Solving large-scale train timetable adjustment problems under infrastructure maintenance possessions

Van Aken, Sander; Bešinović, Nikola; Goverde, Rob M.P.

DOI
10.1016/j.jrtpm.2017.06.003

Publication date
2017

Document Version
Accepted author manuscript

Published in
Journal of Rail Transport Planning & Management

Citation (APA)
Van Aken, S., Bešinović, N., & Goverde, R. M. P. (2017). Solving large-scale train timetable adjustment problems under infrastructure maintenance possessions. *Journal of Rail Transport Planning & Management*, *7*(3), 141-156. https://doi.org/10.1016/j.jrtpm.2017.06.003

Solving Large-Scale Train Timetable Adjustment Problems under Infrastructure Maintenance Possessions

Sander Van Aken [a,b,1], Nikola Bešinović [a], Rob M.P. Goverde [a]
[a] Department of Transport and Planning, Delft University of Technology
P.O. Box 5048, 2600 GA Delft, The Netherlands
[b] KU Leuven, Leuven Mobility Research Center – CIB
Celestijnenlaan 300 Box 2422, 3001 Leuven, Belgium
[1] E-mail: sander.vanaken@outlook.com, Phone: +31 (0)6 38 44 98 30

Abstract
During infrastructure maintenance possessions, commonly not all trains can operate, and the original timetable may have to be adjusted accordingly. To deliver the best service to passengers, operators have to coordinate adjustment measures dealing with multiple possessions at the network level. In this paper, we consider the Train Timetable Adjustment Problem (TTAP) and present a mixed integer programming (MIP) model for solving TTAP. In order to solve large-scale problems, such as national Dutch network, and design high quality solutions, modelling extensions are needed. First, we apply three network aggregation techniques to decrease the problem size, which enables to solve instances on the complete Dutch network within satisfactory computation times. Second, we model turnaround activities for short-turned trains and test different strategies. Third, we introduce flexible short-turning possibilities to the MIP to possibly reduce the number of cancelled train lines. We test the proposed model on real-life cases of Netherlands Railways (NS) and assess the effect on computation times and solution quality. Also, we identify differences with current planners' practice. Planners were positive about the quality of generated solutions and the computation speed. The current model can also be used to decide on combinations of time windows for possessions.

Keywords
Railway timetable, Maintenance, Possessions, Train Timetable Adjustment Problem (TTAP), Periodic Event Scheduling Problem (PESP)

1 Introduction

High capacity railway transport is one of the main drivers of public transport and often forms the backbone of a wider network. In the Netherlands, main operator Netherlands Railways (NS) and infrastructure manager ProRail are front-runners in using mathematical models for various planning problems (Kroon et al., 2009), aiming at providing a high level of service. The problem of designing a timetable is known as the Train Timetabling Problem (TTP). Two types of models can be distinguished: macroscopic models consider stations as nodes and tracks as arcs between them, with given capacities, while microscopic models incorporate details such as block sections and signalling constraints. On the macroscopic level, a timetable includes arrival, departure and through times at stations and some other important locations such as junctions. For networks with dense railway traffic, a timetable is often defined on a periodic basis. Serafini and Ukovich (1989) introduced the periodic event scheduling problem (PESP), which is used in operations research-based tool *Designer of Network Schedules* (DONS) to generate the original macroscopic timetable of NS.

However, an increased number of train services results in a higher need for maintenance, which induces a range of additional planning challenges. Conducting infrastructure maintenance requires *possessions*, which are defined as "non-availability of part of the rail network for full use by trains during a period reserved for the carrying out of works" (RailNetEurope, 2015). The reduced available capacity may make the original timetable impossible to operate. In the Netherlands, traffic planners need about 14 weeks to generate a feasible solution for one day of operation, mostly based on experience (Engel, 2016). Thus, mathematical models could significantly speed up this process and generate more efficient solutions.

Lidén (2015) examined the possible maintenance activities, the involved planning problems and the mathematical models that have been developed for those problems. Adjusting the timetable to preventive possessions, i.e., the ones known long in advance, fits within the tactical planning level. Van Aken et al. (2017) defined the Train Timetabling Adjustment Problem (TTAP) to generate an alternative timetable for a given set of long possessions, i.e., lasting one or more days, whilst minimizing the deviation from the original one. They presented a macroscopic PESP-based model to solve the macroscopic periodic TTAP, which considers complete open-track possessions, and station track possessions. However, bigger instances like a complete national network are still too large to be solved by the model presented in Van Aken et al. (2017). Currently, each planner at NS considers only a small part of the network and the relevant possessions, based on his own experience. This may lead to measures conflicting with those of other planners, resulting in the need for multiple iterations. The additional value of a macroscopic model is the possibility to coordinate adjustment measures to deal with possessions for a complete network.

In this paper, we extend the previous work on solving TTAP by making it applicable to large-scale instances, and by including more real-life constraints into the model. First, we apply three network aggregation techniques to reduce the size of the problem while maintaining a required level of detail, and study the effect of different aggregation levels on the solution quality and the computation speed. Second, we implement the additional turnaround activities for short-turned trains to prevent possible station capacity violations. To this purpose, we introduce and evaluate different short-turning strategies and consider the integration of these turnarounds both in the preprocessing and postprocessing resulting in four different procedures. Third, we present a flexible short-turning procedure as alternative for the fixed preprocessing step in Van Aken et al. (2017). Finally, we test the model for solving TTAP on a real case and compare our solutions with the ones obtained by planners, explaining differences, thereby identifying possibilities for future research.

The remainder of this paper is structured as follows. Section 2 presents previous work relevant to possession scheduling, network aggregation, and the TTAP; and distinguishes our research. Section 3 summarizes the model developed in Van Aken et al. (2017), which serves as basis for our current work. The applied network aggregation techniques and procedures for turnarounds of short-turned trains are described in Sections 4 and 5, respectively. Additionally, the latter introduces the flexible short-turning concept. Section 6 presents three case studies and evaluates effects of different levels of network aggregation, and the different procedures, in terms of computation speed and solution quality. Furthermore, it discusses a real-life case study and its conclusions. Section 7 gives an overview of results and indicates directions for future research.

2 Literature Review

Lidén (2015) presented a survey on the possible maintenance possessions, the involved planning problems and the mathematical models that have been developed for those problems. The scheduling of infrastructure maintenance actions is important on all levels of planning: strategic, tactical and operational. Depending on its time of planning, a possession can be categorized as either preventive or corrective. The former are defined as "maintenance that can be planned long in advance" (Forsgren et al., 2013), such as renewal and replacements of existing tracks (Budai-Balke, 2009). Lidén (2015) also distinguished between two types of possessions: major ones, which cause conflicts with scheduled trains paths; and minor ones, which do not interfere with train operations. In addition, a train path is "the infrastructure capacity needed to run a train between two places over a given period" (RailNetEurope, 2015). We consider major preventive possessions on a tactical level, and in particular, possessions that take one or more days to be completed. Examples of such possessions are renewal works of the station platform, which may cause possession of neighbouring tracks up to several weeks, or repairing signals along the track between two stations taking the whole working day. Since we consider major possessions, adjustments to a timetable are necessary. We distinguish three approaches to tackle the problem regarding major possessions: (a) scheduling only maintenance windows without changing the timetable, which can also be considered as a strategic problem, (b) adjusting timetables for given maintenance possessions and, (c) scheduling train traffic and track possessions simultaneously.

Lidén and Joborn (2015) presented a model for the first approach, where they assessed and dimensioned maintenance windows before a timetable is generated. Kidd et al. (2016) and Vansteenwegen et al. (2016) proposed a train routing model for adjusting a timetable and corresponding train routes due to given possessions, which corresponds to the second approach. The former was tested on the Copenhagen Metro line, and latter on the network around Brussels Central station. Arenas et al. (2017) formulated a MILP model to delay and reroute trains, close to operations, for a given possession, and considered scheduling additional train paths for maintenance trains. Without considering train cancellation, most of the generated timetables were feasible on the microscopic level. Albrecht et al. (2013) and Forsgren et al. (2013) tackled the third approach and both papers solved a scheduling problem for small cases with at most one or two given possessions. Albrecht et al. (2013) proposed a heuristic to solve the problem, while Forsgren et al. (2013) used an exact approach. Also, Luan et al. (2017) developed an integrated MILP model by considering maintenance possessions as virtual trains. For the other trains, the deviation from a given timetable was minimized. They used a cumulative variable approach to model track capacity, thereby incorporating microscopic details. Larger instances required a Lagrangian relaxation framework, which not always resulted in the optimal solution within a given time limit. All approaches focused on non-periodic timetabling and are limited to small networks and small number of possessions, most often one or two.

Van Aken et al. (2017) followed the second approach and introduced the periodic TTAP. They developed a macroscopic model based on PESP to deal with possessions resulting in the closure of a number of station platform tracks, and complete closure of tracks between two stations. The model considers retiming, reordering, short-turning and cancelling trains to generate an alternative periodic timetable. To reduce computation times, a row generation approach was applied to deal with station capacity constraints. The model has been applied to scenarios with multiple possessions on three subnetworks of the Dutch railway network, the largest one comprising one third of the national network. Computation times remained low for all instances; however, the model is not suitable for solving the complete network yet due to the increased problem size.

