CASPT 2015

Advanced methods for improving passenger robustness in railway systems

Sofie Burggraeve · Sofie Van Thielen · Pieter Vansteenwegen

Abstract In nearly saturated station areas the limited capacity is one of the main reasons of delay propagation. Spreading the trains well in these areas has a big impact on the total travel time in practice of all passengers in the railway network in case of frequently occurring small delays. We focus on improving the performance in the bottleneck of the network in order to improve the performance of the whole railway network. This document proposes a method that builds from scratch a routing plan and a cyclic timetable that optimally spread the trains in space and time. An integer route choice model assigns, without considering a timetable, to every train a route such that the maximal switch usage is minimized and that the number of times that each switch is used is quadratically penalized. Thereafter, a mixed integer linear timetable determination model assigns to each train the times at which this train passes through the switches on the route that was assigned by the route choice model. Different from other approaches is that we first fix the routing and thereafter construct the timetable. Compared to a reference routing plan for Brussel's railway station area, the switch usage improved with 7,6%.

Keywords Passenger robustness · Railway routing · Railway timetabling · Recurring delays · MILP

1 Introduction

The focus of this research is on making a timetable and a routing plan from scratch to transport passengers optimally in, out and through a busy railway station network. We assume that this timetable and routing plan can be made feasible outside this bottleneck without many changes since much less

KU Leuven, Leuven Mobility Research Centre - CIB
Celestijnenlaan 300 BOX 2422, 3001 Leuven
Tel.: +32-16-372023
Fax: +32-16-322483
E-mail: sofie.burggraeve@kuleuven.be

constraints are present there. Passengers want both short and reliable travel times. Hence, our goal is to optimize the *passenger robustness*, which means the total travel time of all passengers in practice in case of frequently occurring small delays (Dewilde et al., 2011). Unfortunately, direct implementation of this goal function is computationally highly demanding, as real travel times of all passengers and propagation of delays then have to be calculated. That is why we indirectly strive for passenger robustness by looking for an optimal spreading of the trains in time and space. We restrict our research to timetabling and routing, which are situated on the tactical level of railway planning. Although, for example, also network design, on the structural level, and real-time interventions, on the operational level, have an impact on the travel times of railway passengers in practice and thus on the passenger robustness of the railway system. The timetable and routing plan construction are only designed to mitigate the effect of frequently occurring small delays on the passenger travel times. The impact of large disturbances is not considered during the construction of the timetable or routing plan.

First, we describe how our method is related to the state of the art in timetabling and routing for nearly saturated railway station areas. We also point out the novelties and differences of our approach. Secondly, our methodology for routing and timetabling is explained in detail. The optimization models are described and illustrated on a small case study. Thereafter, the performance of the presented method is illustrated on the railway station area of Brussels (Belgium).

2 Literature

The construction and optimization of a routing plan and a timetable are closely interwoven. An optimal timetable can make routing infeasible and a routing plan can have no feasible timetabling. Research on timetables and routing plans can be divided into two classes. A first class of methods starts from an existing timetable or routing plan and tries to improve these by introducing small and smart modifications. The advantage is that these changes can have a relatively large impact on the performance of the railway system, for example on passenger robustness (Burggraeve et al., 2015; Dewilde et al., 2013, 2014). Moreover, these changes can easily be implemented without bothering the passengers too much. A second class of methods generates a routing plan and timetable from scratch. This class has the advantage that one doesn't need an existing timetable and/or routing plan to start from. Furthermore, one can thoroughly work out an optimal solution without being constrained to small changes and without any bias towards solutions that are similar to the initial timetable and routing plan. In this second class, there are again different branches in ongoing research. In a first branch, timetables are designed on a macroscopic level of the infrastructure and afterwards, on the microscopic level, the best routing throughout the stations is determined (Zwaneveld et al., 1996, 2001). If the routing problem is infeasible on the microscopic level, then the timetable has to be adapted. This approach works top-down. In a second branch, timetables are designed in a bottom-up approach. The first step is to aggregate the microscopic infrastructure (to a macroscopic level). This is done in such a way that it is possible to calculate timetable and time slot allocations with linear and integer programming in a reasonable amount of time and such that transforming the timetable and time slot allocations back to the microscopic level does not create conflicts (Borndörfer et al., 2011).

