An optimization model for integrated planning of railway traffic and network maintenance

Tomas Lidén and Martin Joborn

Journal Article

N.B.: When citing this work, cite the original article.

Original Publication:

http://dx.doi.org/10.1016/j.trc.2016.11.016

Copyright: Elsevier
http://www.elsevier.com/

Postprint available at: Linköping University Electronic Press
http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-132961
An optimization model for integrated planning of railway traffic and network maintenance

Tomas Lidén*, Martin Joborn

Linköping University, Department of Science and Technology, Norrköping SE-601 74, Sweden

Abstract

Railway transportation systems are important for society and have many challenging and important planning problems. Train services as well as maintenance of a railway network need to be scheduled efficiently, but have mostly been treated as two separate planning problems. Since these activities are mutually exclusive they must be coordinated and should ideally be planned together. In this paper we present a mixed integer programming model for solving an integrated railway traffic and network maintenance problem. The aim is to find a long term tactical plan that optimally schedules train free windows sufficient for a given volume of regular maintenance together with the wanted train traffic. A spatial and temporal aggregation is used for controlling the available network capacity. The properties of the proposed model are analysed and computational experiments on various synthetic problem instances are reported. Model extensions and possible modifications are discussed as well as future research directions.

Keywords: Railway scheduling; Maintenance planning; Optimization

1. Introduction

Efficiency in public and freight transportation systems is of great importance for a society. Railways can offer high capacity and relatively low environmental impact, but require that several technical systems like the track, power distribution, safety, telecommunications and trains are tuned and operate well. Disturbances in any of these systems will degrade the service level. Moreover, failures or degradation in one may cause damages in the others, which is particularly true for track, trains and power distribution. Thus, maintenance is essential for upholding reliability, transportation throughput and the benefits of the infrastructure system investments.

Railway infrastructure maintenance consumes very large budgets, is complicated to organise and has numerous challenging planning problems (Lidén, 2015). Specifically, the coordination of maintenance tasks and train traffic is of great importance, since these activities are mutually exclusive. This planning conflict becomes crucial on lines with high traffic density and/or around the clock operation - especially when both traffic demand and maintenance needs are increasing, which is the case in many European countries. Hence, large benefits can be realised if planning, scheduling and effectuation can be improved. As an example of monetary volumes, the European countries are reported to allocate 15 - 25 billion EUR annually on maintenance and renewals for their railway systems consisting of about 300,000 km of track, giving an average spending of 70,000 EUR per km track and year (EIM-EFRTC-CER Working Group, 2012). The spending in Sweden is in the same range: During the year 2014 a budget of 910 million EUR was used for a track network of 14,700 km. At the same time the major passenger traffic operator (SJ AB) and freight traffic operator (Green Cargo AB) had a joint turnover of 1,465 million EUR. Thus the maintenance spending amounts to about 60% of the main traffic operator costs.

*Corresponding author. Tel: +46 70 755 0020
Email addresses: tomas.liden@liu.se (Tomas Lidén), martin.joborn@liu.se (Martin Joborn)

Historically, research about railway scheduling and planning has focused mainly on train operations and timetabling. In such models, maintenance activities can be handled as fictitious slow-moving trains. Kraay and Harker (1995) and Törnquist Krasemann (2015) (both of which treat real-time operational dispatching) mention this possibility, without discussing it further. This is a valid approach if the work tasks to be scheduled have a fixed duration and do not impose other restrictions than the blocking of one track section. It is however not sufficient for situations where (a) a work activity can be interrupted (for letting real trains through), (b) the work closes off several tracks or line segments at the same time, or (c) the work inflicts speed restrictions or other operational restrictions on neighbouring tracks. In such cases maintenance should be treated as separate planning objects, since the interaction between traffic and maintenance can be modelled more correctly. In addition, the following properties distinguish maintenance from traffic: (i) work tasks can be co-located, (ii) the costs have a non-linear dependency on the given shift and possession time and (iii) work force restrictions are differently handled.

Maintenance can also be considered as orthogonal to train traffic, since the former is mainly tied to and organised in bounded geographical areas, while train traffic concerns transportation needs between distant regions. Consequentially, the responsibility for traffic and infrastructure is usually split into separate organisations or companies. Thus the planning tasks are also divided - each party treating the other type of activities as an unknown or given input - and solving the track access (capacity) conflicts with some coordination or resolution procedure. Not surprising, there is often a lack of understanding and even mistrust between representatives for traffic operators, infrastructure management and contractors. This situation will almost certainly result in solutions that are less good or even inefficient for one or more of the parties.

In summary, the volumes, importance and specific properties of railway infrastructure maintenance give strong motivations for investigating how maintenance activities and traffic operation could be planned together efficiently. Further, there has been very little research about how to model or solve such integrated planning problems.

The research question addressed in this paper is how to coordinate maintenance activities and timetabled traffic operation on a common railway infrastructure. The hypothesis is that this coordination problem is possible to model and solve as an optimization problem. We will investigate how well a mixed integer linear programming approach is able to capture and solve the studied planning problem. The aim is to achieve a long term tactical plan for when and how to perform traffic and maintenance, by scheduling train paths as well as train free time windows where maintenance work can be carried out.

The contributions of this work are: (1) the formulation of a mixed integer optimization model which jointly schedules train services and windows for infrastructure maintenance, (2) experiments demonstrating that weekly network problems and 1-2 day long line problems can be solved to near optimality within one hour of computation for synthetic test instances. The model uses an aggregated approach both spatially and temporally and the problem size is shown to grow linearly with the planning horizon if the train scheduling windows are limited.

The purpose of the optimization problem is to find a pattern of maintenance windows that allows for a wanted train traffic to be run and that minimizes the total cost for train operations and maintenance. The train operating cost is measured by total running time, deviation from preferred departure, route cost and cancellations, while the maintenance cost consists of direct work time and indirect setup/overhead time. Routing and scheduling of trains will respect given minimum travel and dwelling durations as well as the line capacity limitations imposed by the maintenance scheduling. Sufficient amount of maintenance windows will be scheduled to fulfill a given work volume, where the number of windows and their temporal size will respect a chosen window option for each network link. The main aspects, which are not considered in the presented model are: (i) train and maintenance resource constraints, (ii) regularity requirements and (iii) the detailed meet/pass planning needed for a conflict-free timetable.

The remainder of the paper is organised as follows: The problem setting together with basic assumptions and delimitations are presented in Section 2 followed by a literature review in Section 3. We then introduce and describe the optimization model in Section 4 while the complete and detailed mathematical notation is given in Appendix A. Computational experiments on a set of generated test instances are presented in Section 5 after which the model is discussed along with possible modifications, extensions and alternate approaches.
in Section 6. Conclusions and possible continuations are presented in Section 7. Finally, Appendix B lists the number of variables and constraints needed for the model and in Appendix C all the detailed computational results are given.

2. Problem setting

As a background we cite a previous paper (Liden and Joborn, 2016):

All non-train activities that require secure access to the railway infrastructure must obtain a (work) possession (RailNetEurope, 2013). Up to now, the planning regime adopted in Sweden has been to let the maintenance contractors apply for these work slots, which usually is done as late as possible. If no room is reserved for maintenance in the timetable it can be difficult to find suitable possessions, which forces the contractors to perform their work on odd times and/or divide the tasks into small chunks which leads to inefficiency and cost increases. If the work cannot be split into smaller tasks, then the timetable must be altered and train operations rescheduled. 

To increase the possibility for suitable work possessions, a new planning regime is now being introduced in Sweden, called maintenance windows, where the infrastructure manager will propose regular, 2-6 h train free slots before the timetable is constructed. Thus the maintenance windows are given as input to the yearly timetable process. In addition, the maintenance windows will be dimensioned and constructed before the procurement of maintenance contracts and will remain more or less unchanged during the contract period, giving stable planning and quotation conditions for the contractors. The goal is to perform almost all planned maintenance on work possessions within the stipulated maintenance windows. 

Hence, the basic idea is to go from a situation with many, small and fragmented work possessions squeezed into an already published timetable (which causes changes and disturbances for the train operations), to a situation with few, large and regular maintenance windows preplanned before the timetable is constructed and the maintenance contracts procured. The overall aim is to increase efficiency, reduce cost as well as planning burden and also to improve robustness and punctuality.

Note the difference between maintenance windows (predetermined train free slots in the timetable) and possessions (the actual reservations for specific work tasks which might be smaller geographically and/or in time). As shown in the above cited paper, the size of a maintenance window determines the total maintenance cost, since the amount of overhead time (preparation, setup and termination) will increase with shorter windows and decrease with larger ones. The result is that the total maintenance cost is inversely proportional to the temporal window size. Due to this non-linear dependency we will consider a set of different window options, say 1h, 2h or 5h slots, for which the necessary number of maintenance occasions and total overhead cost can be calculated (ibid: Appendix A). The optimization model will then decide the most efficient combination of window options and train scheduling.

Although our work is driven by the notion of maintenance windows, the resulting optimization model does not require or impose the planning regime described above. Rather it can be used whenever a known or prognosticated maintenance work volume shall be scheduled together with a given traffic demand - specified as a set of train services. Even though this is a long-term tactical planning problem, the modelling approach might also be applied in mid- or short-term (tactical/operational) scheduling situations.

We have chosen to let the railway capacity be controlled on the level of train counts per time period. This enables an aggregated view of the railway network, where it is not necessary to explicitly represent every meet/pass loop and intermediate stations. With such an aggregated representation, the detailed conflict resolution (meet/pass) planning is not handled and there will be some uncertainty in the exact running times for each train. Thus the user will either have to (a) account for such uncertainties (when aggregating the network) in the input data, or (b) disaggregate the network to a suitable level. In any case, a subsequent timetabling procedure is needed after solving the coordination problem we discuss here. The timetabling
problem might benefit substantially from using the obtained coordination solution as guidance or starting point.

Finally, it should be noted that the focus of this problem is on long distance and modifiable train traffic together with regular maintenance that should be placed at off-peak hours. Short distance and fixed traffic can be handled as a base load which reduces the available capacity, while maintenance during peak hours can be discouraged by high penalties for such time periods. The synthetic test instances have been dimensioned according to this type of planning problem.

3. Literature review

In this section we will describe research literature that are close to our problem setting. The lines of work focus either on maintenance planning (Section 3.1), train scheduling (3.2) or combined approaches (3.3). The vast majority of work concerns train scheduling and operations, but maintenance is starting to receive more attention, while combined approaches are very few.