NS uses DONS to generate new periodic timetables, with a model based on the PESP formulation. In order to reduce the size of the very large constraint graphs representing the complete Dutch network, DONS applies several techniques. Peeters (2003) and Polinder (2015) describe the basic techniques of removing parallel arcs, i.e., arcs that connect the same pair of events, by intersecting their time windows. Additionally, as the running time arcs in DONS have a fixed time duration, they are combined with the subsequent dwell process, which allows to remove the arrival events from the constraint graph.

Louwerse and Huisman (2014) developed a macroscopic model for disruption management in case of partial closure of a double-track line. They allow events to be delayed, thereby limiting the maximum delay. To reduce the problem size, they remove headway activities if the originally scheduled time between both events is larger than the maximum allowable delay. However, they consider the problem as aperiodic, in which headway activities only have a lower bound.

This paper extends the model developed in Van Aken et al. (2017) by incorporating turnaround activities for short-turned trains, considering several procedures. Network aggregation techniques similar to the ones described in Peeters (2003) and Polinder (2015) are applied, but tailored to the TTAP. We extend the idea of removing headway constraints as described in Louwerse and Huisman (2014) to include timetable periodicity. Finally, we consider the complete Dutch network instead of parts of it as in Van Aken et al. (2017), and compare our alternative timetables with the ones generated by planners of NS.

3 A Basic Model for the TTAP

This research builds on the macroscopic model developed in Van Aken et al. (2017) for the periodic TTAP. This section introduces the model, displays the basic concepts and explains the applied logic. We refer the reader to the original paper for a more extensive description of the model.

The input consists of an original macroscopic periodic timetable, i.e., events representing the arrival and departure times π_i at timetabling points such as stations, stops, and switches, comprising the set of nodes E. Events repeat with a cycle time T. Activities a_{ij} link events and define lower (l_{ij}) and upper (u_{ij}) bounds on the time duration, or process time, between the respective events. Different types of activities exist, representing running times, dwell times, passenger connections, train couplings, and train turnarounds. Additionally, headway activities ($A_{headway}$) ensure that conflicting train paths are sufficiently separated in time. Together, events and activities represent a periodic *event-activity graph* $G = (E, A)$, for which Nachtigall (1996) was the first to describe the Periodic Event Scheduling Problem (PESP), to derive a railway timetable. Odijk (1996) was the first to consider headway constraints in PESP models. Mathematically, the constraints can be represented as follows:

$$\begin{aligned} & l_{ij} \leq \pi_j - \pi_i + p_{ij}T \leq u_{ij} && \forall (i,j) \in A && (1) \\ & 0 \leq \pi_i \leq T-1 && \forall i \in E && (2) \\ & p_{ij} \in \{0,1\} && \forall (i,j) \in A && (3) \end{aligned}$$

Constraints (1) represent limits on the duration between activities. If $u_{ij} \leq T-1$ for all activities, then binary variables p_{ij} represent the order between events i and j within the period, i.e., $p_{ij} = 0$ if $\pi_i \leq \pi_j$ and $p_{ij} = 1$ otherwise. In case any $u_{ij} \geq T$, p_{ij} is integer (Peeters, 2003).

The model deals with two types of possessions: (1) an open-track possession of all tracks between two timetabling points; and (2) a possession of a number of platform tracks in a station. Hence, complete closure of a station can be modelled as a combination of open-track possessions of all segments (i.e., the average number of events and processes remaining after network) connected to the station. In case of an open-track possession, trains can no longer run over the closed infrastructure, and the graph G needs to be altered. Van Aken et al. (2017) apply a preprocessing step to short-turn trains as close as possible to the open-track possessions, while not considering turnaround activities. Thus, the graph G is changed accordingly and serves as basis for the extended PESP formulation.

Possessions in stations may result in a capacity shortage at these locations. On the macroscopic level, we do not consider the platform track of a dwelling train. Instead, the number of occupied platform tracks is considered as proxy for station capacity. We apply three measures to deal with limited station capacity: event retiming, reordering and cancellation of complete train lines. The original PESP model is extended to include these measures as follows:

$$\begin{aligned} & d_j^+ = v_j + T\alpha_j - \pi_j(1-x_m) && \forall j \in E && (4) \\ & 0 \leq v_j \leq (T-1)\,(1-x_m) && \forall j \in E_m,\ m \in M && (5) \\ & l_{ij}(1-\gamma_{ij}) \leq v_j - v_i + q_{ij}T \leq u_{ij} + (T-1)\,\gamma_{ij} && \forall (i,j) \in A && (6) \\ & 0 \leq d_j^+ \leq d_{max}^+; \quad \alpha_j \in \{0,1\} && \forall j \in E && (7) \\ & x_m \in \{0,1\} && \forall m \in M && (8) \\ & q_{ij} \in \{0,1\} && \forall (i,j) \in A && (9) \end{aligned}$$

Constraints (4) account for the retiming of trains by linking the time of event j in the alternative timetable, v_j, to the original one, π_j, with a delay d_j^+. If event j is delayed across the period border ($\alpha_j = 1$), we calculate d_j^+ based on the event in the next period of the alternative timetable. Additionally, constraints (7) state that delays have to be positive and cannot exceed a certain maximum value d_{max}^+. Cancellation of train line $m \in M$ is modelled by a binary variable x_m, which equals 1 if m is cancelled. Constraints (5) set all corresponding event times of m, i.e., the set E_m, to 0. Incorporating x_m in constraints (4) ensures that $d_j^+ = 0$ for the corresponding events. However, if a train is cancelled, corresponding activities have to be relaxed. Note that some activity sets, e.g., headway activities, link events of two different train lines. Constraints (6) relax the bounds on the activities, depending on the involved train lines. If events i and j belong to the same train line m, then $\gamma_{ij} = x_m$. If they belong to different train lines, m and n respectively, then the lower and upper bounds have to be relaxed in case of cancellation of at least one of them: $\gamma_{ij} = x_m + x_n$. Hence, constraints are relaxed if $\gamma_{ij} = 1$ or 2. Reordering is modelled implicitly, as variables q_{ij} do not depend on the original event orderings p_{ij} in constraints (9).

For each station, two sets of constraints are used to model station capacity:

$$(1 - c_{ij} - \gamma_{ij})\delta_i \leq v_j - v_i + Tq_{ij} \leq T - 1 - \delta_j(1 - c_{ji} - \gamma_{ij}) \quad \forall i \in X, \forall j \in Y \quad (10)$$

$$\sum_{i \neq j} c_{ij} \leq N^s_{tracks} - 1 \quad \forall j \in X \cup Y \quad (11)$$

Binary variables c_{ij} are used to identify whether a train line m with arrival event $i \in X$ is present at station s when train line n arrives (event $j \in Y$). Parameters δ_i and δ_j represent the time train lines m and n spend in the station after their arrival, respectively. Sets X and Y represent different types of events, such as arrivals connected to a following departure or final arrivals, both with their corresponding δ_i and δ_j. For more explanation, the reader is referred to Van Aken et al. (2017). The times for the corresponding dwell and turnaround activities are assumed to be fixed. Hence, $c_{ij} = 1$ if n arrives during the time m spends at the station. Constraints (11) limit the number of trains that can be present at the station when train n, with arrival event j, arrives, depending on the number of available tracks N^s_{tracks}.

Peeters (2003) modelled station capacity constraints into PESP before, but that resulted in out-of-memory problems. To circumvent these computational issues, Van Aken et al. (2017) proposed a row generation approach based on the assumption that station capacity is sufficient to accommodate all trains in the original timetable. First, it is checked whether station capacity is violated at any of the timetabling points s with a possession. Next, the corresponding TTAP is solved. However, station capacity has to be checked again, as it may become violated at other stations. If $S_{violated} \neq \emptyset$, additional station capacity constraints are added, and the TTAP is solved again. Otherwise, all constraints and station capacities are satisfied, and we have a feasible alternative timetable.

The goal of the TTAP is to minimize passengers' inconvenience caused by these measures. Assuming the original timetable is good to passengers, this goal is formulated as a weighted objective function:

$$\text{minimize} \sum_{j \in E_{arr}} w^{delay}_j d^+_j + \sum_{m \in M} w^{cancel}_m x_m. \quad (12)$$

Objective function (12), with E_{arr} being the set of arrival events and M the set of lines, penalizes delays of arrivals at all stations and train cancellations. As a cancelled train is much worse for passengers, in general $w^{cancel}_m \gg w^{delay}_j$. Short-turning was defined in the preprocessing step and could not be changed by the optimization model. Despite the fact that it also has a considerable impact, it is omitted from the objective function. In the remainder of this paper, we refer to this model as the *initial model.*

4 Network Aggregation Techniques

A macroscopic model for solving TTAP reveals its full potential when it is applied to the complete network instances. In this way, adjustment measures dealing with multiple simultaneous possessions at different locations on the network are coordinated. Currently, a planner considers only a small area, e.g., a main station, at once, which may lead to inconsistencies with other planners' results and requires several iterations. Here, we present three network aggregation strategies that reduce the number of events and activities in the graph G from Section 3. We approximate delays for removed events to obtain comparable objective function values. Network aggregation is carried out after the preprocessing for short-turning, and by doing so, it does not affect the number of short-turning possibilities. This preprocessing step took at most 1 s for the instances in Van Aken et al. (2017). Section 6.1 evaluates the effect of different network aggregation levels on generated solutions.