Our research belongs to the second class of methods. We focus on scheduling railway traffic in the bottleneck of a network in order to improve the performance of the whole railway system (Goldratt, 1986). Such a bottleneck often contains many switches, for example a railway station area. That is why we, in contrast to the bottom-up and the top-down approach, directly look at the microscopic infrastructure level for the construction of an optimal routing plan and timetable. Because we work on the microscopic, and thus detailed infrastructure level, computation time increases quickly with the size of the network and the number of trains on the network. The advantage of this approach is that the routing plan and the timetable are optimally designed for the bottleneck.

Routing problem Research on routing trains through railway station areas (almost) always starts from a timetable, meaning that arrival and departure times in the network and in the stations are known. The routing problem then consists of finding the assignment of routes to the trains that optimizes a given objective function while adhering the timetable. Examples of optimization criteria are maximizing buffer times between trains, minimizing travel times of trains or passengers, minimizing switch usage, maximizing spreading, etc. In our approach, we solve the routing problem before and thus independent of the timetabling problem in order to optimize the switch usage in the bottleneck. Our routing problem can be described as follows. A set of trains has to be routed through a railway station area, which is characterized by many parallel tracks and switches. For each of these trains their origin and destination in the considered network are known. There is no timetable to start from. Different optimization criteria can be used, for example homogenizing train traffic on tracks, optimizing the spreading of the trains in space without making detours, etc. Our focus is on the usage of switches in our network. Remark that the number of times that a switch is used determines the maximal buffer time in that switch and thus its vulnerability for propagating delays. For example, when 10 trains pass a switch in one hour, the optimal spreading of these trains in this switch leads to a train passage every 6 minutes. When only 5 trains pass that switch in an hour, the best possible spreading is only one train in every 12 minutes. That's why we minimize the maximum usage of a switch and simultaneously, we penalize the number of times that each switch is used quadratically.

Timetabling problem The timetabling problem consists of the assignment of time instants to the trains that have to be planned, for example arrival and departure times at stations or reservation and release times of sections of the network. Most models that solve the (cyclic) timetabling problem from scratch are based on the periodic event scheduling problem (PESP) (Serafini and Ukovich, 1989; Cacchiani et al., 2012; Liebchen et al., 2007). In the PESP, arrival and departures times of trains in stations are events. Related events are linked to each other by constraints that put an upper bound and/or lower bound on the time duration between these events. Related events are for example arrival and departure times of the same train, arrival and departure times of trains that provide a transfer, arrival and departure times of trains that make use of the same platform, etc. In our approach not only arrival and departure times in stations are taken into account, but also the passage times of the trains in all switches on their route. We are able to do this, because we already assigned a route to each train by solving the route choice model. The passage times of a train in the switches on its route are related to each other by constraints in the optimization model. The times between different trains on a common switch are not constrained explicitly, but these times are maximized in the objective function of the optimization model.

3 Methodology

We construct a cyclic timetable and routing plan from scratch. We start from a network and a set of trains that has to be routed through this network. For each train, the origin and destination in the network are fixed. The first step is to determine the set of all routes that link the origin of a train with its destination and to build the binary matrix L which contains the information whether or not a switch is part of a certain route. In this section, we first explain the route choice model and secondly the timetable determination model. For each model, the decision variables and parameters, the goal function and the constraints are presented.

3.1 Route choice model

We want to assign routes to the trains such that every switch and platform is used as little as possible with the underlying goal to spread the trains optimally in space. We focus on switches and platforms because they uniquely determine the route of a train through the network. Moreover, two trains can only be in conflict if they share at least one switch. To simplify the formulation, we will only mention switches in the rest of the paper, while in fact meaning switches and platforms. We make no assumptions about the timetable. The timetable will only be determined after the routing plan. In order to achieve that every switch is used as little as possible, we combine two aspects. First, we explicitly minimize the maximum use of a switch. We say that a switch is used x times if there are x trains whose route contains this switch. Thereafter, we minimize the sum of the squares of the usages of all switches and we put the maximum usage, found in the first problem, as a constraint on the usage of each switch. This second optimization problem gives the incentive to further decrease the individual switch usages. We integrate these two minimization problems into one problem by giving the minimization of the maximum number of usages of a switch a much higher weight (HW) and leaving out the constraint on the maximum usage of each switch. The magnitude of HW depends on the problem size. We minimize the sum of the squares of the usages, instead of for example the sum of the usages, to penalize an increase in a switch utilization rate harder the higher the utilization rate of the switch already is. In fact, we minimize the usage of every switch in order to allow for an optimal spreading in time during the timetable determination phase. We now present and explain the optimization model.