3.1. Maintenance planning and scheduling

In Liden (2015) a survey of planning problems regarding railway infrastructure maintenance is presented together with an overview of conducted research (until 2014). Here we give a condensed listing and complement with some recent publications.

On a tactical level, three classes of scheduling problems have been identified in the literature: (i) deterioration-based maintenance scheduling, (ii) maintenance vehicle routing and team scheduling, and (iii) possession scheduling. The first class concerns the scheduling of preventive and corrective maintenance, while considering the deterioration of the track. Most papers focus on tamping, and the uncertainty in predicting the degradation plays a vital role. Some recent papers are Vale et al. (2012); Vale and Ribeiro (2014); Gustavsson (2014); Wen et al. (2016). In the second class the aim is to assign and schedule a given set of maintenance tasks to different maintenance teams with varying capabilities, equipment and home locations. The problem is sometimes labelled curfew planning and if traffic is considered it’s done by imposing constraints on how / which jobs can be scheduled simultaneously. Some recent papers are Peng and Ouyang (2012); Borraz-Sanchez and Klabjan (2012); Camci (2015). The third class (possession scheduling) is closest to the problem studied in this paper and the relevant literature is presented below.

A final field of conducted research is work timing and resource scheduling, which usually concerns an operational or mid to short term tactical level. Peng et al. (2013) and Mohammad Pour et al. (2015) study the construction of periodic inspection / preventive maintenance tours while He et al. (2014) and Baldi et al. (2015) consider the selection of which defects / warnings to rectify immediately and which to postpone, using statistical prediction or Markov models for the degradation of the track. The routing and scheduling of corrective tamping machines / teams is studied in Zhang et al. (2013) and Heinicke et al. (2014). All these approaches use some sort of vehicle routing approach but the detailed possession scheduling and interaction with train traffic is not considered.

Possession scheduling. The classic paper Higgins (1998) presents the problem of scheduling maintenance jobs and assigning them to work crews on a single track line with a given train traffic timetable. The objective is a weighted sum of expected interference delay (train delays due to late ending job as well as job delays due to late trains) and prioritised finishing times (as many trains as possible should run on better track). This is a rare example of using a measure that captures both maintenance and train activities. The resulting non-linear model is solved heuristically with a greedy construction phase followed by tabu search. Another approach is taken in Budai et al. (2006), that study the problem of clustering short routine tasks of preventive maintenance and larger projects together, in order to minimize possession and maintenance cost, without any other consideration to the train traffic. A third example is the model for assigning regular work possession patterns (or so called single-track grids) described by van Zante–de Fokkert et al. (2007), where every part of the infrastructure is made available for maintenance at least once every 4 weeks. The primary objective is to minimize the maintenance cost. Consideration to train traffic is handled manually when constructing the single-track grids but no method for dimensioning these patterns is given.
In the above references, the purpose is to schedule maintenance so as to minimize cost or timetable changes. Boland et al. (2013) on the other hand address the problem of adjusting a given maintenance plan for a complete transportation chain so as to maximize the transported throughput. Traffic is handled as flows of trains and the maintenance activities will impose reductions on the link capacities in the network. A similar approach is used in Savelsbergh et al. (2014) who use a given maintenance plan and optimize a transportation plan for a coal freight network. From these plans an assessment of the asset reliability, resource requirements and contract compliance can be made, supported by a decision support system where several performance indicators and visualization possibilities are demonstrated.

3.2. Train scheduling and operations

This research field has an abundance of literature and we refer to the following surveys for an overview of problem types and modelling approaches: Caprara et al. (2007, 2011); Cacchiani and Toth (2012); Cacchiani et al. (2014); Corman and Meng (2014).

The rolling stock planning problems consider maintenance on the trains, but infrastructure maintenance activities are rarely mentioned or handled in the timetabling or tactical train scheduling models. One exception is Forsgren et al. (2013), which will be discussed in Section 3.3 below.

The replanning of a timetable or operative train plan due to track closures (complete or partial) or maintenance activities have been studied in: Brucker et al. (2005) (scheduling of single track traffic past a working site on a line section); Vansteenwegen et al. (2015) (robust rescheduling due to planned track closures on large stations and junctions); and Veelenturf et al. (2015) and Louwerse and Huisman (2014) (rescheduling of timetables, rolling stock and crew during major disruptions in operational dispatching). In all these cases the track closures are given as fixed input.

The problem we consider in this paper also requires trains to be routed and scheduled. We have chosen to use an approach with cumulative variables as presented in Meng and Zhou (2014), but instead of having a detailed spatial and temporal scale we aggregate both the network and time. In Murali et al. (2015) a similar network aggregation is presented, while time is still fine-grained. Neither of these papers consider maintenance scheduling.

3.3. Combined approaches

Approaches for scheduling both trains and work possessions in the same model are presented in Ruffing (1993), Albrecht et al. (2013) and Forsgren et al. (2013). In all cases, a small number of maintenance possessions shall be introduced into an existing train timetable, by allowing different types of adjustments to the trains. Ruffing (1993) is one of few papers that handle operational restrictions (reduced speed) for trains passing a work site. Albrecht et al. (2013) address the real-time operational control case for a single track line, treat maintenance as pseudo trains and allow train times to be adjusted but not cancelled. A probabilistic meta-heuristic procedure (Problem Space Search) is used for obtaining high quality solutions quickly. Forsgren et al. (2013) treat the tactical timetable revision planning case, where a major work possession shall be scheduled but makes it necessary to re-plan the timetable. The model handles a network with both single and multi-track lines, can shift the work start time, allow trains to be rerouted or cancelled and consider different running times depending on train stops. A clique-based MIP model approach is used to solve the resulting scheduling problem.

These examples of combined approaches use an existing timetable as starting point and solution reference. Furthermore, only a small number of work activities, usually one at a time, are considered. Our work, on the other hand, concerns a long-term tactical planning case where a traffic plan does not yet exist and many work slots shall be coordinated with the train traffic. Hence we need a model that captures this situation and handles the specific needs for both trains and maintenance.

4. Mathematical optimization model

In this section we introduce the optimization model and discuss the modelling choices that have been made. All the details, notational definitions and exact mathematical formulations are given in Appendix A. The model might be classified as an Integrated Railway Service Network and Maintenance Design problem.
4.1. Concepts and illustration

The railway network will be modelled with a set of nodes and links. The mapping between these nodes/links and the corresponding parts of the railway network can be done in several ways. The links, which normally correspond to railway lines, control the traffic capacity and each link is independent regarding maintenance scheduling. Nodes should be placed wherever train services shall start or end, where route choices can be made and where line splitting is necessary. Normally the nodes correspond to railway junctions or stations which allow for dwelling, but they may also correspond to points in the railway network where no train stopping is possible. Conversely, it is not necessary to introduce nodes at every station. Instead a line with several meet/pass loops can be treated as one link. Although the model is fully flexible regarding the spatial granularity, it is intended for a macroscopic network view. As for the temporal dimension, the time periods can also be of varying length (typically one to several hours), but the maintenance scheduling (with subsequent changes in capacity) can only start and end at time period boundaries.

Train services are scheduled according to a set of given routes, which enables the use of route-dependent costs and durations. Hence the model is route-based rather than network flow based. The number of routes per train service will depend on the spatial network granularity and with a macroscopic railway view the amount of routes will be small.

Maintenance will be scheduled freely in the planning period according to a set of given window options, defined by number of occasions with certain time length. For example, a total work volume of 10 hours might have the window options 2x5h, 4x3h and 15x1h where the shorter possession times will require more setup time and thus an increased number of occasions, which increases the work cost. As shown in [Liden and Joborn (2016)](https://doi.org/10.1016/j.raileng.2016.05.017) the maintenance cost is inversely proportional to the possession time and also depends on the amount of overhead time done inside and outside of the possessions. This non-linear dependency is the reason for having the window options as given input data.

Figure 1 shows an illustration of a simple railway network (shown at the bottom) with 5 links. The link between D and E is double track while all other links are single track. Trains between A and E can choose route A-B-D-E or A-C-D-E. During 5 hours - divided into five 1h time periods - 6 train services S1-S6 shall be scheduled between A and E together with two hours of maintenance on each link. A scheduling solution is shown in the train and work graph, where time is on the x axis and maintenance windows are indicated with shading. The train graph for A-C-D is shown on the top, where two trains are scheduled while maintenance is blocking A-B-D. The lower half of the train graph shows A-B-D-E, where maintenance can be done on one track of D-E while trains are passing on the other track.

Figure 1: Solution to a small network example
4.2. Model description

We consider a railway network, which is mapped into a graph representation consisting of a set of nodes $N$ and links $L$. The planning horizon is divided into a sequence $T$ of consecutive time periods $t$ with length $\delta_t$. The period lengths determine the granularity for the maintenance scheduling and can either be set to fixed value, e.g. $\delta_t = 1$ hour, or to be varying over the planning horizon. Capacity is controlled for each time period, on a subset $L^C \subseteq L$ of links, by a nominal capacity $C^{Nom}_l$ (number of trains per time unit) under normal operation conditions, and a reduced capacity $C^{Red}_l$ when maintenance is carried out (and one or more tracks are made unavailable for train traffic).

A set of train services $s \in S$ to be scheduled are given with preferred departure time and a set of possible routes $r \in R_s$ - defined as a sequence of links $L_r \subseteq L$ with given traversal direction and minimum duration per link and node. Train cancellations are handled as (expensive) routes consisting of no links. The scheduling of each train service is done with binary route choice variables $z_{sr}$ and real-valued event variables $e^+_{slt}, e^-_{slt}$ for the detailed entry (+) and exit (−) time on link $l$. The event variables will determine in which time period $t$ each train $s$ will enter and exit a specific link $l$, which is tracked by cumulative (binary) link usage variables $x^+_{slt}, x^-_{slt}$. Since each route has a given direction $d$ over a link, the number of trains can be counted with (integer) variables $n^d_{lt}$ for both directions of all links and time periods. The scheduling window for each train service is limited to a subset $T_s$ of the time periods, such that all events and link usages for service $s$ must be within the time range given by $T_s \subseteq T$.