4.1 Merging Arrival and Departure Nodes for Passing Events

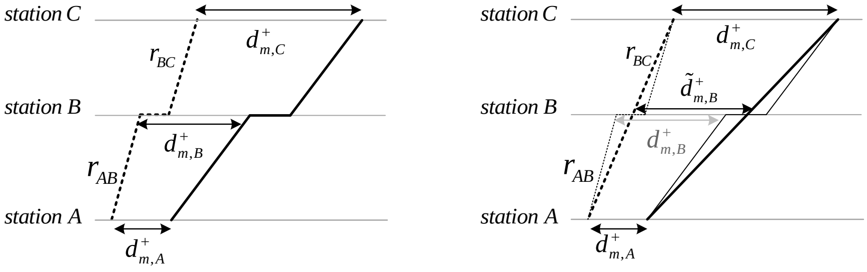
The initial graph G models the passing of a train at a station as two events: an arrival-through, and a departure-through, occurring at the same time. Louwerse and Huisman (2014) merged these events as one, which reduced the number of events in their model considerably. We adopt this approach by removing the departure-through events, and replace the events in the associated headway activities by the respective arrival-through events. This may result in parallel arcs, which are removed by intersecting the time windows as described in Peeters (2003) and Polinder (2015). The objective function (12) does not consider delays of through events, hence, its value does not change.

4.2 Removing Stations and Timetabling Points

Events $i \in E$ occur at a specific timetabling point $s \in S$, which can be for example a (large) station, open-track stop, shunting yard, bridge, or junction. To reduce the size of the network, less interesting timetabling points can be removed. Examples are open-track stops, and passing times at shunting yards and junctions. Let E_s be the set of all events occurring at timetabling point s. We develop four criteria to choose the appropriate timetabling points s, and corresponding events (E_s), which can be removed while maintaining important details in the model.

First of all, station s cannot be removed if E_s holds the first departure or final arrival event of one or more train line(s) in the original timetable (criterion 1). We want to maintain the information on these events, as they may affect station capacity significantly. Additionally, we may still want to add turnaround activities for these events to ensure rolling stock circulations. As we can no longer evaluate station capacity at removed timetabling points, criterion 2 states that s cannot be removed if (an) open-track possession(s) results in trains being short-turned at s. These locations are of particular importance as they are potential locations where station capacity becomes violated due to turnaround activities. Additionally, we do not remove stations s at which a possession reduces the number of available station tracks (criterion 3). Finally, criterion 4 states that timetabling points with more than two track segments connected to it are not removed. For example, two trains arriving or passing through close in time, but originating from different timetabling points, have a headway activity defined between the events. Removing such events would make it impossible to identify how close they get in time. Additionally, no headway process at the previous timetabling points of these trains was present, as they originate from different incoming segments.

The first term in objective function (12) contains the delay of all arrival events only. However, when a train dwells at a timetabling point, i.e., a station or stop, which has been removed from the network, the delay at that location can no longer be determined exactly. This alters the objective function and may affect the final solution. We avoid this as much as possible by estimating the delays at the removed timetabling points based on the delay at



(a) (b)

Figure 1: Train m is scheduled to stop at stations A, B, and C. Dashed lines indicate the original train path with running times r_{AB} and r_{BC}, the full lines the one in the alternative timetable. (a) Delays at all stations are known exactly. (b) When station B is removed from the network, we loose track of delay $d_{m,B}$. To include it in the objective function, we approximate it based on $d^+_{m,A}$ and $d^+_{m,C}$.

the previous and next timetabling point.

Figure 1a shows the path of a train through a series of three timetabling points for both the original (dotted line) and alternative timetable (full line), indicating the (arrival) delay of train m at each timetabling point s, $d^+_{m,s}$. Removing station B results in Figure 1b, with an unknown delay $d^+_{m,B}$ at B. We assume that the delay increases or decreases at a constant rate from $d^+_{m,A}$ to $d^+_{m,C}$. This logic can be translated to an approximation $\tilde{d}^+_{m,B}$ using changed weights for the delays at A and C. Let $w_{m,arr,s}$ represent the weight of delays for the arrival event of train line m at station σ, then:

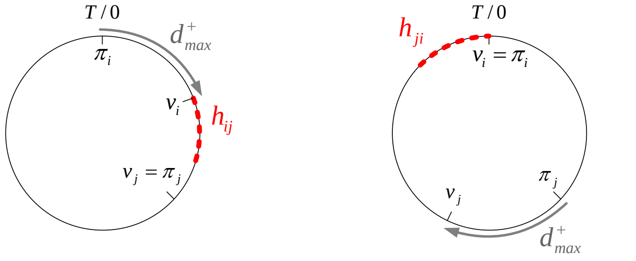
$$\tilde{d}^+_{m,B} = \frac{r_{BC}}{r_{AB} + r_{BC}}\, d^+_{m,A} + \frac{r_{AB}}{r_{AB} + r_{BC}}\, d^+_{m,C} \qquad \forall m \in M_B \qquad (13)$$

$$w'_{m,arr,A} = \left(1 + \frac{r_{BC}}{r_{AB} + r_{BC}}\right) w_{m,arr,A} \qquad \forall m \in M_B \qquad (14)$$

$$w'_{m,arr,C} = \left(1 + \frac{r_{AB}}{r_{AB} + r_{BC}}\right) w_{m,arr,C} \qquad \forall m \in M_B \qquad (15)$$

Equation (13) estimates the delay $d^+_{m,B}$ as a weighted sum of $d^+_{m,A}$ and $d^+_{m,C}$ for each train line m stopping at B (set M_B). Weights are calculated based on the running times r_{AB} and r_{BC} in the original timetable between stations A and B, and B and C respectively. Recall that dwell times at stations are considered to be fixed. Hence, an increase or decrease of the delay can only occur during the running processes. The approximation $\tilde{d}^+_{m,B}$ gets closer to $d^+_{m,A}$ when the running time r_{AB} is smaller relative compared to the total running time from A to C ($r_{AB} + r_{BC}$), i.e., the sooner the train arrives at B after departing from A. Equations (14) and (15) adjust the weights $w_{m,arr,A}$ and $w_{m,arr,C}$ of the events representing the arrival of train line m at A and C, respectively.

Note that this approximation can both over- or underestimate the delay at station B, but it is not possible to predict this on beforehand. Hence, the objective function value is slightly altered, and the extent is independent of the number of timetabling points removed from the graph.



(a) $d_i^+ = d_{max}^+$; $d_j^+ = 0$

(b) $d_i^+ = 0$; $d_j^+ = d_{max}^+$

Figure 2: Unit time circles representing one period with cycle time T. Events i and j with times π_i and π_j that are sufficiently separated in the original timetable, will never violate the corresponding minimum headways h_{ij} and h_{ji} (indicated as red dotted arcs). The events times v_i and v_j are bounded due to the maximum delay d^+_{max}. We call these "non-essential headways".

4.3 Non-essential Headway Constraints

A *non-essential headway* is a headway activity in G that can never be violated for a given TTAP and d^+_{max}. Figure 2 shows two events i and j on a circle representing one period of length T, and the required minimum headways h_{ij} and h_{ji} (as red dotted segments). Assume $\pi_i = 0$. Figure 2a shows that if we schedule event i as close as possible to event j by delaying i with d^+_{max} and fixing j, the headway constraint never becomes violated if $\pi_j \geq \pi_i + h_{ij} + d^+_{max}$. On the other hand, scheduling event j as close as possible to event i (Figure 2b) will never result in headway violations if $\pi_j + d^+_{max} \leq \pi_i + T - h_{ji}$. Extending this to the general case, provides the following condition:

$$h_{ij} + d^+_{max} \leq \pi_j - \pi_i + Tq_{ij} \leq T - h_{ji} - d^+_{max} \tag{16}$$

All headway activities $a_{ij} \in A^{headway}$ for which condition (16) is satisfied, can be removed from the event-activity graph without altering the final solution.

5 Including Turnaround Activities for Short-Turned Trains

As mentioned in Section 3, the initial model applies train short-turning near open-track possessions in a preprocessing step, but does not add turnaround activities at the short-turning locations. However, these are needed to ensure consistent rolling stock circulations. In general, train turnarounds last longer than train dwellings, and may consume a significant amount of station capacity, possibly resulting in station capacity violation.