Parameters
$T = \{t_1, t_2, \cdots, t_n\}$ = set of trains with n the number of trains.
$W = \{w_1, w_2, \cdots, w_k\}$ = set of switches with k the number of switches.
$R = \{r_1, r_2, \cdots, r_p\}$ = set of routes with p the number of routes.
R_t = set of routes that train t can take (based on its origin and destination in the network).
$L = (l_{r,w})_{r \in R, w \in W}$ with $l_{r,w} = 1$ if route $r \in R$ contains switch $w \in W$ and 0 otherwise.
HW = the high weight used to enforce the domination of the minimax criterion in the objective function.

Decision variables
$g_w \in \mathbb{N}$ = number of times switch w is used, $w \in W$.
$x_{t,r} = 1$ if route $r \in R_t$ is assigned to train $t \in T$ and 0 otherwise.

Model
The minimization of the maximum usage of a switch can be represented by:

$$\min \max_{w \in W} g_w. \tag{1}$$

The minimization of the sum of the squared usage of each switch can be represented by

$$\min \sum_{w \in W} g_w^2. \tag{2}$$

The combined weighted goal function can be represented by

$$\min \qquad HW \cdot \max_{w \in W} g_w + \sum_{w \in W} g_w^2. \tag{3}$$

Remark that both terms of this objective function are non-linear. We now explain how we linearize this objective function. For the first term, we introduce a new decision variable z that is bounded below by the g_w's in constraints (6). These constraints assure that z is always larger than the maximum of the g_w's. By minimizing $HW \cdot z$, see (5), the maximum of the g_w's is minimized and the high weight assures that z and thus the maximum use is as low as possible. To linearize the second term, we use the following trick. We introduce $K \cdot |W|$ binary decision variables $b_{k,w}$, with $k \in \{1, \cdots, K\}, w \in W$ and with K an upper bound on the maximum usage of a switch. If switch w is used at least k times ($g_w \geq k$), then $b_{k,w} = 1$, else $b_{k,w} = 0$. The values of the $c_{k,w}$'s in Table 1 now assure the following equality for every switch w:

$$\sum_{k=1}^{K} c_{k,w} b_{k,w} = \sum_{k=1}^{g_w} c_{k,w} b_{k,w} = g_w^2, \tag{4}$$

where the first equality is true by definition of the $b_{k,w}$'s. The (first) summation is a linear combination of decision variables and thus suitable to replace $\sum_{w \in W} g_w^2$ in the goal function. We also add constraints (7). These constraints enforce that $b_{k,w} = 1$, if $g_w \geq k$. If $g_w < k$, then $b_{k,w}$ will be zero, because of its contribution in the goal function of the minimization problem. We use K as coefficient for $b_{k,w}$, as we defined it as upper bound on the maximal usage of a switch and thus on g_w. Remark that the number of trains is a natural upper bound on the maximal usage of a switch as a train can pass a switch at most once. Thus, $K \leq |T|$.

Table 1 Linearization of the quadratic terms

k	$c_{k,w}$	$\sum_{l=1}^{k} c_{k,w}$
1	1	1
2	3	4
3	5	9
4	7	16
$\vdots$	$\vdots$	$\vdots$
K	$2K-1$	K^2

$$\min \quad HW \cdot z + \textstyle\sum_{k=1}^{K} c_{k,w} b_{k,w} \tag{5}$$

$$\text{s.t.} \quad g_w \leq z \qquad \forall w \in W \tag{6}$$

$$g_w - K \cdot b_{k,w} \leq k - 1 \qquad \forall k \in \{1, \cdots, K\}, \forall w \in W \tag{7}$$

$$\textstyle\sum_{r \in R_t} x_{t,r} = 1 \qquad \forall t \in T \tag{8}$$

$$\textstyle\sum_{t \in T} \sum_{r \in R_t} l_{r,w} x_{t,r} = g_w \qquad \forall w \in W \tag{9}$$

$$g_w \in \mathbb{N} \qquad \forall w \in W \tag{10}$$

$$z \in \mathbb{N} \tag{11}$$

$$x_{t,r} \in \{0, 1\} \qquad \forall t \in T, \forall r \in R_t \tag{12}$$

Constraints (8) assure that each train is assigned exactly one route (that contains the origin and destination of that train in the network). Constraints (9) make sure that the number of assigned routes that use switch w equals g_w. Constraints (10) and (12) assure that the number of times that a switch is used is an integer and that the assignment of a route to a train is represented by the binary variable $x_{t,r}$. As the g_w's are integers, z is also an integer, which is represented in constraint (11).

