetables and slot allocations.

A timetable optimization at the microscopic level is completely out of reach, the resulting models become simply too large. However, it would also not be appropriate for many planning purposes. Microscopic data does not have to be considered on a strategic or on a tactical level. Information about train acceleration and braking, or the operation of signals can be compressed into running and minimal headway time information, segments of tracks can often be amalgamated, etc. In similar areas, e.g., in line planning by Borndörfer et al. [2007], or in periodic timetabling by Liebchen [2006], simplified models of the railway infrastructure and approximations of event times, mostly in minutes, are commonly used with success. We also use a *macroscopic model* of the railway system in a standardized format, for which documentation and test instances mapping a part of the German long distance network are freely available in the TTPLIB library, see Erol et al. [2008]. This aggregation brings the network size to a manageable level that allows for a simultaneous optimization of many train slots; it must of course be done in a *conservative* way, such that any timetable that works on the macroscopic level can be translated back to the microscopic setting. This can be done by working with appropriate buffers and is described in detail in a companion paper, see Borndörfer et al. [2010].

The contribution of this paper is to demonstrate the viability of a bottom-up micro-macro transformation approach on a real-world example, the Simplon corridor between Switzerland and Italy. In addition, we study the effects of variations in important parameters such as time discretization. The paper is structured as follows. The problem setting is described in Section 2. Section 3 provides a capacity

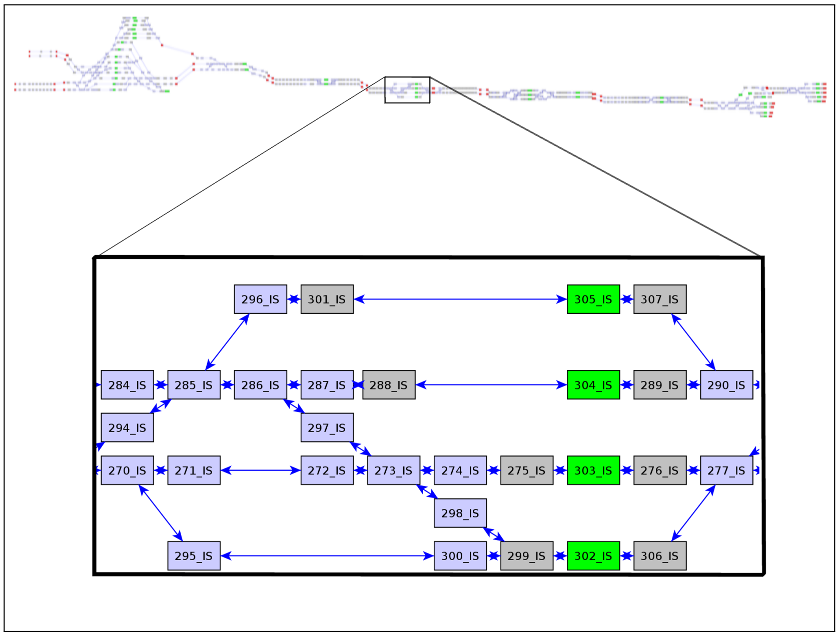


Figure 1: Microscopic representation of the Simplon corridor with zoom on station Iselle as given by the railway simulator `OpenTrack`.

analysis of the Simplon tunnel using state-of-the-art optimization techniques.

2 The Simplon Corridor

There are only two north-south railway connections through the Alps in Switzerland, namely, the Gotthard corridor and the Lötschberg-Simplon corridor. The Simplon connects Switzerland and Italy and is therefore of strategic importance for the international railway freight traffic. It has a length of approximately 45 km and 12 stations. This may sound like a rather small network at first glance, but the routing possibilities at the terminals Brig and Domodossola, the routing possibilities in the intermediate stations Iselle and Varzo, and a rather unusual slalom routing for certain types of cargo trains lead to very complex planning situations. An `OpenTrack` network data export for the part from Brig (BR) in Switzerland to Domodossola (DO) in Italy was provided by the SBB Schweizerische Bundesbahnen. The microscopic network consists of 1154 nodes and 1831 arcs including 223 signals, see Figure 1.

This data was macrotized in two steps. The first step is resort to standardized train driving dynamics that lead to the definition of a handful of *train types*; these are used to compute standardized driving and headway times. This allows to amalgamate larger parts of the microscopic infrastructure network to a *macroscopic network* in the second step. The following subsections describe this process for the Simplon application.