Here, we discuss several possibilities to include turnaround activities at short-turning locations. First, Section 5.1 presents two possible strategies to decide which incoming train will turn around on which outgoing one. Next, Section 5.2 discusses two possibilities for when to decide on these activities: before or after the optimization model? We call the combination of choices on both aspects, the *turnaround procedure*, represented by a tuple (∗`strategy`∗ ; ∗`when in model`∗). Finally, Section 5.3 discusses an alternative for the fixed short-turning, migrating the decision on train short-turning location from the preprocessing step to the optimization model.

5.1 Turnaround Strategies

The preprocessing step decides at which locations trains are being short-turned. For each of these locations, turnaround activities have to be added to the event-activity graph G. Deciding which trains to assign to a turnaround depends on various conditions such as event times, minimum and maximum turnaround times, and train composition. These cannot all be incorporated within a macroscopic model with the purpose of generating an alternative timetable. In general, rolling stock is considered only in a next planning step, with a possible feedback loop to the timetabling stage. Here, we take two main criteria into account:

1. Trains can only turnaround and connect to trains of the same service type, i.e., regional trains connect to regional trains (R), and so do intercity trains (IC).
2. Minimum turnaround times τ_{min} for a certain rolling stock type have to be respected.

Minimum turnaround times τ_{min} depend on several factors: rolling stock type, operational procedures, and the availability of a second train driver on the platform. If no additional driver is present, the current train driver has to walk to the other side of the train, which in most cases results in increased τ_{min}. Here, we present two strategies based on planners' practices, depending on whether shunting tracks are present near the station or not.

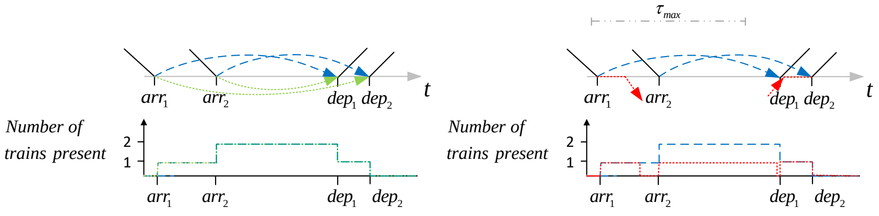
The FCFS Strategy

Assume no shunting tracks or yards are present at the station, i.e., incoming trains will wait on the platform track until they leave again as the outgoing one. Our first strategy adds a turnaround activity between an incoming train, and the first outgoing one satisfying both conditions, following a first-come-first-served (FCFS) logic. Figure 3a shows a station with two arriving trains, which can both turn around on two departing ones, illustrating it does not matter whether a FCFS (blue dashed arrows), or a last-come-first-served (LCFS, green full arrows) logic is applied. In our macroscopic model, station capacity consumption is considered as the number of trains present at the same time, which only depends on the arrival and departure times. Hence, it does not matter whether a FCFS or LCFS strategy is adopted.

When a turnaround activity is added, a process constraint (6) and (possibly) several station capacity constraints (10) have to be included in the model. Assume that at a station σ, incoming train line m with arrival event i is short-turned to train line n with departure event k. Hence, the turnaround time can be calculated as $\tau_{ik} = v_k - v_i + Tq_{ik}$. A corresponding process constraint is added:

$$\tau_{ik}(1 - x_m - x_n) \leq v_k - v_i + q_{ik}T \leq \tau_{ik} + (T-1)(x_m + x_n) \tag{17}$$

As we assume times spent at the station platform track to be fixed, both the lower and upper bound of constraint (17) are equal to the calculated turnaround times, meaning that if arrival i is delayed, departure k is delayed with the same time. The γ_{ij} factor of constraints (6) is replaced with $x_m + x_n$, as cancellation of either train line relaxes the constraint. In case station capacity is violated at the short-turning location, constraint sets (10) and (11) have to be added to the model. Constraints (10) can be used as formulated, with $\delta_i = \tau_{ik}$.



(a) FCFS and LCFS

(b) maxFCFS

Figure 3: Illustration of the impact of turnaround strategies on station capacity consumption, showing a station at which short-turning is applied, with two incoming trains (arrival times arr_i and arr_j) and two outgoing trains (departure times dep_i and dep_j). (a) Both FCFS (dashed blue lines) and LCFS (full green lines) strategy result in the same number of trains present in time. (b) The maxFCFS strategy with maximum turnaround time τ_{max} (red full lines), reduces the number of station platform tracks needed to one, compared to two for the regular FCFS (dashed blue lines).

The maxFCFS Strategy

A second strategy that is commonly used by planners, is shunting a train to a nearby shunting track or shunting yard to free up a station platform track, if turnaround activities take longer than a maximum turnaround time τ_{max}, e.g., 10 min. We define this as the *maxFCFS strategy*. If a train has to be shunted, it occupies the platform for a certain time t_{shunt}, e.g., 2 min, which is shorter than τ_{max}. Figure 3b shows a situation in which this reduces the minimum number of platforms needed: the maxFCFS strategy (red dotted lines) allows to shunt the first arriving train, getting it back to the station afterwards. Hence, in case only one platform track remains at the station, this strategy may avoid the cancellation of trains.

For turnaround activities with a duration $\tau_{ik} \leq \tau_{max}$, constraints can be adjusted in the same way as for the FCFS strategy. However, if trains are shunted, a turnaround constraint (17) is not added, but the shunting time t_{shunt} has to be taken into account when adding station capacity constraints. For the arrival event i, t_{shunt} represents the time spent at the station platform track after event i, and constraints (10) can be used with $\delta_i = t_{shunt}$.

Before departure event k, the train is shunted back to the platform track, thereby occupying it for t_{shunt}. Hence, for identifying whether another train with arrival event j is present at the same time, we have to relate v_j with the arrival of the shunted train in constraints (10) and $\delta_k = t_{shunt}$. However, simply taking $v_k - t_{shunt}$ as the arrival event of train n does not take into consideration the possibility of crossing the period border during the shunting process. Assume $v_k = 1$ min, $t_{shunt} = 2$ min and $T = 60$ min, the train will be shunted to the platform track and occupy it from time 59 in the previous period, until time 1 in the current period. To identify these situations, the following constraints are added:

$$0 \leq v_k - t_{shunt} + Tq_k^{shunt} \leq T - 1 \qquad\qquad \forall k \in E_{s,dep}^{shunt} \qquad (18)$$

The set $E_{s,dep}^{shunt}$ consists of all outgoing departure events of rolling stock units that are short-turned at station σ and shunted during their turnaround. Constraints (18) identify whether this shunting process crosses the period border $\left(q_k^{shunt} = 1\right)$ or not. Hence, in station capacity constraints (10), we replace v_k with $v_k - t_{shunt} + Tq_k^{shunt}$.

Algorithm 1 Before TTAP

```
Given: original timetable $(\pi, p)$, possessions

procedure ($strat$; BEFORETTAP)
    $S_{ST} \leftarrow$ Short-turn trains
    for $s \in S_{ST}$ do
        Add turnaround activities at $s$
    end for
    Apply network aggregation
    $S_{violated} \leftarrow$ Check station capacity
    while $S_{violated} \neq \emptyset$ do
        for $s \in S_{violated}$ do
            Add station constraints
        end for
        $(v, q, x) \leftarrow$ Solve TTAP
        $S_{violated} \leftarrow$ Check station capacity
    end while
end procedure

Result: Alternative timetable $(v, q, x)$
```

Algorithm 2 After TTAP

```
Given: original timetable $(\pi, p)$, possessions

procedure ($strat$; AFTERTTAP)
    $S_{ST} \leftarrow$ Short-turn trains
    Apply network aggregation
    $S_{violated} \leftarrow$ Check station capacity
    while $S_{violated} \neq \emptyset$ do
        for $s \in S_{violated}$ do
            Add station constraints
        end for
        $(v, q, x) \leftarrow$ Solve TTAP
        for $s \in S_{ST}$ do
            Add turnaround activities at $s$
        end for
        $S_{violated} \leftarrow$ Check station capacity
    end while
end procedure

Result: Alternative timetable $(v, q, x)$
```

5.2 Including Turnaround Strategy in the Framework

After describing train turnaround constraints for short-turned trains, we still need to decide where to add them in the model. We introduce two procedures to do so. First, turnaround constraints can be added before the first iteration of the optimization model. Second, they can be added after optimization iterations finished. We indicate to the former as ($strat$, beforeTTAP) and to the latter as ($strat$, afterTTAP), $strat$ being one of the presented strategies, FCFS and maxFCFS. The set S_{ST} contains all stations s at which at least one train is short-turned, and is a result of the preprocessing step.

Algorithm 1 formulates the ($strat$; beforeTTAP) procedure. Here, turnaround activities are added based on the arrival and departure times in the original timetable. Including turnaround activities before the optimization is run for the first time, results in the problem becoming more constrained: the turnaround time is fixed within the model, meaning that a delay for the arrival of the incoming train always causes the same delay for the departure of the outgoing train. The advantage is that station capacity is already taken into account in the first iteration, reducing the probability of a second iteration.