Remark If a train has more possible origin and/or destination points in the network, this can easily be integrated in the model by adding the new possible routes for this train to his set R_t.

3.2 Timetable determination model

Objective
The goal of the timetable determination model is to spread the trains optimally in time in each switch. We start from the output of the route choice model, so every train has already been assigned exactly one route. Furthermore, as the distances between two points in the network and the speed limits for trains and links are known, we can easily find the time that a train needs between any pair of switches on its route in ideal circumstances (no disturbances). The approach also works if there are supplements included in the train travel times or even waiting times, as long as the travel times are fixed and known beforehand. So, if we fix the start time of each train in the network, this uniquely determines the position of the trains at every moment in time. The start times (and thus all passage times at switches) will be optimized by a mixed integer linear program that maximizes the time between every two passages in every switch.

To optimally spread the trains, we again focus on two aspects. The approach is similar to that of the route choice model. Primary, we maximize the minimal buffer time in a switch, i.e. the time between two consecutive passages in that switch. Secondary, we maximize the sum of the minimal buffer times between every two trains that have at least one switch in common. This second optimization goal gives the incentive to further increase the buffer times between train pairs. Thus here, maximizing the minimal buffer time over all train pairs gets a high weight in the objective function (HW') and this weight depends again on the problem size.

We construct a cyclic hourly timetable, so we suppose that a certain trip repeats itself every hour. This is what we call a series of a train. It is not the same physical train that performs this trip every hour, but we represent each train of a certain series with the same symbol. The cyclicity then induces two time buffers between the passages of a train pair. Let A and B represent two trains from different series which trips have at least one switch in common. For example, suppose that train A passes a certain switch at 4 minutes past the hour and train B passes that switch at 50 minutes past the hour. Then we have 46 minutes between A and B and 14 minutes between B and A. In our optimization model, we want to work with the shortest buffer time between two trains, which is the buffer of 14 minutes in the example. The shortest buffer time between two trains in a switch can be found as the minimum of the difference of their passage times in that switch and the absolute value of 60 minutes more or less than this difference. In the example, the shortest buffer time is 14 minutes and this is indeed the minimum of $\{|4-50|, |4-50-60|, |4-50+60|\} = \{46, 106, 14\}$. Remark that the shortest buffer time will always be contained in the interval $[0, 30]$.

Parameters
$S = \{s_1, s_2, \cdots, s_q\}$ = set of places/switches where trains can enter the network.
s_t = the origin (start point) of train t in the network.
r_t = the route that was assigned to train $t \in T$ in the route choice model. This route consists of switches (nodes) and links.
$d_{t,s,w}$ = the time that train $t \in T$ needs in ideal circumstances to go from s, where its enters the network, to switch $w \in r_t$ (on its route).
T_w = the set of trains for which r_t contains switch w.

Decision variables
$x_{t,s} \in [0, 60]$ = the moment at which train $t \in T$ enters the network in point $s_t \in S$.
$x_{t,w} \in [0, 60]$ = the moment at which train $t \in T$ arrives at switch $w \in W$ in the interval $[0, 60]$.
$x^{aux}_{t,w} \in [0, 120]^{(i)}$ = the moment at which train $t \in T$ arrives at switch $w \in W$ by adding $d_{t,s,w}$ to $x_{t,s}$, i.e. $x^{aux}_{t,w} = x_{t,s} + d_{t,s,w}$. It's not sure that this time point is in $[0, 60]$.
$x^{bin}_{t,w} \in \{0, 1\} = 0$ if $x^{aux}_{t,w} \in]60, 120]$ and 1 if $x^{aux}_{t,w} \in [0, 60[$, if $x^{aux}_{t,w} = 60$, then $x^{bin}_{t,w}$ can be either 0 or 1. This variable indicates whether the passage time of train t in switch w, based on the start time of train t in the network and the time that train t needs to go from switch s to switch w, belongs to the one hour interval $[0, 60]$ or not, i.e. $x^{aux}_{t,w} = x^{t,s} + d_{t,s,w} \leq 60$ or $x^{aux}_{t,w} = x^{t,s} + d_{t,s,w} \geq 60$
$y_{t_i,t_j,w} \in [0, 30]$ = the shortest buffer time between train t_i and t_j in switch w.
$y^{aux}_{t_i,t_j,w} \in [0, 60]^{(ii)}$ = a buffer time between train t_i and train t_j in switch w, but not necessarily the shortest buffer time.
$y^{bin}_{t_i,t_j,w} \in \{0, 1\} = 0$ if $y^{aux}_{t_i,t_j,w} \in]30, 60]$ and 1 if $y^{aux}_{t_i,t_j,w} \in [0, 30[$, if $y^{aux}_{t_i,t_j,w} = 30$, then $y^{bin}_{t_i,t_j,w}$ can be either 0 or 1. This variable indicates whether the buffer time $y^{aux}_{t_i,t_j,w}$ is the shortest buffer time or not.