2.1 Train Types

The decision which train types to consider is a crucial point, because a more detailed consideration of driving dynamics allows the construction of tighter schedules. For a capacity analysis, however, a modeling strategy is appropriate that captures the main characteristics, but abstracts from minor special characteristics of individual trains. We use six different train types, two for passenger trains and four for freight trains.

The different, but invariable stopping patterns of *regional trains* (R) and *intercity trains* (EC) and their very different driving dynamics (due to the different engines used) result in considerable differences in running and headway times for such trains. They are therefore considered as two train types. We do, however, ignore different train compositions, i.e., in length and in the number of wagons. Hence, R and EC are the two types of *passenger trains* that we consider.

Freight trains come in four different types. *GV Auto* are special train services that transport passengers and their automobiles from Brig (BR) to the next station after the Simplon tunnel, which is Iselle (IS). There these trains cross all other tracks to reach an isolated ramp. Because of these unique routing requirements at Iselle, we consider them as belonging to an individual freight train type on their own. *GV RoLa* and *GV SIM* are train types that transport freight vehicles (*GV RoLa*) and containers (*GV SIM*). They have a larger height and width than standard freight trains, and they can use only one of the tracks in the tunnel between Iselle and Preglia. This results in a so-called “slalom route” that these trains have to take from Brig. In Iselle they have to change to the right track[1] until Preglia, i.e., it is possible to change again to the standard side in the intermediate station Varzo to let other trains pass. Furthermore, the running times of these trains types, especially for the direction from Brig to Domodossola, differ significantly, namely, a *GV RoLa* needs about 7 minutes more than a *GV SIM*. They also use different routes in the area of Domodossola. Thus separate train types *GV RoLa* and *GV SIM* are

[1] In Switzerland trains are usually running on the left side.

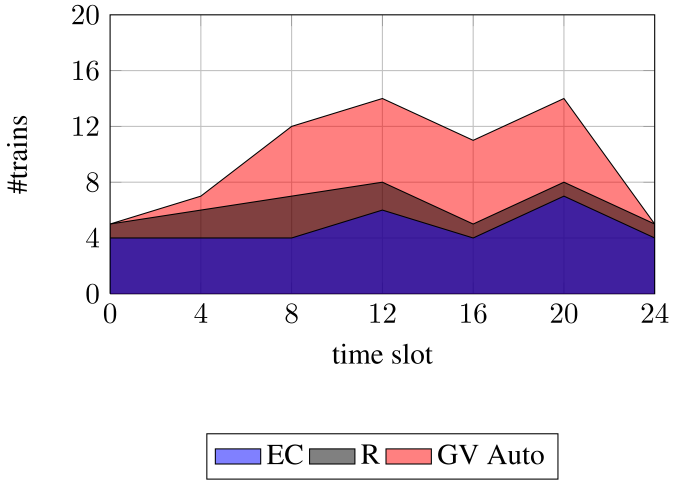


Figure 2: Given distribution of passenger or fixed traffic in the Simplon corridor for both directions.

introduced. Finally, *GV MTO* are standard freight trains which use the standard tracks in the Iselle-Preglia tunnel.

SBB was interested in running additional freight trains through the Simplon such that we concentrated on freight traffic. We assume, in particular, that the passenger trains are given and cannot be changed. Hence the slots for passenger trains R and EC from Brig to Domodossola and vice versa are fixed. In addition, the *GV Auto* trains, which are not operated all day, are also fixed. All these trains must, however, be considered with respect to their influence on the remaining traffic, i.e., with respect to their headways and with respect to station capacities. Figure 2 shows the distribution of passenger and GV Auto trains during the day.

2.2 Network Aggregation

The train types introduced in Section 2.1 can run on 28 different routes through $G = (V, E)$. The routes differ in their stopping pattern and in various ways to pass through Varzo. These routes are the basis of the aggregation of the microscopic network. They partition the network into segments, on which driving and headway times can be computed individually. In other words, if a train route runs on a track segment and no other routes cross, one can compute the parameters that are relevant for a slot allocation on this segment beforehand, and compress the segment.