The ($strat$; afterTTAP) procedures add turnaround activities only after each iteration of the optimization model, see Algorithm 2. Hence, it does not link the delays of the respective connected events on beforehand. Although this allows more flexibility, minimum turnaround time requirements may result in loosing a possible outgoing train that is not delayed, in case the arrival of the incoming train is delayed too much. The opposite may also happen if the departure of an outgoing one is delayed. After adding these turnaround activities, station capacity has to be checked again, as turnarounds in general take more time at the station than regular dwell activities. As a result, the main disadvantage is the possible need for a second iteration.

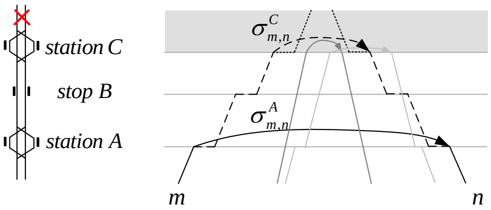


Figure 4: Illustration for the flexible short-turning concept using two train lines m and n. The segment above station C is closed due to an open-track possession. Variables $\sigma^A_{m,n}$ and $\sigma^C_{m,n} \in \{0,1\}$ indicate whether train m short-turns and connects to n at short-turning location A or C respectively. Short-turning at station A ($\sigma^A_{m,n} = 1$) may avoid conflicts with the other trains (in grey) at station C, which has two platforms. This decision cancels all events and processes indicated by dashed lines.

5.3 Flexible Short-Turning

We introduce flexible short-turning possibilities for the optimization model. More specifically, we define a set of candidate short-turning locations for a pair of lines and let the optimization model decide which one to use.

Figure 4 shows a corridor with two stations, A and C, and a stop B, which does not allow short-turning. In case of a fixed short-turning strategy, an open-track possession north of station C forces train short-turning at this location for all trains running on the corridor. However, the three resulting turnaround activities lead to station capacity violation, as only two platform tracks are available at C. The optimization model may delay and/or cancel train lines. Providing a second short-turning possibility at station A for at least one pair, may avoid the need for train cancellation, but cancels some other events and processes. We illustrate the concept and how it affects existing process and station capacity constraints by focussing on the black train lines in Figure 4.

We adapt the preprocessing module to provide a number of short-turning locations $\big(S_{ST,(m,n)}\big)$ for a pair of train lines $(m,n) \in M \times M$. In our example, these are the stations A and C. Binary variables $\sigma^s_{m,n}$, where $\sigma^s_{m,n} \in S_{ST,(m,n)}$, indicate that train lines m and n short-turn at a station s. Different events and activities are cancelled depending on which short-turning location is chosen. Regardless of the choice, events and dwell activities next to the open-track possession (indicated by dotted lines in Figure 4) have to be cancelled. Choosing station C as the short-turning station ($\sigma^C_{m,n} = 1$), adds a turnaround activity at that location. Short-turning trains at station A ($\sigma^A_{m,n} = 1$) results in trains not running between this station and station C, thereby cancelling the departure from A, and arrivals and departures at B and C of train line m. In the other direction, departures from C and B, and arrivals at B and A of train line n are affected. All activities associated with these events have to be cancelled as well, a new turnaround activity at station A is added. Preprocessing identifies for each event of lines m and n by which σ and x variables it is cancelled, resulting in a cancellation term γ_i for each event $i \in E$. For example, events at stop B of train line m are cancelled by either train line cancellation, or short-turning at A, hence $\gamma_i = \sigma^A_{m,n} + x_m$.

The optimization model can be adjusted by replacing cancellation variables x_m in event constraints (4) and (5), activity constraints (6) and station capacity constraints (10), with

Table 1: Lower and upper bounds for station capacity constraints (21) in case of flexible short-turning at station s, depend on the nature of events i and j, of train lines m and n respectively. The set $E^s_{arr,ST}$ consists of all arrival events associated with possible short-turning variables. Sets X and Y can be any other event type as in Van Aken et al. (2017).

Set i	Set j	Lower bound LB	Upper bound UB
$E^s_{arr,ST}$	Y	$(\sigma^s_{m,n} - c_{ij} - \gamma_j)\tau_{ik}$	$T - 1 - \delta_j(\sigma^s_{m,n} - c_{ji} - \gamma_j)$
X	$E^s_{arr,ST}$	$(\sigma^s_{m,n} - c_{ij} - \gamma_i)\delta_i$	$T - 1 - \tau_{jl}(\sigma^s_{m,n} - c_{ji} - \gamma_i)$
$E^s_{arr,ST}$	$E^\sigma_{arr,ST}$	$(\sigma^s_{m,n} - 1 - c_{ij})\tau_{ik}$	$T - 1 - \tau_{jl}(\sigma^s_{m,n} - 1 - c_{ji})$

the event-specific cancellation terms γ_i and γ_j for events i and j respectively. Additional constraints to be added, include:

$$\sum_{s \in S_{ST,(m,n)}} \sigma^s_{m,n} + x_m = 1 \qquad \forall m \in M \qquad (19)$$

$$\tau_{ik}\ \sigma^s_{m,n} \leq v_k - v_i + q_{ik}T \leq \tau_{ik} + (T-1)(1 - \sigma^s_{m,n}) \qquad \forall s \in S_{ST,(m,n)} \qquad (20)$$

Constraints (19) ensure that the model selects one short-turning location, or cancels train line m. Let i be the arrival event of train m at station s, and k the departure of train n in the other direction, resulting in a turnaround time $\tau_{ik} = v_k - v_i + Tq_{ik}$. Constraints (20) are similar to constraints (17), enforcing the turnaround time if s is chosen as short-turning location, i.e., $s^\sigma_{m,n} = 1$. Otherwise, lower and upper bounds are relaxed.

Some of the station capacity constraints have to be adjusted too. Let $E^s_{arr,ST}$ denote the set of arrival events for short-turned trains at station s, X and Y can be any other event sets as in Van Aken et al. (2017). Each event $i \in E^\sigma_{arr,ST}$ is associated with a short-turning variable $\sigma^s_{m,n}$, and a turnaround time $\delta_i = \tau_{ik}$. However, station capacity constraints only come into play if trains m and n are being short-turned at s. Hence, next to the station capacity constraints already defined by constraints (10), the following ones are added, with LB and UB depending on the sets events i and j belong to (see Table 1):

$$LB \leq v_j - v_i + Tq_{ij} \leq UB \qquad (21)$$

Table 1 shows how lower and upper bounds depend on the nature of events i and j. The basic idea is that if at least one of both events i or j is an arrival linked to a short-turning possibility, these constraints are relaxed in case trains do not short-turn here ($s^m_\tau = 0$), or one of both events is cancelled. Sets X and Y and associated values δ_i and δ_j are described in Van Aken et al. (2017). As an example, consider the first row of Table 1, with train m short-turning at s ($\sigma^s_{m,n} = 1$), and event j not cancelled ($\gamma_j = 0$). The lower bound reduces to $(1 - c_{ij})\tau_{ik}$, and train n can only arrive during the turnaround activity of train m if $c_{ij} = 1$.

Finally, decision variables $\sigma^s_{m,n}$ have to be incorporated in the objective function (12). As choosing an earlier short-turning, e.g., at station A instead of C in Figure 4, partially cancels train lines m and n, the penalty represents the sum of the running times of cancelled activities relative to the train lines' *journey times*. The latter is defined as "time elapsed between the first departure and the final arrival of a train line".

6 Computational Experiments

The proposed model for solving TTAP was tested on four cases on the national Dutch train network. The first three cases were used to quantify effects of different levels of network aggregation and turnaround procedures, while the fourth one was used to evaluate the performance of the model by comparing results with the ones generated by planners. The former were manually created scenarios, while the latter represents a real-life case. Data on event times and processes were taken from DONS and represented a weekend timetable, as most major possessions are scheduled during weekends. The timetable included a total of 288 train paths. This resulted in 10,926 events, 50,210 processes, and 564 timetabling points before network aggregation. Each of three created cases consisted of 20 possessions in total, with an equal ratio for open-track and station track possessions. Open-track possessions were generated by randomly selecting segments of the network. For station track possessions, at least one was scheduled in a "big" station, i.e., within the top 10% concerning the number of train arrivals. The other nine were selected randomly, ensuring that station capacity became insufficient for at least half of them. The fourth, real-life case included 22 open-track possessions and 4 station track possessions.