For the maintenance, a required volume $V_l$ (total time) is given for a subset $L^M \subseteq L^C$ of the links, together with a set of maintenance window options $o \in W_l$, each defined by a required number of occasions $n_o$ with possession time (window size) $\theta_o$. Binary window option variables $w_{lo}$ will select the window option for each link $l \in L^M$, and the scheduling of the maintenance windows are done accordingly, with (binary) work variables $y_{lt}$ that indicate whether maintenance is carried out in period $t$ or not and (binary) work start variables $\nu_{lo}$ that show whether maintenance on link $l$ according to work option $o$ is started in time period $t$ or not. The maintenance windows can be scheduled to start at any time period within the planning horizon that allows for completion of the possession time.

The objective function is a cost sum to be minimized, for i) total train running time, ii) deviation from the preferred departures, iii) route cost, iv) direct maintenance work cost, and v) indirect (setup/overhead) time cost. The cost setting for the maintenance windows is discussed in Lidén and Joborn (2016) and these factors will include supplements for night or weekend work hours etc.

The model variables and constraints are illustrated in Figure 2 as a constraint graph with nodes for the variables and arcs showing which variables the constraints operate on. The crucial constraints fall into the categories Traffic, Maintenance and Capacity, as follows:

- **T1.** choice of one route per train service (assignment constraint over $z_{sr}$ variables)
- **T2.** all links in the chosen route should be visited (coupling of $x$ and $z$ variables)
- **T3.** coupling of event times ($e^+_{slt}, e^-_{slt}$) to link usage ($x^+_{slt}, x^-_{slt}$)
- **T4.** enforcing correct travel durations (separation of event variables $e^+_{slt}, e^-_{slt}$) over links and nodes along the chosen routes ($z_{sr}$)
- **M1.** choice of one work option per maintained link (assignment constraint over $w_{lo}$)
- **M2.** fulfillment of the required work volume (this constraint is redundant to the enforcing of window occasions in M4 as discussed in Appendix A)

![Figure 2: Overview of variables (nodes) and constraint couplings (arcs)](image-url)
M3. coupling of work start variables \((v_{tot})\) to option choice \((w_{to})\) and work time \((y_{it})\)

M4. enforcing of window count \((\eta_{o})\) and possession time \((\theta_{o})\) for the chosen work option \((w_{to})\), start \((v_{tot})\) and time \((y_{it})\)

C1. coupling of train counts \((n_{dt})\) to link usage \((x_{+},x_{-})\)

C2. limiting the train counts \((n_{dt})\) according to available capacity \((C_{Nom},C_{Red})\) depending on whether maintenance \((y_{lt})\) is performed or not.

The main coupling of traffic and maintenance scheduling is done by the capacity constraints C2, but runtime supplements can be considered in the T4 constraints to account for traffic restrictions (speed limitations and/or single track operation) during maintenance work as indicated by the dashed arc in Figure 2.

The family of variables and constraints described so far are necessary in any model addressing the same problem. An important design choice is however to use the cumulative variables for tracking the train service link usage \((x_{+},x_{-})\). The cumulative property and calculation of link usage is illustrated in Figure 3 for a train service \(s\) that traverses a link \(l\). It enters the link in period \(i\) (at time \(e_{i}^{+}\)) and exits in period \(i+1\) (at time \(e_{i+1}^{-}\)). Thus \(x_{+}^{+}\) becomes 1 for \(t \geq i\) and \(x_{-}^{-}\) becomes 1 for \(t \geq i+1\). The link usage \((u_{slt})\) is calculated by \(x_{+}^{+} - x_{-}^{-} - 1\) and consistency with the chosen route is guaranteed by checking that the final \(x_{+}^{+},x_{-}^{-}\) values correspond to \(z_{sr}\).

This formulation approach gives easily expressed constraints for the link routing (T2) and event/usage coupling (T3) - especially when the link traversal starts in one time period and ends in another. For further discussion about the use of cumulative variables we refer to Meng and Zhou (2014). In Section 6.3 we will discuss some other modelling approaches that could be considered.

Figure 3: Coupling of event variables for entry/exit \((e^{+},e^{-})\), cumulative variables \((x^{+},x^{-})\), link usage \((u)\) and routing \((z)\). A train service \(s\) enters link \(l\) in time period \(i\), exits in \(i+1\) and will use the link in those periods.

4.3. Problem size and train scheduling window

The model size and growth rate determine how large problem sizes that can be solved. If every train can be scheduled at any time in the planning horizon, the solution space will become huge and contain many symmetries. Thus we need to consider measures for reducing the problem size and breaking symmetries. Most important is to reduce the number of time periods considered per train, i.e. the size of \(T_{s}\), since this will reduce the number of link usage variables \((x)\) and cumulative constraints, which are most numerous. Limiting the train scheduling window to, say, \(|T_{s}| = 6\) for a daily problem with 24 time periods will reduce the number of variables and constraints by almost 75%. Having very tight train scheduling windows might however constrain the model too much and result in inferior solution quality. Therefore we will investigate the effect of varying the train time period size in the computational experiments (Section 5).

In Appendix B formulas for calculating the number of variables \(V\) and constraints \(C\) are given, expressed for problems with a planning horizon of \(p\) days having \(S = S/p\) train services and \(T = T/p\) time periods per
As shown above and in the appendix it is necessary to limit the scheduling window $T_s$ for each train so that $T_s \ll p \bar{T}$. The problem size is then estimated by $V, C = O \left( p \times \left[ 5 \times L_\delta \times T_s + L^M \times W_1 \times T \right] \right)$ which gives a linear growth with the scheduling horizon. For example, a weekly plan divided into one hour slots, for a network with 100 links having 200 daily train services with $|T_s| = 6$, each traversing at most half of the links would need the order of 0.5 M variables and 0.5 M constraints. Further reduction of problem size can be achieved by considering periodic train services, indicating that monthly problem instances is plausible memory wise – however not saying anything about how efficiently such problems can be solved.

4.4. Problem reductions

Apart from reducing the number of time periods per train service, other problem reduction and symmetry breaking methods might be needed. We will investigate two possibilities which address the train scheduling part. (Symmetry-breaking for the maintenance scheduling part is also of interest, since normally there are several time periods with no cost difference, which allows for many alternate and possibly non-distinguishable solutions.)

Train service ordering. For trains with the same origin-destination, routing options and performance, there is no cost distinction between solutions that have the same total runtime and departure deviation. Specifically it makes no sense to allow for the ordering of two such trains (with overlapping scheduling windows), to be altered. Thus we can introduce constraints (see (A.9)) that enforce the wanted train order, with the intention of limiting the search space and brake some of the symmetries.

Scheduling all trains - tighter bounds. In the model, the event variables for all unused routes will become zero, which is consistent with the link usage coupling constraints T3. With the current approach for handling cancellations, which uses no links, a cancelled train will have its initial departure and final arrival also set to zero. Thus the lower bound of all event variables must be zero. For trains that must be scheduled we could use a higher lower bound for the initial departure and final arrival variables, based on the allowable scheduling windows for each train. If we know that it is possible to schedule all trains we could increase the lower bound for all trains and force all cancellation route choices to be zero (or removed).

Alternate ways of modelling train cancellations could also be considered which would allow tighter bounds even for cancellable trains.

5. Computational experiments

This section describes the experiments that have been done in order to establish the behaviour and performance on various problem sizes and structures. After describing the data instances and the basic behaviour of the model, we will present the results from the tests regarding (1) varying problem sizes and train scheduling window in Section 5.2 (2) some additional problem reductions in Section 5.3 and (3) longer computation times for the large cases in Section 5.4.

The model has been implemented in Python with OR-Tools as modelling layer and using the commercial optimization package Gurobi as MIP solver. In our experiments, the Gurobi solver gives small variations in results and performance. For the purpose of this paper, these variations are not significant enough to motivate multiple runs and averaging of the results.

5.1. Test instances and basic behaviour

A set of synthetic problem instances of varying size has been generated. These data instances are available as JSON files together with a set of Python parsers at https://github.com/TomasLiden/mwo-data.git.

All the instances are listed in Table 1 with case name, problem type, size, properties and the number of variables and constraints for the various train scheduling windows used in Section 5.2. There are 9 line instances and 5 network instances, having a planning horizon of five hours to one week divided into $\delta_t = 1h$.

\(^1\) The notation $E = |E|$ is used for the size (number of elements) of a set $E$. 
Table 1: Problem instances and number of variables and constraints. The Case column gives the instance name. Problem size and properties gives the problem type, number of links (L), time periods (T), train services (S) and capacity settings (Cap: CNom,d/CNom). The V and C columns give number of variables and constraints \( \times 10^3 \) for different settings of T.

<table>
<thead>
<tr>
<th>Case</th>
<th>Problem size and properties</th>
<th>( T_s = 2 )</th>
<th>( T_s = 6 )</th>
<th>( T_s = 12 )</th>
<th>( T_s = T )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Type</td>
<td>L</td>
<td>T</td>
<td>S</td>
<td>Cap</td>
</tr>
<tr>
<td>L1</td>
<td>Line 4</td>
<td>5</td>
<td>20</td>
<td>6/10</td>
<td>0.7</td>
</tr>
<tr>
<td>L2</td>
<td>Line 4</td>
<td>5</td>
<td>20</td>
<td>5/9</td>
<td>0.7</td>
</tr>
<tr>
<td>L3</td>
<td>Line 4</td>
<td>12</td>
<td>40</td>
<td>6/9</td>
<td>1.4</td>
</tr>
<tr>
<td>L4</td>
<td>Line 4</td>
<td>12</td>
<td>40</td>
<td>5/10</td>
<td>1.4</td>
</tr>
<tr>
<td>L5</td>
<td>Line 9</td>
<td>24</td>
<td>40</td>
<td>5/8</td>
<td>3.6</td>
</tr>
<tr>
<td>L6</td>
<td>Line 9</td>
<td>48</td>
<td>80</td>
<td>5/8</td>
<td>7.3</td>
</tr>
<tr>
<td>L7</td>
<td>Line 18</td>
<td>24</td>
<td>80</td>
<td>5/8</td>
<td>11.5</td>
</tr>
<tr>
<td>L8</td>
<td>Line 18</td>
<td>96</td>
<td>160</td>
<td>5/8</td>
<td>28.5</td>
</tr>
<tr>
<td>L9</td>
<td>Line 25</td>
<td>168</td>
<td>350</td>
<td>5/8</td>
<td>79.5</td>
</tr>
<tr>
<td>N1</td>
<td>Net 9</td>
<td>5</td>
<td>20</td>
<td>3/5+4/8</td>
<td>0.9</td>
</tr>
<tr>
<td>N2</td>
<td>Net 9</td>
<td>24</td>
<td>50</td>
<td>3/5+4/8</td>
<td>2.8</td>
</tr>
<tr>
<td>N3</td>
<td>Net 9</td>
<td>48</td>
<td>100</td>
<td>3/5+4/8</td>
<td>6.1</td>
</tr>
<tr>
<td>N4</td>
<td>Net 9</td>
<td>96</td>
<td>200</td>
<td>3/5+4/8</td>
<td>12.2</td>
</tr>
<tr>
<td>N5</td>
<td>Net 9</td>
<td>168</td>
<td>350</td>
<td>3/5+4/8</td>
<td>21.2</td>
</tr>
</tbody>
</table>

Note: Variables and constraints are given before presolve, which will remove up to 25% / 10% of the variables / constraints.