Clearly, the routes meet at the stations, such that the macroscopic network must necessarily contain a node for each of the twelve stations. Some more macroscopic *pseudo nodes* are needed to model all train route interactions correctly, i.e., divergences, convergences, and crossings. Applying the NETCAST micro-macro transformation algorithm described by Borndörfer et al. [2010] produces a macroscopic network $N = (S, J)$ with 55 nodes and 87 tracks. 32 of these nodes are pseudo stations. Most of them are located directly in the front area of stations. The other 23 nodes are possible start, end, or waiting nodes along the corridor.

This automatically constructed network was further aggregated in a second step by applying some reductions that are not yet implemented in NETCAST. We kept only those pseudo stations that handle crossing conflicts, namely, for *GV Auto* on the route from Brig to Iselle and those for a detailed modeling of the station Varzo. The reason for this detailed treatment of Varzo is that the routing through this station is crucial for the capacity of the whole corridor. In Varzo the over-width freight trains can pass each other, such that a locking of the entire area between Iselle and Preglia can be avoided for *GV SIM* and *GV RoLa* trains from the other direction, when one of them runs through the tunnel. All other potential pseudo nodes were aggregated to the closest station node in a conservative manner, i.e., the headway times for the incident tracks had to be slightly overestimated. In addition, some nodes that represent different platforms at the same station were aggregated. After these modifications the network consists of 18 stations and 40 tracks. For comparison, we also consider a "traditional" macroscopic network that is solely based on station nodes; clearly, a conservative model based on such an aggregation will employ oversized buffers and therefore waste capacity. Let us list the macroscopic networks, that we constructed using NETCAST on the basis of microscopic `OpenTrack` data:

- ▷ network with station area aggregation (18 stations and 40 tracks), *simplon_big*,
- ▷ network with full station aggregation (12 stations and 28 tracks), *simplon_small*.

After some experiments with these networks, the expertise of SBB about the operational conditions in the Simplon corridor led to the introduction of an additional technical blocking time for combinations of *GV RoLa* trains with any other trains in the front area of Domodossola. The headway times of cargo trains were set to a fixed value of some minutes instead of the simulation values in order to guarantee certain departure and arrival distances in the marshaling yard of Brig. We further improved the macroscopic model by adding buffer times for standard headways and headways for the opposite direction. In this way, two more macroscopic networks were generated:

	Brig-Domodossola		Domodossola-Brig	
Δ (sec.)	running	headway	running	headway
1	1778	272	1794	251
6	297	46	299	42
12	158	23	149	21
30	60	10	60	9
60	30	5	30	5
300	6	1	6	1

Table 1: Running and headway times for EC trains with respect to different time discretizations Δ.

- ▷ network with station area aggregation (18 stations and 40 tracks) and technical times, *simplon_tech*,
- ▷ network with station area aggregation (18 stations and 40 tracks) and technical and buffer times *simplon_buf*.

2.3 Time Discretization

An important point that we have not yet discussed is the impact of different time discretizations on a macroscopic model. Clearly, a coarser treatment of times simplifies the scenario, but must be compensated by large buffer times, which reduces the corridor capacity. We illustrate this trade-off now by means of a simple example.

Disregarding realistic traffic assumptions, the "most efficient" utilization of a railway infrastructure is to use the fastest train type as much as possible. For the Simplon corridor, this is an EC train. The EC driving times for both directions are listed in Table 1. Here, we denote by d the rounded running time with respect to the time discretization Δ and by h the technically minimal rounded headway time, respectively.

It can be seen that assuming a coarse discretization Δ in units of 5 minutes, only 12 trains can be run in each direction per hour. Only when Δ is smaller than 12 seconds, the maximum capacity of 13 or 14 trains per direction and hour becomes (theoretically) available. The choice of the time discretization has therefore a major influence on the capacity of a macroscopic railway network, even for very simple examples with two stations, two tracks, and one train type.