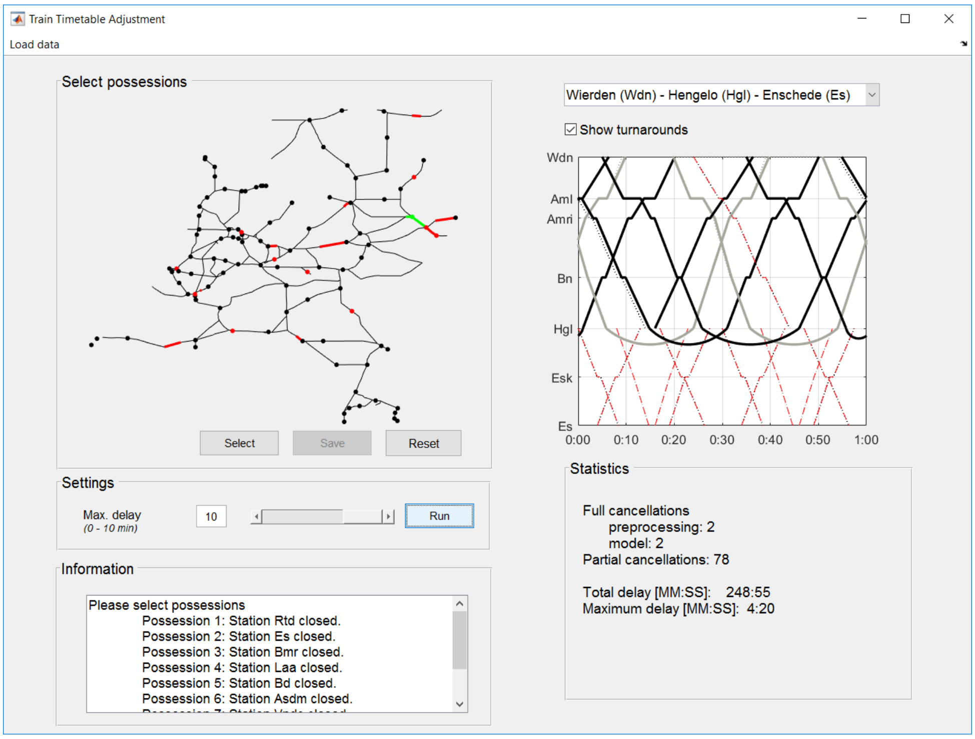
Figure 5 shows a prototype tool, which planners can use to solve instances of the complete Dutch network. The interactive network plot on the left allows selecting open-track and station track possessions at any location on the network, and visualizes them in red. Planners can also define a maximum delay d^+_{max}, and potentially other model parameters. After running the TTAP model, they can select a certain corridor for which they want to access the resulting time-distance diagram. This corridor is also highlighted in green on the network plot. Finally, the bottom right part reports the number of fully and partially cancelled train lines, together with the total and maximum delay.

The model was developed in Matlab and solved using Gurobi 6.4.1. Experiments were run using an Intel core i7-5500U (2.4 GHz) processor and 8 GB RAM. Initial experiments showed that most computation time was consumed after reaching a 0.1% optimality gap; hence, problems were solved up to a 0.1% optimality gap. Weights $w^{cancel} = 10^6$ and $w^{delay} = 1$ were applied. Maximum delay d^+_{max} was set to 600 s. For maxFCFS strategies, $\tau_{max} = 600\ s$ and $t_{shunt} = 120\ s$ were used, based on planners' practice.

Section 6.1 compares different levels of network aggregation, based on the three techniques described in Section 4. Section 6.2 presents results for four possible procedures described in Section 5. Results are compared in terms of solution quality, i.e., total delay and number of cancellations, and speed, i.e., number of iterations and computation time. Section 6.3 evaluates the solutions of our model with the ones obtained by NS planners and depicts the main differences.

6.1 Comparing Different Levels of Network Aggregation

We compared different levels of network aggregation for the (maxFCFS; afterTTAP) procedure, assuming shunting possibilities to be present at all stations. Each single technique was tested independently: removing departure-through events ("depthru"), removing 10% of timetabling points ("10% TT points"), and removing non-essential headway processes ("NEHWP"). Table 2 reports the size of the aggregated network in number of events, processes and timetabling points, as well as the total delay and computation time for each case. Due to criteria 2 and 3, network size may vary, and thus the average number of events and



Max. delay *(0 - 10 min)* 10

Please select possessions
Possession 1: Station Rtd closed.
Possession 2: Station Es closed.
Possession 3: Station Bmr closed.
Possession 4: Station Laa closed.
Possession 5: Station Bd closed.
Possession 6: Station Asdm closed.

Full cancellations
preprocessing: 2
model: 2
Partial cancellations: 78

Total delay [MM:SS]: 248:55
Maximum delay [MM:SS]: 4:20

Figure 5: Prototype tool for solving TTAP, showing the result for the Wierden (Wdn) - Enschede (Es) corridor obtained for the third considered case. In the time-distance diagram on the right, red dashed lines represent cancelled lines. Train paths of intercity (IC) trains are plotted in grey for both the original (dotted) and alternative timetable (full); black lines represent regional (R) trains. Blue dotted lines represent trains paths in the original timetable.

processes remaining after network are reported.

Table 2 shows that "depthru" and "NEHWP" reduced the size of the graph significantly. The former decreased both number of events (-29%) and processes (-44%), which is due to merging two subsequent running processes and removing duplicate headway processes. Values reported for "NEHWP" show that a large number of headway processes never become violated with $d^+_{max} = 600\ s$. As expected, these techniques did not affect the final solution, which validates their implementation. The opposite was true when removing timetabling points: the reduction in network size was smaller, which was caused by removing only small timetabling points using the random number generator. Objective function values for all cases showed minor deviations from the original one, which indicates that the linear approximation works well.

Computation times showed mixed results. The "depthru" technique seemed to work well for cases 1 and 3, whereas case 2 did not show improvements. Despite the reduction in network size was rather small, removing timetabling points had a positive effect on computation time for all cases. Finally, "NEHWP" resulted in significantly lower computation times for cases 1 and 2, but not for case 3. Note that part of these observations may be at-

Table 2: Results for applying the three network aggregation techniques, reporting remaining network size, i.e., number of events, processes, and timetabling points, total delay and computation times for each case study. "Depthru", "10% TT points" and "NEHWP" refer to merging through events, removing stations, and removing non-essential headway processes.

Aggregation technique	Network size			Total delay [s]			Computation time [s]		
	# events	# processes	# TT points	1	2	3	1	2	3
None	10,926	48,940	564	5,315	3,006	2,227	918	247	54
Depthru	7,758	27,699	564	5,315	3,006	2,227	680	250	36
10% TT points	10,232	46,068	508	5,330	3,039	2,224	504	138	47
NEHWP	10,926	28,992	564	5,315	3,006	2,227	495	150	64

tributed to the MIP solver applying general problem reduction techniques. However, in general, our aggregation techniques showed significant improvements in computation times and relatively good correlation of objective functions with the initial (non-aggregated) model.

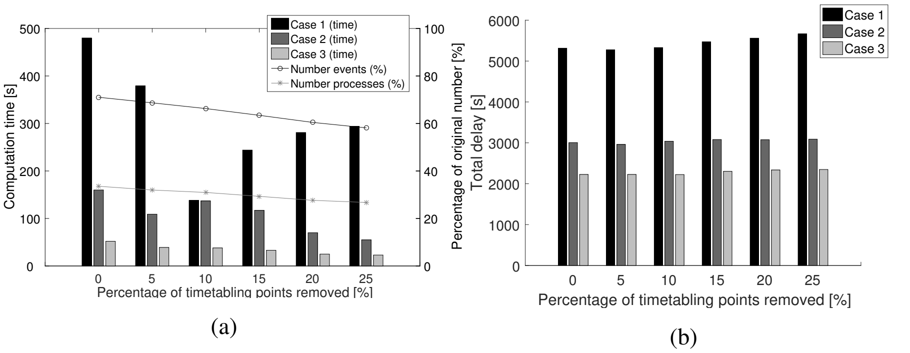
Next, we evaluated the effect of applying all three network aggregation techniques, with the number of timetabling points removed varying from 0 to 25%, with 5% increments. (Due to possession-independent criteria 1 and 4, this percentage is limited to 54%.) Figure 6a reports the total computation time for all cases, and the number of events and processes in the remaining network as a percentage of the original network size. Again, criteria 2 and 3 (Section 4.2) resulted in different timetabling points being removed for each case and we report the average sizes in terms of number of events and processes. When no timetabling points had been removed, the number of events and processes were lower than the original ones, as "depthru" and "NEHWP" were applied. Figure 6a shows that computation time decreased steadily for cases 2 and 3 for increasing number of timetabling points removed, with small variations for intermediate values. For the first case, the variation was even higher. These observations might follow from the random seeds used by the MIP solver. In future research we will address this by comparing average values over a large number of runs. However, we can conclude that removing timetabling points normally decreases computation times, whilst not changing the solution significantly.

Figure 6b shows that the total delay varied within acceptable margins from the original one. Hence, we conclude that the linear approximation for delays at removed stations and stops works well independently of the percentage of removed stations. This is attributed to under- and over-estimations balancing each other. The applied optimality gap of 0.1% may explain part of these deviations too.

6.2 Comparing Turnaround Procedures

We evaluated four possible procedures by combining the two strategies mentioned in Section 5.1 with the algorithms described in Section 5.2, using the same three cases. For network aggregation, a combination of all three techniques was applied, removing 20% of timetabling points. Figure 7a reports the total computation time and the number of iterations required for each case and procedure. Figure 7b displays the total delay and number of cancellations, if any, for each case and procedure.