Figure 4: Network layout

(a) Line instance L5

(b) Network instances

periods and with 20 to 350 train services. The trains are uniform with no runtime or cost differences - only the preferred departure times differ. The minimum runtime from origin to destination is 1h. These simplifications, which make the trains almost indistinguishable for the solver, are made since we’re mainly interested in the scalability and solvability of the model. Real life instances will of course use more realistic settings for costs and runtimes.

The line instances range from 4 to 25 links, where all but one of the instances (L4) are single track with slightly varying capacity settings. The layout for the one day instance L5 is shown in Figure 4a.

The network instances all share the same network layout (see Figure 4b), which consists of 9 links where all are single track except the two double track links (n1,n2) and (n5,n6). The two alternate routes between n1 and n6 (via n3 and n4 respectively) in effect makes the whole stretch into a double track connection. However, the distance over n3 is shorter and will be preferred whenever possible. Thus the only time the trains will choose the longer route is when forced by limited capacity - as seen in Figure 5 between 15-17h, when a maintenance window on n2-n3-n5 forces two trains to run via n4.

The solution examples shown in Figure 5 are obtained with a default cost setting (given in Appendix C) and with maintenance window sizes 1, 2 and 5 hours. L5 has a time varying cost which favours maintenance placement in the middle of the scheduling period as seen in the figures, while the network instance N2 has no cost difference for the time periods, which explains the arbitrary placement of the maintenance windows for this case. Since the work volume is in the range \([1.7, 2.3]\) a window size of 5h is unnecessarily long. If we add the possibility of scheduling a 3h window we get the results shown in Figure 6. The model behaves as expected - selecting more favourable window options if possible and rescheduling the trains accordingly.
5.2. Varying problem sizes and train scheduling windows

In this experiment, different problem instances and settings for the train scheduling window will be tested. The purpose is to find out how quickly the scheduling problem can be solved, how the performance changes with growing instance size and what effect the train scheduling window has on solution time and quality.

With small train scheduling windows the search space will be reduced which will give shorter solution times, but such scheduling limitations might give inferior solution quality (i.e. a larger objective value). In our tests, the minimum runtime of each train is 1h. Since we use one hour time periods, the smallest possible size of $T_s$ is 2, which means that the train must start in the time period of its preferred departure time and arrive in the next one. The scheduling windows have then been increased by ±2h and ±5h which gives $|T_s| = 6$ and $|T_s| = 12$. Finally we let the trains be scheduled freely in the whole planning horizon ($T_s = T$). The number of variables and constraints for these settings are shown in Table 1. All the detailed test results are given in Appendix C, from which we filter out specific parts and normalise the objective values by subtracting an estimated lower bound (given in the appendix) for easier comparison.

Table 2 presents the final solution values, time to find an acceptable solution (relative gap < 1% or the final solution), total computation time and the relative optimality gap when using train scheduling windows of 2 and 6 time periods. In addition the last column shows the value difference for the two settings of the scheduling windows.

First we note that the double track line instance L4 and the smallest network instances (N1-N2) are easy to solve. For the $|T_s| = 2$ case, 11 instances are solved to proven optimality, 2 more to near optimality while the largest line instance L9 does not reach an acceptable solution within the time limit. Having $|T_s| = 6$ gives longer solving times, such that 5 instances are solved to proven optimal solutions, 6 more to near optimality and the largest line instances L7-L9 obtain poor solutions.

Secondly, we see that for all instances with a near optimal solution, an acceptable solution is found quickly - usually in less than 5 minutes - but that it takes considerable time to close the optimality gap. This indicates a weak linear relaxation.

All instances with a solution value higher than 10 (which is the cancellation penalty) has at least one cancelled train service. Looking at the value difference in the last column, we see that a scheduling window...
Table 2: Solution values, time for acceptable solution (ATime), total computation time (TTime) and optimality gap (** = proven optimal) for scheduling window size 2 and 6. Solution value equals integer objective minus lower bound (LB) in Appendix C. The last column is the value difference for (near) optimal solutions with $|T_s| = 2$ and $|T_s| = 6$.

| Case | $|T_s| = 2$ | $|T_s| = 6$ | Value diff |
|------|-------------|-------------|------------|
|      | Value & ATime & TTime & Gap | Value & ATime & TTime & Gap |          |
| L1   | 66.84 0 0 *** | 1.52 6 240 *** | 65.32      |
| L2   | 59.08 0 0 *** | 2.89 3 300 *** | 56.19      |
| L3   | 54.50 0 0 *** | 2.18 300 3600 0.17 | 52.32      |
| L4   | 1.40 0 0 *** | 1.40 1 1 *** | 0.00 0.00 |
| L5   | 20.56 < 19 19 *** | 2.91 300 3600 1.09 | 17.65      |
| L6   | 12.49 < 60 60 *** | 3.63 480 3600 0.81 | 8.86       |
| L7   | 77.56 300 2460 *** | 29.52 > 3600 3600 16   |
| L8   | 25.00 1200 3600 0.07 | 75.69 > 3600 3600 21 |
| L9   | 5790 > 3600 3600 709 | 5788 > 3600 3600 803 |
| N1   | 2.01 0 0 *** | 2.01 2 2 *** | 0.00       |
| N2   | 3.13 2 2 *** | 3.13 < 120 120 *** | 0.00       |
| N3   | 4.42 300 1560 *** | 4.43 300 3600 0.34 | -0.01      |
| N4   | 7.15 < 120 120 *** | 7.24 300 3600 0.21 | -0.09      |
| N5   | 9.19 300 3600 0.05 | 9.75 300 3600 0.31 | -0.56      |

of 2 hours is too restrictive for the single track line instances, since it results in one or more such cancellation. None of the network instances suffer from this effect and thus a small scheduling window can be used for these cases, while the line instances require a larger setting.

Increasing the train scheduling window further makes the problem even harder to solve without any gain in objective value. Setting $T_s = T$ gives such a large search space that Gurobi cannot find the linear relaxation (LP) solution for L9 and N5 and no integer (IP) solution for L6-L8 within 1h. Hence we conclude that $|T_s| = 2$ can be used for the network instances, while $|T_s| = 6$ works well for most line instances.

In Figure 7, the solution progress for the non-trivial cases solved to (near) optimality with $|T_s| = 6$ is illustrated by plotting two IP values and three LP values. The IP values are the first and last integer solutions, while the LP values are: i) the initial LP value (linear relaxation value of the root node) and, ii-iii) the LP bounds corresponding to the two IP solutions. The objective values have been reduced with the average IP/LP value of the final solution for each case, which centers the values around 0. Thus the y axis gives the objective improvement and the absolute MIP gap. The figure shows that the initial LP value is rather weak, especially for the network instances, and that it is difficult to close the optimality gap for the larger instances. The initial IP value on the other hand is good for all cases except L6 and N5, which shows the heuristic capability of the MIP solver.

In summary we can draw the following conclusions:

- Single track line instances are harder to solve than network instances and double track lines. With a computational time limit of 1h, the largest line instance solved to near optimality is 24-48 h long (L5 and L6) while the network instances can be up to seven days long (N5). The difficulty of the line instances is due to the need for large scheduling changes (train departure deviation and/or longer travel durations) when single track sections are closed for maintenance. In addition, the line instances have a higher capacity utilization than the network instances.

- The train scheduling window can be set to $|T_s| = 2$ for the network instances. For the line instances this setting is too restrictive, since several train services must then be cancelled in order to make room for the required maintenance windows.
5.3. Problem reductions

In this experiment we test how two additional problem reductions will affect the solving efficiency. The purpose is to find out whether improvements in computation time and/or solution quality can be achieved. A time reduction of 1 minute or more is counted as an improvement, while the IP/LP objective value should be at least 0.1 lower/higher to be counted as an improvement.

The reductions are (a) train ordering constraints and (b) tighter bounds when no train cancellations are allowed (see Section 4.4), which are applied to the \(|T_s| = 6 \) case reported in Section 5.2. The reductions are imposed individually and then together, with a separation value \( \Delta = 0 \) in the ordering constraints (A.9). The number of declined/equal/improved measurements for computation time, IP and LP values (first and last) have been collected and plotted in Figure 8 as relative amounts, based on the detailed results given in Appendix C. We can for example see that applying train ordering constraints increases / reduces the solution time for 4% / 16% of the measurements. The IP and LP values on the other hand gets worse / improves in about 28% / 12% of the measurements. The trend is similar for the other settings.

It is hard to draw any clear conclusions from the results. We can however say that:

- Neither of these reductions have any profound effect on the computation time or solution quality.
- The reductions have a small positive impact on computation time, but the LP and IP values do not generally benefit from them.
- Imposing no train cancellations is more beneficial for the solution quality than the train ordering constraints.
- Applying both reductions only slightly improves the computation time, but not the quality.
- When studying the individual instances it can be seen that imposing no train cancellations helps the two large cases L8-L9, where improved IP solutions are found. However it seems to make the IP solutions harder to find and in one case (L7) no IP solution is obtained within the time limit of 1h.

We suspect that the advanced capabilities of modern MIP solvers explain why these reductions have little impact on the end result.
5.4. Longer computation times

In this final experiment we investigate how good solutions can be found for the largest line instances when allowing longer computation times. The purpose is to see if reasonable solution quality can be found within 12h of computation, which is an acceptable requirement in a normal planning situation.

We let the three largest line cases (L7-L9, with slightly varying settings) run for 12h and collect the relative optimality gap with a frequency of 1h, which is shown in Figure 9. Markers are plotted for each new IP solution. The L7 case, which has more trains per time, i.e. a higher capacity usage, needs slightly different settings to obtain good results. The figure shows that L7 and L8 reaches a gap close to 1% after 8 and 2 hours respectively, while L9 needs 12 hours of computation before reaching 2%. This is a reasonable quality but shows the difficulty of solving line instances with many trains or long planning horizons.