		type		direction		freight trains		
name	train requests	passenger	freight	BR-DO	DO-BR	*GV RoLa*	*GV SIM*	*GV MTO*
4h-tp-as-d	41	15	26	23	18	4	9	13
4h-tp-as-n	36	8	28	20	16	7	10	11
4h-tp-s-d	42	15	27	23	19	4	8	15
4h-f20-s	38	14	24	22	16	6	12	6
4h-f15-s	46	14	32	26	20	8	16	8
4h-f12-s	54	14	40	30	24	10	20	10
4h-f10-s	62	14	48	34	28	12	24	12
4h-f7.5-s	78	14	64	42	36	16	32	16
24h-tp-as	390	63	327	203	187	69	108	150
24h-tp-s-n	219	63	156	110	109	48	54	54
24h-tp-s	297	63	234	149	148	60	78	96
24h-f24-s	183	63	120	92	91	30	60	30
24h-f20-s	207	63	144	104	103	36	72	36
24h-f15-s	255	63	192	128	127	48	96	48
24h-f12-s	303	63	240	152	151	60	120	60
24h-f10-s	351	63	288	176	175	72	144	72

Table 2: Statistics of demand scenarios for the Simplon case study

2.4 Demand

The capacity of the Simplon corridor is estimated by saturating it with freight trains, that are selected from fictional request sets. To this purpose, we have constructed 16 train request sets listed in Table 2. The first eight request sets cover a four hour time horizon (prefix "4h" in the request set name) either from 8am to 12am (suffix "d" for day) or from 0am to 4am (suffix "n" for night). The other request sets are used to calculate a timetable for an entire day (24h).[2]

Three of the 4h request sets are called "testplan" (tp), which means that they are used to evaluate the correctness of the micro-macro transformation on the basis of a microscopically feasible timetable that has been generated manually by the authors. The same applies to the three "testplan" request sets that cover the whole day. Some of the test request sets, e.g., 24h-tp-as, have the disadvantage that the requests are not symmetrically distributed with respect to both directions. We therefore distinguish between asymmetric (as) and symmetric (s) request sets, which do not have this drawback.

We also remark that almost all "tp" request sets do not match the train type distribution that is desired by SBB. Namely, traffic demand in practice takes the form that every second request is a *GV SIM*, while the others are *GV RoLa* and *GV MTO* in equal parts. To approximate this characteristic, we generated some more requests using a uniform distribution according to the desired train demand pattern. The resulting request sets are named with the infix "fx", where x denotes the period time of the freight trains. We remark that we are aware of the fact that in practice

[2] The "n" in the second 24h request is a reminder that freight trains drive more frequently at night.

traffic demand is not uniformly distributed, however, for want of better data and for the purpose of demonstrating the principal viability of our model in an analysis of a theoretical capacity of the corridor, we deem this data good enough.

3 Capacity Analysis

We provide in this section a capacity analysis of the Simplon corridor using our micro-macro aggregation approach. The goal of this study is to saturate the residual capacity of the corridor by running a maximum number of fictitious freight trains (*GV MTO*, *GV SIM*, *GV RoLa*) between the passenger trains (remember the passenger trains are given as fixed).

We remark that there are a lot of side-constraints for such additional trains that we do not consider. Requirements such as desired arrival or departure time windows at certain stations, dwell time requirements, the balance of train traffic in opposite directions, and other constraints are ignored, partly because of lack of data, partly because there is no point for such constraints in an analysis of a theoretical capacity maximum. These considerations are also the reason for using the following simple objective function:

- ▷ a basis value for each scheduled train depending on type and direction,
- ▷ a penalty for deviations from optimal arrival and departure times,
- ▷ and very small penalties for travel time increases or avoidable stops.

We constructed the macroscopic scenarios associated with all request sets and with all four macroscopic networks, namely *simplon_small*, *simplon_big*, *simplon_tech*, and *simplon_buf*. Furthermore, we varied the time discretization of the model using step sizes of 6, 10, 30, and 60 seconds. The resulting macroscopic track allocation problems were solved using the integer programming based track allocation optimizer TS-OPT, see Borndörfer & Schlechte [2007], the solutions were disaggregated using NETCAST and verified by `OpenTrack`. For each run of TS-OPT, a time limit of one day (86400 seconds) was used. All computations were done on machines with a 3 GHz Intel Quad Core Processor and 8GB of RAM running Suse-Linux 11.2.