Figure 7a shows that choosing a procedure strongly influenced computation times. For case 1, computation times were higher when adding turnaround activities after the first iteration. We attribute this to the increased freedom of the optimization model: event times



(a)

(b)

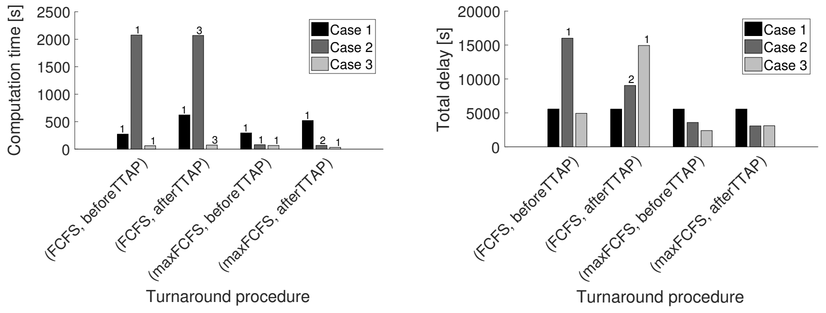
Figure 6: Results for removing an increasing percentage of timetabling points: (a) network size, i.e., number of events and processes, and computation time, and (b) total delay.

of short-turned trains are not yet connected by a turnaround activity and can be adjusted independently, i.e., the delay of the arrival does not influence the delay of the departure. For cases 2 and 3, the increase in running time resulted from the need for a second and even third iteration. In case of the maxFCFS strategy, computation times decreased significantly for case 2, as the model no longer had to decide on which train line to cancel.

Figure 7b shows that the total delay decreased considerably when we applied a maxFCFS strategy, which did not add turnaround activities for most of the short-turnings. As a result, turnaround activities do not compete for station capacity, which was already illustrated in Figure 3b. Case 1 did not require train cancellation for any of the procedures, and results show that total delay merely varied. Hence, the fact that events are not connected to each other for procedures ($strat$; afterTTAP), did not result in large benefits.

Due to the shorter station track occupation in the maxFCFS strategy, no train cancellations were needed. For the FCFS strategy, one or two train lines had to be cancelled for cases 2 and 3, depending on the procedure. For a (FCFS; beforeTTAP) procedure, the additional turnarounds are taken into account in the first iterations and influence retiming. Only including them afterwards with the (FCFS; afterTTAP) procedure, resulted in different turnarounds, of which one lasted about 1,800 s in case 3, resulting in station capacity violation. Retiming was not sufficient to solve the problem and trains had to be cancelled.

The possessions for the third case are highlighted in red on the network in Figure 5 and included, amongst others, an open-track possession between Hengelo (Hgl) and Enschede (Es), and a station track possession at Hgl, where only two out of six platform tracks remained in operation. Figures 8a and 8b show the time-distance diagrams for the Wierden (Wdn) - Enschede (Es) corridor (indicated in green on the network in Figure 5), respectively applying the (FCFS; beforeTTAP) and (FCFS; afterTTAP) procedures. The possessions of this case study are highlighted in red on the network in Figure 5 and included, amongst others, an open-track possession between Hengelo (Hgl) and Enschede (Es), and a station track possession at Hgl, where only two out of six platform tracks remained in operation. For the afterTTAP algorithm (Figure 8b), possessions elsewhere in the network caused a delay to an incoming R train in Hgl (dashed green circle). Hence, due to minimum turnaround time constraints, the short-turning procedure turned this train on the R train leaving Hgl just after 0:30. As a result, the incoming R train at 0:43 could not be assigned to a platform track any-



(a)

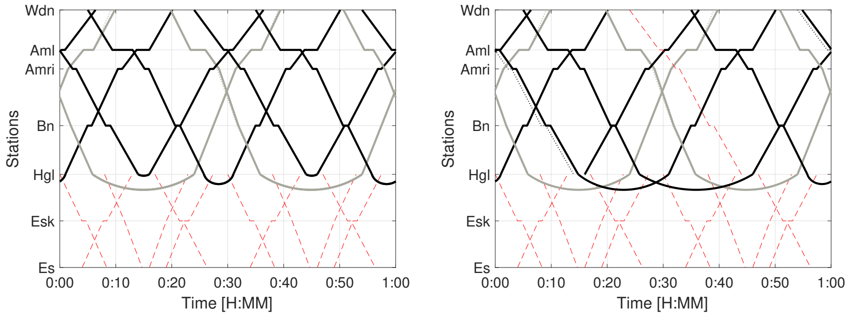
(b)

Figure 7: Results of applying four different procedures for adding turnaround activities for short-turned trains. (a) Computation time: bars indicate total computation time, the number of iterations is mentioned on top. (b) Solution quality: bars indicate total delay after the final iteration; in case trains are cancelled, their number is mentioned on top.

more (all were occupied) and had to be cancelled because of limited station capacity. Such a capacity shortage did not occur for the beforeTTAP algorithm (Figure 8a) as it resulted in a different set of turnaround activities.

We conclude that it is not straightforward to determine the optimal algorithm, i.e., beforeTTAP or afterTTAP. To avoid possible additional iterations, beforeTTAP seems to be a better choice. One can question the realism of the maxFCFS strategy, as in reality, shunting may not be possible at each short-turning location. This can be solved by a mixed strategy if a list of short-turning locations with shunting possibilities is available.

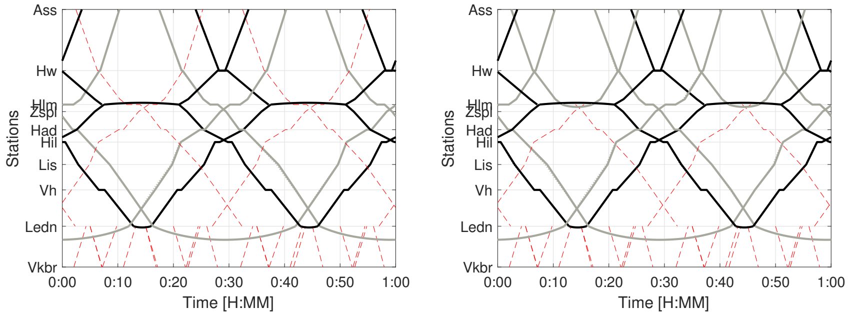
Choosing an earlier short-turning location in the flexible short-turning procedure (flexST) only becomes beneficial if it can avoid a train cancellation at the regular short-turning location. In this procedure, trains turn on the same train series in the other direction, which



(a)

(b)

Figure 8: Time-distance diagrams for the Wierden (Wdn) - Enschede (Es) corridor in case study 3, after applying (a) (FCFS; beforeTTAP) and (b) (FCFS; afterTTAP). Red dashed lines represent cancelled lines. Train paths of IC trains are plotted in grey for both the original (dotted) and alternative timetable (full); black lines represent R trains.



(a) (b)

Figure 9: Time-distance diagrams for a case study with an open-track possession between Leiden (Ledn) and a bridge (Vkbr), and a station track possession at Ledn, which left only one platform track in operation. (a) Result for a fixed short-turning location at Ledn. (b) Result for a flexST procedure with an alternative short-turning location at Haarlem (Hlm). Red dashed lines represent cancelled lines. Train paths of IC trains are plotted in grey for both the original (dotted) and alternative timetable (full); black lines represent R trains.

essentially differs from the FCFS and maxFCFS. For the latter, the only requirements are (1) same train type; and (2) satisfying minimal turnaround time. For these reasons, the flexST has been evaluated using a specifically designed case study to prove its functionality.

Our case study included an open-track possession directly south of Leiden (Ledn), accompanied by a station track possession which only allowed to use one out of six platform tracks at Ledn. Figures 9a and 9b show the time-distance diagrams of the corridor between Amsterdam Sloterdijk (Ass) and a bridge south of Ledn (Vkbr) for a fixed short-turning location for each train pair, and the presented flexST procedure, respectively. Figure 9b shows that fixing the short-turning location at Ledn led to the cancellation of four IC trains for their complete journey along the Ass-Ledn corridor, as station capacity at Ledn was insufficient to accommodate the required turnaround activities. In this case, both IC and R trains could also have been short-turned at either Haarlem (Hlm) or Ledn. Providing this additional flexibility to the optimization model, avoided cancellations of half of the IC trains between Ass and Hlm as shown in Figure 9b.

The provided example illustrates the applicability of the flexST procedure and proves its main benefits. Still, more experiments with this strategy are needed to assess the full potential of the flexST procedure.

6.3 Comparing the Model's Results with Planners' Practices

In close cooperation with planners at NS, a real-life case study was selected to assess the potential of the model as a decision support tool, by comparing the model's alternative timetable with the one generated by planners. For one weekend, two time windows of one hour with the highest number of possessions were selected. In real-life, open-track possessions are more common than station track possessions, with the latter being less severe than those of our artificial case studies. Hence, both problems were solved within 1 min, with train cancellations being fully attributed to preprocessing. Our results were strongly corre-

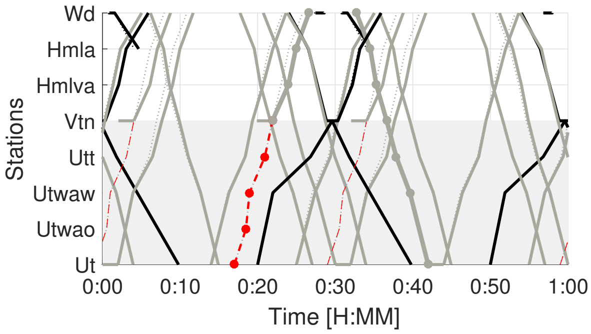


Figure 10: Time-distance diagram of the corridor Woerden (Wd) - Utrecht (Ut), including a possession between Ut and timetabling points Utwao and Utwaw. Full lines indicate the alternative timetable for IC (in grey) and R trains (in black), dashed lines represent cancelled lines. For trains of the IC 500 series, dots indicate events at timetabling points.

sponding with planners' results, but achieved much faster. Few differences were observed, for which possible future developments are identified.