Figure 9: Solution progress (relative optimality gap) for instances L7-L9 with longer computation times. Each marker indicate a new IP solution.

6. Discussion

In this section we will first describe some possible extensions and modifications to the model. Then some aspects that make the problem hard to solve are discussed, followed by a section about alternate modelling approaches.
6.1. Extensions and modifications

The model can be extended and/or modified in various ways. Here we focus on aspects concerning the interplay between maintenance and traffic, but will briefly discuss some purely traffic oriented features as well.

Supplements when passing a work site. If trains are allowed to pass a maintenance work site on double or multi-tracks, the safety regulations usually prescribe some speed or operative restriction. Thus a runtime supplement should be added for passing trains. This can be achieved by having indicator variables that track when a train utilizes a link during a maintenance window. The details for achieving this is described at the end of Appendix A and gives a modification to the link duration constraints T4 - also shown with a dashed line in Figure 2.

Supplements for train meets on single tracks. For single track traffic, supplements are needed that capture the meet/pass stops that must take place. For normal traffic this can be included in the regular runtimes but for links that turn into single tracks during maintenance the supplements should be added by the model. To some extent this can be achieved by the method just described (that detect passing of work sites), but this assumes that a reasonable prediction for the number of train meets can be given beforehand. If a more accurate handling is needed an extension has to be made which relies on an additional set of indicator variables that track the opposing train counts.

Periodic trains, long time periods. The model assumes that all train services can be scheduled individually. For timetabled railway traffic it is however common that each service should run the same schedule every operative day although it does not necessarily run all days in the planning horizon. This gives varying capacity usage per scheduling day which opens up for different placement of maintenance windows per day. This is specifically interesting when the maintenance need can be fulfilled with fewer windows than there are scheduling days. The model we have described can support such periodic trains that affect the capacity usage on given operative days over a longer time period. However additional variables will be needed if the operative days are to be decided and/or if the services should be possible to cancel per operative day. The problem size of such a model is discussed at the end of Appendix B.

Unscheduled trains / existing traffic. Certain traffic such as regional and commuter trains might not be interesting to schedule - given that they will not change routes etc due to maintenance and other traffic. Such traffic can be given as a fixed traffic load (number of trains) for each link, time period and direction. These factors are then reducing the available capacity in the C2 constraints or increasing the train counts in the C1 constraints.

Window separation and regularity. There may be scheduling constraints regarding how the maintenance windows should be placed, either depending on how often visits should be made to a certain link, or how many links could have maintenance in the same time period (due to limited maintenance resources). Such constraints or costs remains to be investigated, but these aspects should be possible to include in the model.

Mixing window sizes. Our current model assumes that a window option gives the same possession time for all windows per link. It might however be of interest to mix different window sizes on a specific link. To some extent this can be achieved by letting an option have two or more window sizes, each with a specified number of occasions. A full flexibility would be nice to have, but gives some problems in calculating the amount of overhead time which is the main cost driver for the window options. This issue remains to be further investigated.
6.2. Complicating factors

The computational experiments show that scheduling problems that are close to the capacity limits are difficult to solve, either due to many trains or when tracks are completely closed during maintenance. Another complicating factor appears when train cancellations are needed, especially if there is little cost difference between (a) avoiding cancellations by scheduling many short maintenance windows, and (b) cancelling some trains and scheduling fewer and longer windows. All instances that have redundant track capacity - either by choosing alternate routes or if allowing for traffic past work sites - are easier and quicker to solve.

The linear relaxation of the problem is weak, as shown by the LP values in the experiments and the difficulty of closing the optimality gap. We have studied the linear relaxation solutions and can pinpoint some particular problems regarding the cumulative variables and the coupling between event times and link usage which gives unreliable train counting over the time periods. Similar problems can be observed for the fractional work solution variables which gives a weak estimate of the capacity usage over time. In addition, the same LP objective value can be obtained in a multitude of ways, most of which are structurally far from a correct integer solution. Thus the branching procedure will not get sufficient guidance from the LP solutions. Methods for improving the LP bounds and strengthening the model is subject to future work.

6.3. Alternate model approaches

There are other model approaches that could be considered, some of which we discuss here.

*Non cumulative usage variables.* Instead of cumulative link entry/exit variables, one could use non-cumulative binary variables $u_{slt}$, indicating which time period(s) a train service occupies a link. This approach might however make the routing constraints (T2) weaker, since going from an equality constraint for one variable ($x_{slT} = ...$) to an inequality constraint for a sum ($\sum u_{slt} \geq ...$). Also the event coupling constraints T3 would require indicator variables that detect the entry ($u_{slt} = 0 \rightarrow u_{sl,t+1} = 1$) and exit ($u_{sl't'} = 1 \rightarrow u_{sl,t'+1} = 0$). The number of constraints for upholding these indicator variables would be in the same order as the ones needed for assuring the cumulative property. The drawback of the cumulative variables is however that they give a weak linear relaxation as discussed previously.

*Alternate modelling of cancelled trains.* We have described a model where cancelled trains will get all event variables set to zero, including the initial departure and final arrival. Thus there will be penalties both for selecting the cancellation route and for deviating from the preferred departure time, while no runtime cost will be inflicted. There are alternate modelling approaches, such as having a specific cancellation variable or a dummy route/link (with infinite capacity) that cancelled trains will use. The chosen solution should ideally work well both for individual scheduling of all trains as well as for periodic trains over a multi-day planning horizon.

Another issue that might be considered is that train resource balances should be maintained, particularly when cancelling train services but perhaps in general, which will require inventory constraints on the origin and destination nodes.

*Non-linear modelling, constraint programming.* We have chosen to use linear constraints only. Some constraints (e.g. T3 and M3) could have been expressed in a quadratic or non-linear form which would reduce the number of constraints but make the model non-linear and require another type of solver. Using a finite domain constraint programming approach could also be considered, which might be able to substantially reduce the number of binary variables.

7. Conclusion

We have addressed the integrated scheduling of rail traffic and network maintenance by proposing a mixed integer optimization model suitable for long-term tactical capacity planning. It has been shown that the size of this model grows linearly with the scheduling horizon and that large problem instances - weekly or possibly monthly problems for national networks - is manageable from a memory point of view. Further,
we have shown that a commercial MIP solver can find near optimal solutions to weekly network and double track instances within 1h on a laptop computer, while single track line instances are harder to solve - the largest instances solved to within optimality in 1h are one to two days long. The importance of limiting the scheduling window for the trains have been studied in the computational experiments as well as the effect of using some reduction techniques (train ordering/separation and tighter bounds for non-cancellable trains). The latter have no profound effect on computation time or solution quality but can help for certain instances.

The optimization model which has been presented, schedules both trains and maintenance on an aggregated, long-term scale. Apart from capturing the essential aspects of the planning problem we believe the approach can give valuable results on real life planning instances and that the solutions could help subsequent detailed timetabling methods substantially.

A natural next step is to perform evaluations on real life instances, but there are also several other interesting future research directions. Improvements and further development of the current model could include: resource considerations, periodic trains, better modelling of train cancellations, mixed traffic with overtaking trains, stronger linear relaxation properties, linear inequalities, reductions and symmetry breaking. The handling of maintenance window start/end times could be more flexible - so as to allow fractional times - and the train counting could be more accurate by considering the actual occupancy time in each time period. Other modelling and solution approaches could also be studied.

8. Acknowledgements

This work is performed as part of the research project “Efficient planning of railway infrastructure maintenance”, funded by the Swedish Transport Administration with the grant TRV 2013/55886 and conducted within the national research program “Capacity in the Railway Traffic System”. We thank the two anonymous reviewers for their constructive and helpful comments on the initial manuscript, which greatly helped to improve the quality of the paper.

References


URL http://dx.doi.org/10.1016/S0927-0507(06)14003-7


URL http://dx.doi.org/10.1002/9780470400531.eorms0647


URL http://dx.doi.org/10.1109/TITS.2014.2358392


URL http://dx.doi.org/10.1002/rp2.11501


URL http://dx.doi.org/10.1007/s13676-014-0067-z


URL http://dx.doi.org/10.1057/jors.2014.7


URL http://dx.doi.org/10.1007/s13676-014-0071-3


URL http://dx.doi.org/10.2307/3010526


URL http://dx.doi.org/10.1016/0191-2615(94)00034-W


URL http://dx.doi.org/10.1016/j.trpro.2015.09.011


URL http://dx.doi.org/10.1007/j.13676-014-0067-z


URL http://dx.doi.org/10.1016/j.trb.2014.05.005


URL http://dx.doi.org/10.1016/j.trp.2015.09.011


URL http://dx.doi.org/10.1016/j.trc.2015.08.022


URL http://dx.doi.org/10.1016/j.trb.2012.07.004


URL http://dx.doi.org/10.1016/j.trp.2014.02.003


URL http://dx.doi.org/10.1016/s13676-014-0066-0


Appendices

A. Detailed model description

In this appendix we give a complete description of the MIP model. We will frequently refer to the first or last element in a sequence $E$. To ease the notation we will use the following convention: $\lfloor E \rfloor =$ first$(E); \lceil E \rceil =$ last$(E)$.

Sets and parameters

The fundamental data are:

- $N$: a set of network nodes
- $L$: a set of network links $l = (i, j) : i, j \in N \land i \neq j \land i \prec j$
- $R$: a set of routes $r$, each defined as a sequence of links $(l_k)_{k \in L}$ and the corresponding traversal direction $(d_k)_{k \in L} : d_k = 1$ if the route traverses the link in the $(i, j)$ (forward) direction, 0 otherwise. One route alternative may be a cancellation - consisting of no links.
- $S$: a set of train services $s$ to be scheduled
- $T$: the sequence of (consecutive) time periods $t$ in the planning period
- $W$: a set of maintenance window options.