Table 3 lists exemplary solution statistics for all request scenarios and network *simplon_big* using a discretization of 10s. The tables gives:

- ▷ number of trains,
- ▷ number of columns of the integer programming formulation,
- ▷ number of rows of the integer programming formulation,

instance	#trains	#cols	#rows	v(lp)	bbound	v(ip)	gap	t(lp)	t(ip)
4h-tp-as-d	35	70476	30432	149.35	147.27	147.27	-	0.00	18.68
4h-tp-as-n	27	35859	17136	151.21	146.39	146.39	-	0.03	14.60
4h-tp-s-d	36	106201	45873	90.77	70.57	70.57	-	23.28	2054.04
4h-f20-s	30	173929	69531	152.52	145.97	145.97	-	54.23	2397.83
4h-f15-s	34	110920	46870	151.76	136.90	136.90	-	18.82	1440.07
4h-f12-s	36	211745	84107	189.57	186.36	186.36	-	107.78	12508.98
4h-f10-s	37	235430	93501	206.09	200.33	200.33	-	153.58	12124.92
4h-f7.5-s	37	135746	56968	79.26	72.15	72.15	-	37.97	11856.11
24h-tp-as	203	462769	196238	1035.94	984.77	984.77	-	102.73	63588.77
24h-tp-s-n	154	284038	117208	794.62	760.63	760.63	-	40.45	1609.42
24h-tp-s	174	403017	167548	888.97	843.30	843.30	-	76.02	27391.87
24h-f24-s	143	444199	178162	722.29	697.12	697.12	-	92.60	4454.76
24h-f20-s	156	471759	195167	791.31	752.49	752.49	-	93.70	3779.25
24h-f15-s	174	660642	250673	919.22	885.43	861.84	2.74	235.06	86400.40
24h-f12-s	179	662236	259676	985.46	958.76	958.76	-	213.54	79497.37
24h-f10-s	193	791285	312943	1090.47	1069.70	1041.08	2.75	426.75	86400.71

Table 3: Solutions for network *simplon_big* using a time discretization of 10s and a time limit of 86400 seconds.

- ▷ optimal value of the linear relaxation (v(lp)),
- ▷ (best) proven upper bound (bbound),
- ▷ (best) objective function value of integral solution (v(ip)),
- ▷ optimality gap in percent,
- ▷ time needed to solve the linear relaxation,
- ▷ the total running time of TS-OPT.

A first important result is that TS-OPT is indeed able to compute a feasible, i.e., conflict free, slot allocation for all instances within one day. Figure 3 shows an example of a resulting train diagram with a valid block occupation for request set 24h-tp-as, network *simplon_buf*, and a discretization of 30s. The tractability of these instances is due to the network aggregation algorithm of NETCAST, see Borndörfer et al. [2010], which produces reasonably sized macroscopic networks that give rise to reasonably sized track allocation problems. There is no instance where TS-OPT needs more than 600 MB of main memory, and TS-OPT can therefore compute feasible solutions for almost all problems. This give evidence that our micro-macro aggregation approach works.

Not every instance could be solved to proven optimality for each network and time setting. But the 4h-requests never took more than three and a half hours to be solved to optimality, and even for the really complex uniformly distributed daily scenarios feasible solutions with small optimality gaps could be computed. Moreover, the instance with the maximum number of train requests (24h-tp-as with 390 train requests) could be solved to optimality for each network and all time discretizations of 30 seconds and more. Table 3 shows that such an instance produces a timetable with 203 trains, which means that 140 freight train slots out of the requested potential 327 train slots are routed in the optimal schedule. This establishes a theoretical