First, planners may additionally connect two short-turned lines, by extending one of them to another station, to provide passengers with direct connections. Second, in areas with multiple timetabling points, e.g., a junction near a large station, trains may get rerouted to avoid open-track possessions occupying a small area, e.g., a single switch. As our model uses a timetable with fixed routes, this is not taken into consideration. Third, in large station areas, possessions may result in platform tracks to become unreachable for certain corridors. A macroscopic model does not take this into consideration, and needs a microscopic counterpart to check route feasibility, which may potentially also change the minimum headway between trains. Finally, extensions to the macroscopic model may be needed to include partial open-track possessions, e.g., only one track remains on a double-track segment. This could be done by imposing different headways for affected trains.

Figure 10 illustrates the second and third observations for the corridor between Woerden (Wd) and Utrecht Central Station (Ut), with an open-track possession between timetabling points Ut and Utwao, and Utwao and Utwaw, all located within the station area of Ut. As a result of this possession, the corridor towards Wd could not be reached from some of the platform tracks at Ut. Trains of series IC 500 arriving at Ut are not routed through the possessed area, trains in the other direction, whose events are indicated by the red dots, pass by point Utwao in the original timetable. Hence, the model short-turned the outgoing ones in Vleuten (Vtn). Planners' results included changes to the local route, i.e., outgoing trains did no longer pass by Utwao, thereby avoiding the possession. To achieve this, the platform track had to be changed, which cannot be accomplished by our macroscopic model.

Together with planners, two possible preliminary applications of our macroscopic model have been identified. First, alternative timetables generated by the current model can be used as a starting solution, which planners have to specify and adapt to fit constraints on the microscopic level. Second is situated in the possession scheduling stage, upon deciding on the days and time windows of possessions. The macroscopic model for TTAP can be used to quickly assess the effects of different combinations of possessions without going through the complete timetable adjusting process. At this step, the impact on train traffic can be quantified which will help in selecting combinations with the least impact on train operations and passenger satisfaction.

7 Conclusions

We extended the macroscopic model for the Train Timetable Adjustment Problem (TTAP) developed in Van Aken et al. (2017) to adjust a given timetable to a specified set of station track and complete open-track possessions by train retiming, reordering, short-turning and cancellation. The current model introduces more real-life constraints and tackles solving large-scale instances. For the latter, we presented three network aggregation techniques to reduce the problem size and enable the model to solve larger instances within short computation times, without affecting the final solution. Network aggregation allowed to solve four case studies on the complete Dutch railway network. We evaluated the effect of different levels of network aggregation on computation time, concluding that, in most cases, all techniques contribute to solving the network faster. Increasing the number of removed timetabling points does not affect the final solution and objective function value due to our linear approximation of delays.

To model station capacity consumption for short-turned trains, we presented four procedures for adding turnaround activities by applying two different strategies at two locations in the model. Strategies define which short-turned arrivals and departures are connected, and include two variants of a restricted FCFS. Turnaround activities can be added both before the TTAP model, or afterwards within the row generation step. We adjusted the station capacity constraints of the previous model accordingly. Adding turnaround activities before the TTAP model may avoid additional iterations. The maxFCFS strategy rendered good results, but may lack in realism, as may not all stations have shunting possibilities. Hence, we advice a mixed strategy. Additionally, we formulated a flexible short-turning procedure which gives the optimization model multiple options for short-turning.

We assessed the potential of the model for decision support, by comparing the model's results with the ones generated by planners. Our model generated solutions fast and planners were positive about the results. The current model can be applied to: first, provide planners with an initial solution which they have to further refine at the microscopic level and second, assess the impacts of combinations of possessions.

Future research could focus on three areas. First, a microscopic counterpart, which can find feasible routes in station areas, has to be developed. If no feasible routes can be found, feedback to the macroscopic model should be provided. Second, the macroscopic model could be extended to include partial open-track possessions. Further development of the flexible short-turning concept, including more experiments, may lead to better solutions with less cancellations. Finally, providing interactivity with planners is an alternative pathway to obtain more flexible solutions. We believe that with these developments, the TTAP model can be successfully used as a decision support tool to generate alternative timetables to provide more effective operations and significantly reduce the computation times.

Acknowledgements

This research was conducted in close cooperation with Netherlands Railways (NS), and we thank all people involved for their contribution. In particular, we thank Jasper Engel for providing the data and case study, and for carefully examining the results; and Gerben

Scheepmaker for his leadership and thorough support throughout the project.

References

Albrecht, A., Panton, D., Lee, D., 2013. "Rescheduling rail networks with maintenance disruptions using Problem Space Search", *Computers & Operations Research*, vol. 40, pp. 703-712.

Arenas, D., Pellegrini, P., Hanafi, S., Rodriguez, J., 2017. "Timetable Optimization during Railway Infrastructure Maintenance", In: *7th International Conference on Railway Operations Modelling and Analysis (RailLille2017)*, 4-7 April, Lille, France.

Budai-Balke, G., 2009. *Operations Research Models for Scheduling Railway Infrastructure Maintenance*, PhD Thesis, Erasmus University Rotterdam, the Netherlands.

Engel, J., 2016. Personal interview on February 29th, Utrecht, the Netherlands.

Forsgren, M., Aronsson, M., Gestrelius, S., 2013. "Maintaining tracks and traffic flow at the same time", *Journal of Rail Transport Planning & Management*, vol. 3, pp. 111-123.

Kidd, M.P., van der Hurk, E., Lusby, R.M., Pisinger, D., 2016. "Timetabling in an urban railway network with time-dependent frequencies", In: *The Ninth Triennial Symposium on Transportation Analysis (TRISTAN IX)*, 17-23 June, Aruba.

Kroon, L., Huisman, D., Abbink, E., Fioole, P., Fischetti, M., Maróti, G., Schrijver, A., Steenbeek, A., Ybema, R., 2009. "The New Dutch Timetable: The OR Revolution", *Interfaces*, vol. 39, pp. 6-17.

Lidén, T., 2015. "Railway Infrastructure Maintenance - A Survey of Planning Problems and Conducted Research", *Transportation Research Procedia*, vol. 10, pp. 574-583.

Lidén, T., Joborn, M., 2015. "Maintenance windows for railway infrastructure maintenance. An assessment and dimensioning model for the study of traffic impact and maintenance cost", In: *Conference on Advanced Systems in Public Transport (CASPT2015)*, 19-23 July, Rotterdam, the Netherlands.

Louwerse, I., Huisman, D., 2014. "Adjusting a railway timetable in case of partial or complete blockades", *European journal of operational research*, vol. 235, pp. 583-593.

Luan, X., Miao, J., Meng, L., Corman, F., Lodewijks, G., 2017. "Integrated optimization on train scheduling and preventive maintenance time slots planning", *Transportation Research Part C: Emerging Technologies*, vol. 80, pp. 329-359.

Nachtigall, K., 1996. "Periodic Network Optimization with Different Arc Frequencies", Discrete Applied Mathematics, vol. 69, pp. 1–17.

Odijk, M. A. (1996). "A constraint generation algorithm for the construction of periodic railway timetables". *Transportation Research Part B: Methodological*, vol. 30, pp. 455-464.

Peeters, L., 2003. *Cyclic railway timetable optimization.*, PhD Thesis, Erasmus University Rotterdam, the Netherlands.

Polinder, G., 2015. *Resolving infeasibilities in the PESP model of the Dutch railway timetabling problem*, MSc Thesis, Delft University of Technology, the Netherlands.

RailNetEurope, 2015. *Glossary of Terms Related to Network Statements* [online], RailNet-Europe, `http://www.rne.eu/ns_glossary` [Accessed 11 Mar. 2016].

Serafini, P., Ukovich, W., 1989. "A Mathematical Model for Periodic Scheduling Problems", *SIAM J. Discrete Math.*, vol. 2, pp. 550-581.

Van Aken, S., Bešinović N., Goverde R.M.P., 2017. "Designing alternative railway timetables under infrastructure maintenance possessions", *Transportation Research Part B:*

Methodological, vol. 98, pp. 224-238.

Vansteenwegen, P., Dewilde, T., Burggraeve, S., Cattrysse, D., 2016. "An iterative approach for reducing the impact of infrastructure maintenance on the performance of railway systems", *European Journal of Operational Research*, vol. 252, pp. 39-53.