From these the following subsets and sequences are defined:

- $L_r, N_r$: the links and nodes in route $r \in R$
- $R_s \subset R$: the set of possible routes $r$ for train service $s \in S$
- $L_s, N_s$: all links and nodes that can be visited by train service $s$, given by $L_s = \bigcup_{r \in R_s} L_r$; $N_s = \bigcup_{r \in R_s} N_r$
- $N_s^O, N_s^D$: the origin and destination node for train service $s$
- $L^C \subseteq L$: all links that have a limited capacity
- $L^M \subseteq L^C$: all links where maintenance shall be scheduled
- $T_s$: the allowed time periods for train service $s$
- $W_l \subseteq W$: the set of possible maintenance window options for link $l \in L^M$

In addition the following input data is needed:

- $\tau_s$: the preferred departure time for train service $s$
- $\pi_{srl}$: minimum duration over link $l \in L_r$ when train service $s$ uses route $r \in R_s$
- $\epsilon_{sn}$: minimum duration (dwell time) at node $n \in N_s$ for train service $s$
- $V_l$: the required maintenance volume (total time) for link $l \in L^M$
- $\beta_t, \delta_t$: start time and length of time period $t \in T$
- $\eta_o, \theta_o$: number of occasions and required possession time for maintenance window option $o \in W$
- $C_{l, Nom,d}$: nominal track capacity (number of trains per time unit) in each direction for link $l \in L^C$, during normal operation
- $C_{l, Red,d}$: reduced track capacity in each direction for link $l \in L^M$, when maintenance is carried out
- $C_{l, Nom}$: nominal link capacity (sum for both directions) for link $l \in L^C$
- $C_{l, Red}$: reduced link capacity (sum for both directions) for link $l \in L^M$
- $\sigma_{time}$: cost per travel time for train service $s$
- $\sigma_{dev}$: cost per time unit for train service $s$, when deviating from the preferred departure time $\tau_s$
- $\sigma_{route}$: cost for using route $r$ for train service $s$
- $\lambda_{time}$: cost per time unit when performing maintenance work on link $l$ in time period $t$
- $\lambda_{start}$: cost for starting maintenance work (including setup costs) on link $l$ with window option $o$ in time period $t$
Variables and bounds

The variables and bounds are defined as follows:

$z_{sr}$ route choice: $= 1$ if train service $s$ uses route $r$, 0 otherwise

$e_{sl^+}, e_{sl^-}$ event time: entry(+)/exit(-) time for service $s$ on link $l$, 0 if not using link $l$

$e_{s}^{O}, e_{s}^{D}$ departure/arrival time at the origin/destination for service $s$

$f_{s}$ departure time deviation for train service $s$, i.e. how much $e_{s}^{O}$ deviates from $\tau_{s}$

$x_{sl^+}, x_{sl^-}$ link usage: $= 1$ if train service $s$ has entered/Exited link $l$ in time period $t$ or earlier, 0 otherwise

$n_{lt}^{o}$ number of train services traversing link $l$ in direction $d \in \{0,1\}$ during time period $t$

$w_{lo}$ maintenance window option choice: $= 1$ if link $l$ is maintained with window option $o$, 0 otherwise

$y_{lt}$ maintenance work: $= 1$ if link $l$ is maintained in time period $t$, 0 otherwise

$v_{lot}$ work start: $= 1$ if maintenance on link $l$ according to window option $o$ is started in time period $t$, 0 otherwise

For train services that cannot be cancelled or must be scheduled the lower bound for the $e_{s}^{O/D}$ variables can be set to $\beta_{[T_{i},]}$

The main decision variables are $z_{sr}, e_{sl^+}, e_{sl^-}, w_{lo}, y_{lt}$ and $v_{lot}$. The $x_{sl^+}, x_{sl^-}$ and $n_{lt}^{o}$ variables are indicators/status variables keeping track of the routing and train capacity usage over time. We also define a set of fixed helper variables

$$x_{sl[T_{i},]}^a := 0 \quad \forall a \in \{+,-\}$$

$$y_{lt[T_{i},]} := 0$$

which makes the formulation more compact.

The $x_{sl^+}, x_{sl^-}$ variables are cumulative, i.e. once they have become 1, the latter variables (for that service and link) will also be 1. This property is utilized for detecting in which time periods a train is using a certain link, with the following notation:

$$u_{slt} = x_{sl^+} - x_{sl^+}$$

It is not necessary to define binary variables for $u_{slt}$, but this short hand replacement will be used for notational ease.

Objective function

The objective function (to be minimized) consists of five cost components:

$$q = \sum_{s \in S} \left[ \sigma_{s}^{time}(e_{s}^{D} - e_{s}^{O}) + \sigma_{s}^{dev} f_{s} + \sum_{r \in R_{s}} \sigma_{sr}^{route} z_{sr} \right] +$$

$$\sum_{l \in L^t} \sum_{t \in T} \lambda_{lt}^{time} \delta_{lt}y_{lt} + \sum_{o \in W_{l}} \sum_{t \in T} \lambda_{lot}^{start} v_{lot}$$

(A.1)
The first three components concern the train operations and capture costs for total running time, time deviation from the preferred departure time and route choice (or cancellation). The last two components concern the maintenance work and capture costs for total work and setup/overhead time including their variation for different time periods. The work related components can be combined into just one by removing the $y$-dependent cost and increasing $\lambda_{\text{start}}$ by a factor $\lambda_{\text{time}}^{\theta_0}$. This method is necessary to use if the window options does not have the same work cost per time unit. It also has the benefit of not penalizing unused window time, e.g. when $\theta_0$ is a fractional value of $\sum t \delta_{ylt}$.

**Constraints**

The constraints fall into the three categories Traffic, Maintenance and Capacity. In the following, the left hand labels (T1 etc) refer to Section 4.2 and Figure 2.

**Traffic.** The basic traffic constraints are:

\[ T_1: \quad \sum_{r \in R_s} z_{sr} = 1 \quad \forall s \in S \quad (A.2) \]

\[ x_{a|T_r}|^{s} \geq x_{a|T_r}|^{s-1} \quad \forall s \in S, l \in L_s, t \in T_s, a \in \{+, -\} \quad (A.3) \]

\[ T_2: \quad x_{a|T_r}|^{s} = \sum_{r \in R_l} z_{sr} \quad \forall s \in S, l \in L_s, R_l = \{ r \in R_s : l \in L_r \}, a \in \{+, -\} \quad (A.4) \]

for selecting one route (A.2), assuring that $x$ variables are cumulative (A.3) and that all links in the selected route will be visited (A.4).

The initial departure at the origin node and the final arrival at the destination node are linked to the event times as follows:

\[ e^O_s = \sum_{l \in L^{dep}_s} e^+_{sl} \quad \forall s \in S, L^{dep}_s = \{ [L_r] : r \in R_s \} \quad (A.5) \]

\[ e^D_s = \sum_{l \in L^{arr}_s} e^-_{sl} \quad \forall s \in S, L^{arr}_s = \{ [L_r] : r \in R_s \} \quad (A.6) \]

which also give the departure deviation:

\[ f_s \geq e^O_s - \tau_s \quad \forall s \in S \quad (A.7) \]

\[ f_s \geq \tau_s - e^O_s \quad \forall s \in S \quad (A.8) \]

Train separation and ordering can be enforced between trains that have the same origin-destination and operational class (given by a naming convention, equal performance characteristics or some other criteria). For each group $g = 1...G$ of such equal train services $S^{OD}_g$, sorted in increasing order of preferred departure time, the following constraints enforce the wanted separation (where $\Delta$ is the separation value and the last term invalidates the constraint for cancelled trains):

\[ e^b_s - e^b_{s-1} \geq \Delta - (\beta|T| + \delta|T|) z_{sr} \quad \forall b \in \{O, D\}, g = 1...G, s \in S^{OD}_g \setminus [S^{OD}_g], r \in R_s : L_r = \emptyset \quad (A.9) \]

The following constraints couple the event times and the link usage variables:
\[ e_{st}^a \geq \sum_{t \in T_s} \beta_t (x_{slt}^a - x_{sl,t-1}^a) \quad \forall s \in S, l \in L_s, a \in \{+, -\} \quad (A.10) \]

\[ e_{st}^a \leq \sum_{t \in T_s} (\beta_t + \delta_t - \xi_t) (x_{slt}^a - x_{sl,t-1}^a) \quad \forall s \in S, l \in L_s, a \in \{+, -\} \quad (A.11) \]

where a small fractional value \( \xi_t \) is used in (A.11) to remove ambiguous solutions and achieve that \( \beta_t \leq e_{st}^a < \beta_t + \delta_t = \beta_{t+1} \). (This gives slightly pessimistic bounds, as further commented upon in the last subsection of this appendix.)

Finally, necessary event variable separation is enforced by:

\[ e_{sl}^a \geq \sum_{t \in T_s} \pi_{sr} z_{sr} \quad \forall s \in S, r \in R_s, l \in L_r \quad (A.12) \]

\[ \sum_{(n,j) \in L_s} e_{s(n,j)}^+ - \sum_{(i,n) \in L_s} e_{s(i,n)}^- \geq \epsilon_{sn} \sum_{r \in R_s, n \in N_r} z_{sr} \quad \forall s \in S, n \in N_s \setminus \{N_s^O, N_s^D\} \quad (A.13) \]

for minimum travel durations over links \( A.12 \) and dwell time at nodes \( A.13 \). Constraint \( A.12 \) can be modified so as to allow for train running supplements (as described below). For nodes where train stopping is not allowed, constraint \( A.13 \) should be modified to an equality constraint, with zero as right hand side.

**Maintenance.** The basic maintenance constraints are:

\[ \sum_{o \in W_l} w_{lo} = 1 \quad \forall l \in L_M \quad (A.14) \]

\[ \sum_{t \in T} \delta_t y_{lt} \geq V_l \quad \forall l \in L_M \quad (A.15) \]

for choosing one window options \( A.14 \) and assuring that the required amount of work time is scheduled \( A.15 \).

Then a set of constraints couple the work start variables \( v_{lot} \) to the window choice \( w_{lo} \) and work variables \( y_{lt} \):

\[ v_{lot} + 1 \geq y_{lt} - y_{lt-1} + w_{lo} \quad \forall l \in L_M, o \in W_l, t \in T \quad (A.16) \]

... \[ v_{lot} \leq y_{lo} \quad \forall l \in L_M, o \in W_l, t \in T \quad (A.17) \]

... \[ v_{lot} \leq y_{lt} \quad \forall l \in L_M, o \in W_l, t \in T \quad (A.18) \]

\[ v_{lot} \leq 1 - y_{lt-1} \quad \forall l \in L_M, o \in W_l, t \in T \quad (A.19) \]

Finally, the scheduling for the selected maintenance window option is controlled by:

\[ \sum_{t \in T} v_{lot} \geq \eta_o w_{lo} \quad \forall l \in L_M, o \in W_l \quad (A.20) \]

\[ \sum_{i: \beta_i < \beta_t + \theta_o + \delta_i} \delta_i y_{lt} \geq \theta_o v_{lot} \quad \forall l \in L_M, o \in W_l, t \in T \quad (A.21) \]

regarding number of windows \( A.20 \) and length of each possession \( A.21 \). The constraints \( A.20 \) can be tightened by excluding time periods which do not allow for completion of the possession time \( (\theta_o) \) within the planning horizon. The summation in \( A.21 \) includes one more time period than needed to cover the
possession time, which means that the constraint prohibits two consecutive maintenance windows. To allow this feature the upper limit should be adjusted.