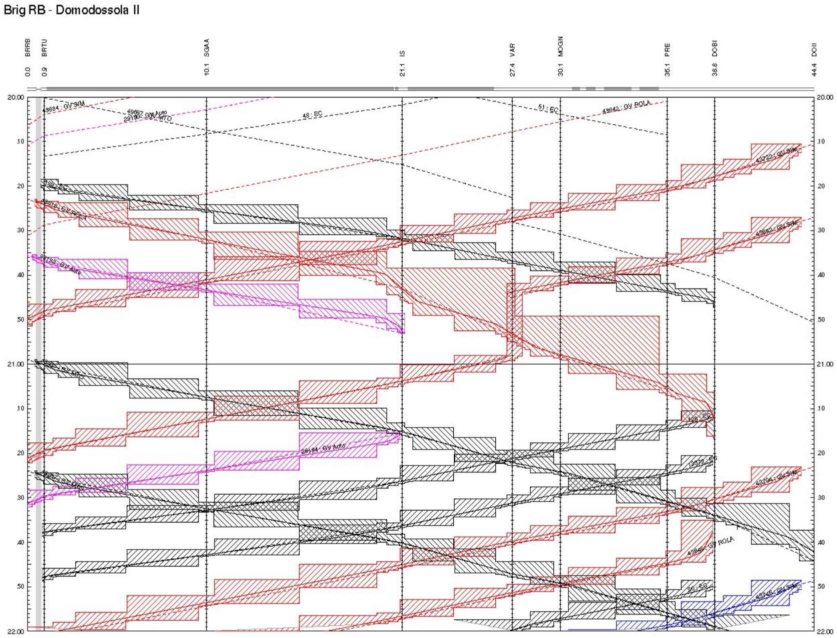


Figure 3: `OpenTrack` traffic diagram with block occupations for request set 24h-tp-as.

instance	24h-tp-as					24h-f15-s				
discret. in sec.	6	10	30	30	60	6	10	30	30	60
dep_steps (sec.)	30	50	150	30	60	30	50	150	30	60
wait_steps (sec.)	60	100	300	60	60	60	100	300	60	60
#cols	504314	318303	114934	370150	178974	649494	375694	115293	392146	172462
#rows	222096	142723	53311	170525	81961	234529	146044	49458	163388	74200
t(lp) (sec.)	135.67	48.88	17.77	54.13	151.67	190.36	64.59	2.83	47.44	103.50
t(ip) (sec.)	72774.55	12409.19	110.34	81683.02	2411.20	2923.76	2639.62	34.83	8265.71	1043.48
size of IP (MB)	50	30	10	36	18	64	36	10	38	16
#trains	196	187	166	188	180	176	163	143	155	145

Table 4: Solution data of two instances respect to the chosen time discretization for *simplon_small*

capacity of the Simplon corridor of more than 200 trains per day. Adding technical and buffer times in network *simplon_buf*, it is still possible to schedule 170 trains. This number is almost identical to the saturation in the timetable that is currently in operation and can be taken as an indication of both the accuracy of the model as well as the quality of the current timetable. We can also observe that not every request set produces a saturated timetable that runs between 160 and 200 trains per day. This highlights the fact that the demand, i.e., the number of requested trains of different types and the degrees of freedom in routing them have a crucial effect on the capacity of a corridor.

We also analyzed the effects of different time discretizations. Table 4 gives an overview on the sizes of the resulting track allocation problems for two test instances. We distinguish two different discretization parameters, namely, we denote by dep_steps the step size for train departure events and by wait_steps the step size for train dwell activities, respectively. As expected, problem sizes normally[3] decrease with coarser time discretizations, and the same holds for the running times. Anyway, TS-OPT can solve even instances with more than 500.000 variables.

An exception to the rule – coarser time discretization implies a decrease in problem size – can be observed by comparing the first 30s and the 60s instances. This irregularity originates from a different parameter setting with respect to possible departure and waiting times, see Table 4. In the first 30s discretization scenario a train can only depart at times that are multiples of 150 seconds, see definition of dep_steps, and the waiting times must be a multiple of five minutes, see definition of wait_steps. That is a rather rough model with limited degrees of freedom. We therefore changed the parameters for the 60s runs, such that the time steps are narrower and more similar to the 6s case. We also did 30s runs with departure and waiting times similar to the 6s cases, such that the influence of those two parameters could be analyzed. It turns out that there is not only a connection between time discretization and the number of scheduled trains, but there is also an often even stronger connection between departure and waiting time steps and the number of scheduled trains. We therefore also must pay attention to these parameters. We finally remark that the combinatorial complexity and/or the computational tractability of a particular track allocation instance can not be reliably predicted or estimated by looking at simple scenario statistics.

Another important point is the influence of network aggregation on the number of

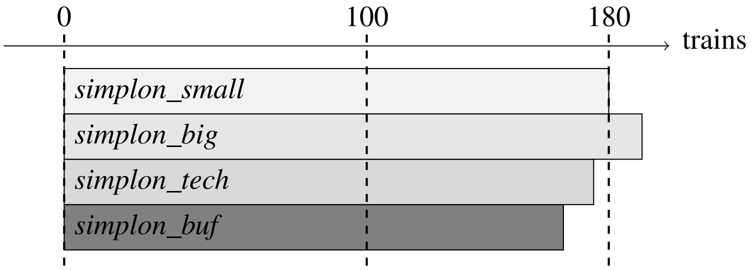
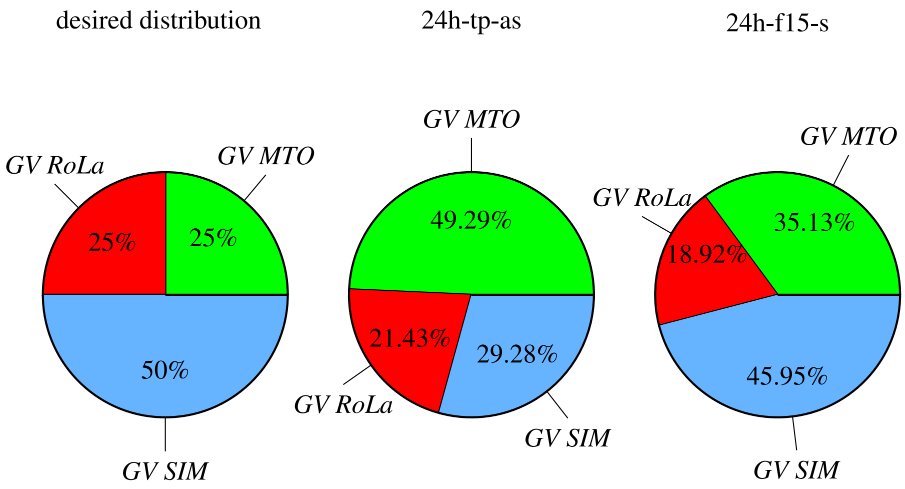


Figure 4: Comparing the number of scheduled trains for different networks (*simplon_*) for request set 24h-tp-as using a 60s time discretization.

[3] There is no general relation between problem size and solution time as one can see by a comparison of the 6s-discretization runs.



Freight Trains scheduled vs. requested	24h-tp-as 140	 327	24h-f15-s 111	 192
GV RoLa	30	69	21	48
GV SIM	41	108	51	96
GV MTO	69	150	39	48

Figure 5: Distribution of scheduled freight trains for the request sets 24h-tp-as and 24h-f15-s using network *simplon_big* and a time discretization of 10 seconds.

scheduled trains. As shown in Figure 4, a more detailed network model leads to a major increase in the number of scheduled trains. But by introducing specific headway times, we again loose about 8% of the trains and an additional 6% by also considering buffer times.

Up to now we only considered the total number of scheduled trains as a measure for the corridor capacity. But it is also important to keep the *structure* of the computed timetable in mind. Figure 5 shows the train type distribution of the three freight train types for two request sets. This little example is representative for the general observation that the train type distribution associated with uniformly distributed requests is much closer to the desired distribution, see Figure 5, than that of the requests based on the testplan timetable. The latter timetables feature a higher fraction of *GV MTO* requests than desired; in fact, these trains do not run on a slalom route in the corridor and are therefore easier to schedule. The higher percentage of *GV SIM* and *GV RoLa* trains in the uniformly distributed request sets often leads to bigger problems than that resulting from the testplan request sets, see Table 4.

Another observation is that the majority of timetables schedules more trains from Domodossola to Brig than vice versa. This is not surprising as the models do not contain any symmetry constraints. We did, however, try to achieve some balance by manipulating the objective function. Introduce such global constraints could be an interesting aspect of future work.

Acknowledgments

We thank Martin Grötschel, Gottfried Ilgmann and Klemens Polatschek for their important support in organising und realizing this project, and in particular the Simplon case-study. Finally, we thank Daniel Hürlimann for his excellent support for the simulation tool `OpenTrack`.

References

Borndörfer, Erol, Graffagnino, Schlechte & Swarat (2010). Aggregation Methods for Railway Networks. ZIB Report 10-23, ZIB, Takustr. 7, 14195 Berlin.

Borndörfer, Erol & Schlechte (2009). Optimization of macroscopic train schedules via TS-OPT. In Hansen, Wendler, Weidmann, Lüthi, Rodriguez, Ricci & Kroon (Eds.), *Proceedings of the 3rd International Seminar on Railway Operations Modelling and Analysis - Engineering and Optimisation Approaches*, Zürich, Switzerland.