The work volume constraints (A.15) are redundant, since (A.20) will enforce the correct number of windows. If the work options have different staffing levels (and thus different work cost per time unit) the constraints (A.15) should not be used. The redundant constraint can however be useful to add for the MIP solver.

**Capacity / coupling.** The following set of constraints assure the correct train counts $n^d_l$ per link and direction, by summing the appropriate link usage variables $u_{slt}$:

$$C1: \quad n^h_l = \sum_{s \in S, l \in R_s \cap d_i = h} u_{slt} \quad \forall l \in L^C, t \in T, h \in \{0, 1\}$$ (A.22)

Then the following constraints will control the capacity usage under normal running conditions and when capacity is reduced due to maintenance:

$$C2: \quad n^h_l \leq \begin{cases} (1 - y_{lt})C_{l}^{Nom,d} \delta_t + y_{lt}C_{l}^{Red,d} \delta_t & \forall l \in L^M, t \in T, h \in \{0, 1\} \\ C_{l}^{Nom,d} \delta_t & \forall l \in L^C \setminus L^M, t \in T, h \in \{0, 1\} \end{cases}$$ (A.23)

$$C2: \quad n^0_l + n^1_l \leq \begin{cases} (1 - y_{lt})C_{l}^{Nom} \delta_t + y_{lt}C_{l}^{Red} \delta_t & \forall l \in L^M, t \in T \\ C_{l}^{Nom} \delta_t & \forall l \in L^C \setminus L^M, t \in T \end{cases}$$ (A.24)

Here, (A.23) will control the (track) capacity in one direction while (A.24) control the total (link) capacity for both running directions. Hence, mainly the first set of constraints will be active on links corresponding to multi-track lines while both may be limiting for links corresponding to single track lines.

**Extension: passing a work site**

The basic model described above can be extended with various features. One such aspect concerns speed restrictions when passing work sites and/or runtime supplements due to single track traffic in connection with maintenance work. These supplements are introduced by changing the link duration constraint (A.12) to the following form:

$$e_{sl} - e_{sl}^+ \geq \pi_{sr}z_{sr} + \pi_{sl}^w p_{sl} \quad \forall s \in S, r \in R_s, l \in L_r$$ (A.25)

where $p_{sl}$ are binary variables indicating if maintenance work is scheduled on link $l$ when train service $s$ traverses it or not. The first term on the right hand side is the minimum duration as previously and in the second term $\pi_{sl}^w$ is the runtime supplement.

To correctly determine $p_{sl}$, we need indicator variables for each time period, as follows:

$\tilde{p}_{slt}$  concurrent passage and work (binary): = 1 if train service $s$ uses link $l$ in time period $t$ and maintenance work is ongoing, 0 otherwise

The following constraints couple $\tilde{p}_{slt}$ to the link usage ($u_{slt}$) and work ($y_{lt}$) variables and establish the relation between $p_{sl}$ and $\tilde{p}_{slt}$:

24
\[ p_{slt} + 1 \geq u_{slt} + y_{lt} \quad \forall s, l \in L_s \cap L^M : C_i^{\text{Red,d}} > 0, t \in T_s \quad (A.26) \]
\[ p_{slt} \leq u_{slt} \quad \forall s, l \in L_s \cap L^M : C_i^{\text{Red,d}} > 0, t \in T_s \]
\[ p_{slt} \leq y_{lt} \quad \forall s, l \in L_s \cap L^M : C_i^{\text{Red,d}} > 0, t \in T_s \]
\[ 2p_{slt} \geq \sum_{t \in T_s} \hat{p}_{slt} \quad \forall s, l \in L_s \cap L^M : C_i^{\text{Red,d}} > 0 \]

In the last constraint the factor 2 is needed to allow for the case when a train enters the link in one time period and exits in the next, which gives two \( \hat{p}_{slt} \) non-zero values.

If a speed restriction / runtime supplement should be inflicted some time periods before and/or after a maintenance work starts, the \( y_{lt} \) term can be changed to a suitable sum over \( v_{lot} \) in the first and third constraint above.

**Less pessimistic coupling of event times and link usage**

The T3 constraints \((A.10)-(A.11)\) will prohibit or discourage solutions where a train service leaves a link at exactly the period ending time, since a value \( e_{sl} = \beta_t + \delta_t \) implies \( x_{sl,t+1} = 1 \), meaning that the train would use the link also in time period \( t + 1 \). A less pessimistic train counting is achieved by the following constraints:

\[
\begin{align*}
\epsilon^s_{sl} & \geq \sum_{t \in T_s} LB^+_t(x^a_{slt} - x^a_{sl,t-1}) : LB^+_t = \beta_t, LB^-_t = \beta_t + \xi \\
\epsilon^a_{sl} & \leq \sum_{t \in T_s} UB^+_t(x^a_{slt} - x^a_{sl,t-1}) : UB^+_t = \beta_t + \delta_t - \xi, UB^-_t = \beta_t + \delta_t
\end{align*}
\]

which give the boundary restrictions \( \beta_t \leq e^s_{sl} < \beta_t + \delta_t \) and \( \beta < e^a_{sl} \leq \beta_t + \delta_t \). The computational results in this paper have been obtained with the pessimistic formulation. The more optimistic formulation gives only minor changes in quality and/or solution times for all instances, except L2 which gets a structurally different and cheaper solution. Thus all conclusions in the paper holds also for the optimistic formulation.

**B. Problem size**

This appendix lists the number of variables and constraints needed for the model in Appendix A using the notation \( |E| \) for the size (number of elements) of a set \( E \). In Table B.1 the number of variables and constraints are given for the different parts of the model.

Taking the dominant parts for the basic model (not using the extension \((A.23)-(A.26)\)) we find that the number of variables and the number of constraints are given by

\[ V = O(2S \times L_s \times T_s + L^M \times W_t \times T) \]
\[ C = O(S \times (2T_s \times L_s + R_s \times L_t) + 5L^M \times W_t \times T) \]

If assuming that: i) the train services travel over half of the links (i.e. \( L_s = L_r = L/2 \)), ii) they can be scheduled at any time (\( T_s = T \)) and that iii) all links are maintained and capacity controlled (\( L^M = L \)), we get \( V = O((S + W_t) \times L \times T) \); \( C = O((S + 5W_t) \times L \times T) \). Further, if the number of maintenance window options are much fewer than the train services, we have \( V, C = O(S \times L \times T) \).

Under these assumptions, a plan for \( p \) days having \( S = S/p \) train services and \( T = T/p \) time periods per day will have \( V, C = O(S \times L \times T \times p^2) \), which indicates a quadratic growth with the scheduling horizon (in days). For example, a weekly plan divided into one hour slots, with 200 train services per day running over a network with 100 links, would require about 23M (million) variables and equally many constraints.
This growth rate is not tractable and shows the necessity to limit the scheduling window for each train service \((T_s)\), which is reasonable since every train service cannot be scheduled at any time in the planning period \((T)\). Hence it is necessary that \(T_s\) is significantly smaller than \(T\). Using the original counts and again expressing them in terms of daily train services and time periods we get

\[
V = \mathcal{O} \left( 2p \tilde{S} \times L_s \times T_s + L^M \times W_l \times p\tilde{T} \right)
\]

\[
C = \mathcal{O} \left( p \tilde{S} \times (2T_s \times L_s + R_s \times L_r) + 5L^M \times W_l \times p\tilde{T} \right)
\]

which can be simplified to \(V, C = \mathcal{O} \left( p \times \left[ \tilde{S} \times L_s \times T_s + L^M \times W_l \times \tilde{T} \right] \right)\) - by noting that \(R_s \times L_r < 2T_s \times L_s\) and removing numerical values.

With this limitation on the train scheduling window we get a linear growth with the scheduling horizon (in days). Using \(T_s = 6\) and the same assumptions as previously results in about 0.5M variables and 0.5M constraints (with the simplified equation) for a weekly plan, which from a memory point of view is a tractable size.

Including the extensions that add link duration supplements due to passing work sites will increase the amount of variables by \(S \times L^M \times T_s\) and the constraints by \(S \times L^M \times (3T_s + 1)\), which is roughly 50% respectively 100% more - if affecting every link in the network. In practice these variables and constraints will only be needed on a subset of the links (double track lines where train passing is allowed during maintenance), which will limit the growth in size.

Further size reductions is achieved by having periodic trains (as described in Section 6.1), where each train service have the same timetable every day it is operated. With such an approach we get

\[
V = \mathcal{O} \left( 2\tilde{S} \times L_s \times T_s + L^M \times W_l \times p\tilde{T} \right)
\]

\[
C = \mathcal{O} \left( \tilde{S} \times (2T_s \times L_s + R_s \times L_r) + 5L^M \times W_l \times p\tilde{T} \right)
\]

Then the number of variables are moderate while the number of constraints is dominated by the maintenance/capacity part. Using the previous example we will need about 0.2M variables and 0.5M constraints.
for a weekly problem, while a monthly problem instance will need 0.3M variables and 1.3M constraints.

C. Test results

In this appendix all the detailed test results are given. Table C.1 presents optimization results when varying the size of the train scheduling window while Table C.2 concerns different combinations of problem reductions. The $|T_s| = 6$ result columns in Table C.1 are the same as the first result columns in Table C.2 since this is the base case for the problem reduction experiment.

The model used for the experiment in Table C.1 is \( (A.1)-(A.8), (A.10)-(A.24) \), while for Table C.2 the separation constraint (A.9) have been added along with removal of train cancellation variables and imposing of tighter lower bounds for the departure and arrival time variables.

The cost settings are as follows: A normal route choice costs \( \sigma_{route} = 1 \) monetary unit (mu), a cancellation 10 mu, a deviation from the wanted departure \( \sigma_{dev} = 0.1 \) mu/h and the runtime \( \sigma_{time} = 1 \) mu/h. The cost for maintenance windows are set to \( \lambda_{time} = 0.1 \) mu per hour and \( \lambda_{start} = 0.1 \) per possession.\(^2\) With these settings the train cost will dominate and since the shortest runtime for each train is 1h, a rough lower bound for the objective value will be \( LB = (1 + 1)|S| + 0.1\eta_o(1 + \theta_o)|L^M| \approx 2|S| \) mu. Thus a problem with 80 trains will cost at least 160 mu. An upper bound for the train cost is \( \approx (10 + |T|/2)|S| \), given by the cancellation of every train service, which amounts to 1760 mu for 80 trains on a 24h plan.

The result tables are organised as follows: The problem instance name is given in the first (Case) column and the LB value described above in the second column (LB), followed by the test results in the subsequent columns. A set of solution values are given and the wall-clock time it took to find them. The values are given in the form IP/LP/Gap, where IP is the integer solution value, LP is the best linear bound and Gap is the relative difference, calculated as \( (IP - LP)/LP \times 100 \). For all test runs a time limit of 1h has been used and for the instances with more than 50 trains the relative MIP gap in the solver has been set to 0.1%, which will reduce the search space somewhat.

There are six different outcomes given in the result tables:

1. No LP solution found (e.g. last column for L9 and N5)
2. No IP solution found, in which case the LP solution is given (e.g. last column for L6-L8)
3. Time for finding initial (root node) LP solution and best IP value after 1h (e.g. last column for N3 and N4)
4. IP value after 5m and 1h (e.g. the three last columns for L5)
5. Time for finding optimal IP and time for proving optimality - if time differs substantially (e.g. the third column for L1, L2 and N2)
6. Time for finding optimally proven IP (e.g. the first column for L1-L6, N1-N2 and N4).

Cyan coloring have been used for global optimal solutions, which is the best possible objective value when no problem reductions are used. In Table C.2 a bold font mark improvements as compared to having no reductions (first column).

The tests have been run with Gurobi 6.0.5 as MIP solver on a MacBook Pro with a 2.6 GHz Intel Core i5 processor, 8 GB 1600 MHz DDR3 memory and OSX 10.10.5. Apart from the relative MIP gap and time limit, the default parameter settings in Gurobi have been used.

The test results are analysed and interpreted in Section 5.2 and 5.3.

\(^2\) In some of the line instances the maintenance costs have a small variance over the scheduling period so as to favour placement around a specific time of the day (noon).
Table C.1: Solving performance for various train scheduling windows. LB is the rough lower bound for the objective. The "Time" columns give the wall clock time. The three values under "Stat" are IP/LP/Gap. Proven optimal solutions (gap < 0.01) are marked with gap = ***. Global optima are marked with cyan. Run time limit = 1h.

<table>
<thead>
<tr>
<th>Case</th>
<th>Ts = 2</th>
<th>Ts = 6</th>
<th>Ts = 12</th>
</tr>
</thead>
<tbody>
<tr>
<td>L1</td>
<td>40.4</td>
<td>0s</td>
<td>107.24/107.24/***</td>
</tr>
<tr>
<td>L2</td>
<td>40.4</td>
<td>0s</td>
<td>99.48/99.48/***</td>
</tr>
<tr>
<td>L3</td>
<td>80.4</td>
<td>0s</td>
<td>134.90/134.90/***</td>
</tr>
<tr>
<td>L4</td>
<td>80.4</td>
<td>0s</td>
<td>81.80/81.80/***</td>
</tr>
<tr>
<td>L5</td>
<td>80.9</td>
<td>19s</td>
<td>101.46/101.45/***</td>
</tr>
<tr>
<td>L6</td>
<td>160.9</td>
<td>1m</td>
<td>173.39/173.37/***</td>
</tr>
<tr>
<td>L7</td>
<td>161.8</td>
<td>20m</td>
<td>239.36/238.99/0.15%</td>
</tr>
<tr>
<td>L8</td>
<td>321.8</td>
<td>35m</td>
<td>346.81/346.52/0.08%</td>
</tr>
<tr>
<td>L9</td>
<td>702.5</td>
<td>1h</td>
<td>6492.43/802.18/709%</td>
</tr>
<tr>
<td>N1</td>
<td>40.9</td>
<td>0s</td>
<td>42.91/42.91/***</td>
</tr>
<tr>
<td>N2</td>
<td>100.9</td>
<td>2s</td>
<td>104.03/104.02/***</td>
</tr>
<tr>
<td>N3</td>
<td>200.9</td>
<td>10m</td>
<td>205.32/205.20/0.06%</td>
</tr>
<tr>
<td>N4</td>
<td>400.9</td>
<td>2m</td>
<td>408.05/408.01/***</td>
</tr>
<tr>
<td>N5</td>
<td>700.9</td>
<td>10m</td>
<td>710.09/709.64/0.06%</td>
</tr>
</tbody>
</table>

*Table C.1: Showing performance for various train scheduling windows. LB is the rough lower bound for the objective. The "Time" columns give the wall clock time. The three values under "Stat" are IP/LP/Gap. Proven optimal solutions (gap < 0.01) are marked with gap = ***. Global optima are marked with cyan. Run time limit = 1h.*

**Note:** The symbols under "Stat" are IP/LP/Gap. The table contains solutions (gap < 0.01) that are marked with ***, Global optima are marked with cyan. Run time limit = 1h.
Table C.2: Solving performance with problem reductions. (same notation as in Table C.1). Improved values (compared to "no reductions") are marked with **bold** font.

| Case | LB | $|T_s| = 6, no reductions$ | $|T_s| = 6, train ordering$ | $|T_s| = 6, no cancellations$ | $|T_s| = 6, both reductions$ |
|------|----|----------------|---------------------------|-----------------|---------------------------|
| L1   | 40.4 | 6s   | 41.92/41.77/0.36% | 7s   | 41.92/41.92/*** | 5s   | 41.92/41.91/*** | 2s   | 41.92/41.92/*** |
|      | 4m   | 41.92/41.91/*** | | | | | | |
| L2   | 40.4 | 3s   | 43.29/41.56/0.46% | 9s   | 43.29/43.28/*** | 6s   | 43.29/43.29/*** | 5s   | 43.29/43.29/*** |
|      | 5m   | 43.29/43.29/*** | | | | | | |
| L3   | 80.4 | 13m  | 82.58/82.26/0.39% | 15m  | 82.58/82.26/0.39% | 25m  | 82.58/82.37/0.26% | 5m   | 82.58/82.18/0.49% |
|      | 1h   | 82.58/82.44/0.17% | | 49m  | 82.58/82.57/*** | | | 1h   | 82.58/82.42/0.20% |
| L4   | 80.4 | 1s   | 81.80/81.80/*** | 2s   | 81.80/81.80/*** | 1s   | 81.80/81.80/*** | 1s   | 81.80/81.80/*** |
| L5   | 80.9 | 5m   | 83.89/82.68/1.46% | 5m   | 83.93/82.75/1.42% | 5m   | 84.36/82.67/2.04% | 1h   | 83.59/82.90/0.83% |
|      | 1h   | 83.81/82.91/1.09% | | 1h   | 83.79/82.93/1.03% | | 1h   | 83.66/82.86/0.97% | |
| L6   | 160.9 | 5m   | 237.73/162.99/46% | 5m   | 164.92/163.07/1.14% | 5m   | 164.63/163.11/0.93% | 5m   | 164.36/163.14/0.85% |
|      | 1h   | 164.53/163.21/0.81% | | 1h   | 164.48/163.26/0.75% | | 1h   | 164.51/163.16/0.88% | |
| L7   | 161.8 | 5m   | 252.76/164.73/53% | 5m   | 908.32/164.94/451% | 5m   | 164.63/163.11/0.93% | 5m   | 164.51/163.23/0.80% |
|      | 1h   | 191.32/164.89/16% | | 1h   | 871.02/329.01/165% | | 1h   | 164.51/163.23/0.80% | |
| L8   | 321.8 | 7s   | _._/328.23/._._ | 5m   | 2392.85/328.97/0.62% | 5m   | 374.35/329.03/1.4% | 5m   | 370.73/329.06/1.3% |
|      | 1h   | 397.5/329.06/21% | | 1h   | 355.71/329.03/2.03% | | 1h   | 343.12/329.11/4.26% | |
| L9   | 702.5 | 2s   | _._/174.75/._._ | 5m   | 6491.50/718.58/803% | 5m   | 370.73/329.06/1.3% | 5m   | 343.12/329.11/4.26% |
|      | 1h   | 6490.9/718.83/803% | | 1h   | 815.78/719.01/13.5% | | 1h   | 815.78/719.01/13.5% | |
| N1   | 40.9 | 2s   | 42.91/42.91/*** | 5s   | 42.91/42.91/*** | 2s   | 42.91/42.91/*** | 1s   | 42.91/42.91/*** |
| N2   | 100.9 | 2m   | 104.03/104.02/*** | 2m   | 104.03/104.02/*** | 1m   | 104.03/104.02/*** | 1m   | 104.03/104.02/*** |
| N3   | 200.9 | 20m  | 205.33/204.59/0.36% | 20m  | 205.33/204.58/0.37% | 20m  | 205.33/204.63/0.42% | 20m  | 205.33/204.62/0.37% |
|      |      | 1h   | 205.33/204.64/0.34% | | 1h   | 205.33/204.87/0.23% | | 1h   | 205.33/204.60/0.38% | |
| N4   | 400.9 | 10m  | 408.14/407.29/0.21% | 5m   | 408.09/407.30/0.20% | 5m   | 408.93/407.25/0.39% | 5m   | 411.55/407.26/1.05% |
|      |      | 1h   | 408.14/407.30/0.21% | | 1h   | 408.14/407.29/0.21% | | 1h   | 408.14/407.30/0.21% | |
| N5   | 700.9 | 5m   | 712.31/708.37/0.56% | 10m  | 710.74/708.36/0.34% | 5m   | 711.84/708.35/0.49% | 5m   | 712.81/708.33/0.63% |
|      |      | 1h   | 710.65/708.42/0.31% | | 1h   | 710.20/708.39/0.26% | | 1h   | 710.18/708.41/0.25% | |

Note: LB indicates the lower bound, and Stat represents the statistical values.