Borndörfer, Grötschel, Lukac, Mitusch, Schlechte, Schultz & Tanner (2006). An Auctioning Approach to Railway Slot Allocation. *Competition and Regulation in Network Industries 1(2), 163–196*.

Borndörfer, Grötschel & Pfetsch (2007). A Column-Generation Approach to Line Planning in Public Transport. *Transportation Science 41(1), 123–132*.

Borndörfer & Schlechte (2007). Models for Railway Track Allocation. In Liebchen, Ahuja & Mesa (Eds.), *ATMOS 2007 - 7th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems*, Dagstuhl, Germany. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany.

Brännlund, Lindberg, Nou & Nilsson (1998). Railway Timetabling using Langangian Relaxation. *Transportation Science 32(4), 358–369*.

Cacchiani (2007). *Models and Algorithms for Combinatorial Optimization Problems arising in Railway Applications*. PhD thesis, DEIS, Bologna.

Cacchiani, Caprara & Toth (2008). A column generation approach to train timetabling on a corridor. *4OR 6(2), 125–142*.

Cai & Goh (1994). A fast heuristic for the train scheduling problem. *Comput. Oper. Res. 21(5), 499–510*.

Caimi, Burkolter & Herrmann (2004). Finding Delay-Tolerant Train Routings through Stations. In Fleuren, den Hertog & Kort (Eds.), *OR*, pp. 136–143.

Caimi, Burkolter, Herrmann, Chudak & Laumanns (2009). Design of a Railway Scheduling Model for Dense Services. *Networks and Spatial Economics 9(1), 25–46*.

Caprara, Fischetti & Toth (2002). Modeling and Solving the Train Timetabling Problem. *Operations Research 50(5), 851–861*.

Caprara, Galli & Toth (2007). Solution of the Train Platforming Problem. In Liebchen, Ahuja & Mesa (Eds.), *ATMOS*, volume 07001 of *Dagstuhl Seminar Proceedings*. Internationales Begegnungs- und Forschungszentrum fuer Informatik (IBFI), Schloss Dagstuhl, Germany.

Erol, Klemenz, Schlechte, Schultz & Tanner (2008). TTPLIB 2008 - A Library for Train Timetabling Problems. In J. Allan, E. Arias, C.A. Brebbia, C. Goodman, A.F. Rumsey, G. Sciutto (Ed.), *Computers in Railways XI*. WIT Press.

Fischer, Helmberg, Janßen & Krostitz (2008). Towards Solving Very Large Scale Train Timetabling Problems by Lagrangian Relaxation. In Fischetti & Widmayer (Eds.), *ATMOS 2008 - 8th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems*, Dagstuhl, Germany. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany.

Gröger (2002). *Simulation der Fahrplanerstellung auf der Basis eines hierarchischen Trassenmanagements und Nachweis der Stabiliät der Betriebsabwicklung*. PhD thesis, Veröffentlichungen d. Verkehrswiss. Inst. der Rheinisch-Westfälischen Techn. Hochsch. Aachen.

Hansen & Pachl (2008). *Railway, Timetable & Traffic*. Eurailpress, Hamburg.

Hürlimann (2001). *Object oriented modeling of infrastructure elements and business processes in railways*. PhD thesis, ETH Zürich.

Liebchen (2006). *Periodic timetable optimization in public transport*. Springer.

Lusby, Larsen, Ryan & Ehrgott (2006). Routing Trains Through Railway Junctions: A New Set Packing Approach. Technical report, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby.

Siefer & Radtke (2005). Railway-Simulation Key for Better Operation and Optimal Use of Infrastructure. In *Proceedings of the 1st International Seminar on Railway Operations Modelling and Analysis*.

Wendler (1999). *Analytische Berechnung der planmässigen Wartezeiten bei asynchroner Fahrplankonstruktion*. Verkehrswiss. Inst. der Rheinisch-Westfälischen Techn. Hochsch. Aachen.

Zwaneveld, Kroon, Romeijn, Salomon, Dauzere-Peres, Van Hoesel & Ambergen (1996). Routing Trains Through Railway Stations: Model Formulation and Algorithms. *Transportation Science 30(3), 181–194*.