(i) auxiliary variable for $x_{t,w}$
(ii) auxiliary variable for $y_{t_i,t_j,w}$

$v_{t_i,t_j,w} \in \{0,1\} = 1$ if $x_{t_i,w} \leq x_{t_j,w}$ and 0 otherwise. This variable is an indicator for the sign of $x_{t_i,w} - x_{t_j,w}$.

Model
The maximization of the minimum buffer time in a switch can be represented by:

$$\max_{\substack{x_{t,w}\in[0,60]\\ \forall t\in T, w\in W}} \min\{|x_{t_i,w} - x_{t_j,w} + h\cdot 60||h \in \{-1,0,1\}, w \in W, t_i, t_j \in T_w\}. \quad (13)$$

A buffer time between two trains in a switch is a positive number, that is why the objective function contains absolute values. As explained above, the shortest buffer time between two trains in a switch can be found as the minimum of the difference of their passage times in that switch and the absolute value of 60 minutes more or less than this difference. This is here captured by $|x_{w,t_i} - x_{w,t_j} + h\cdot 60|$ where the parameter h can be -1, 0 or 1. For fixed trains t_i and t_j in T and a switch $w \in W$, the shortest buffer time between these trains in that switch is assigned to the variable $y_{t_i,t_j,w}$. Thus, we can rewrite the goal function as:

$$\max_{\substack{x_{t,w}\in[0,60]\\ \forall t\in T, w\in W}} \min_{\substack{w\in W,\\ t_i,t_j\in T_w}} y_{t_i,t_j,w}. \quad (13')$$

The maximization of the sum of the minimum buffer times between every two trains can be represented by:

$$\max_{\substack{x_{t,w}\in[0,60]\\ \forall t\in T, w\in W}} \sum_{t_i,t_j\in T} \min_{w\in r_{t_i}\cap r_{t_j}} y_{t_i,t_j,w}. \quad (14)$$

The combined weighted goal function is:

$$\max_{\substack{x_{t,w}\in[0,60]\\ \forall t\in T, w\in W}} \quad HW' \cdot \min_{\substack{w\in W,\\ t_i,t_j\in T_w}} y_{t_i,t_j,w} + \sum_{t_i,t_j\in T} \min_{w\in r_{t_i}\cap r_{t_j}} y_{t_i,t_j,w}. \quad (15)$$

However, it is obvious that this objective function is non-linear. So again we introduce auxiliary variables. We define the variables z_{t_i,t_j}, for all trains t_i, t_j that have at least one switch in common. We bound these variables above by $y_{t_i,t_j,w}$ for every switch $w \in W$ that their routes have in common. This is described in constraints (18). As a consequence, the maximization of the sum of the z_{t_i,t_j}'s implies the maximization of the sum of the minimal buffer times between every train pair. We also introduce a new decision variable z which we bound above by all the z_{t_i,t_j}'s. This is described in constraints (17). As a consequence, the maximization of z implies the maximization of the minimum of the $y_{t_i,t_j,w}$'s. The timetable determination model then becomes:

$$
\begin{aligned}
\max \quad & HW' \cdot z + \sum_{\substack{t_i,t_j \in T: \\ r_{t_i} \cap r_{t_j} \neq \emptyset}} z_{t_i,t_j} && && (16)\\
\text{s.t.} \quad & z - z_{t_i,t_j} \leq 0 && \forall t_i, t_j \in T_w : i < j && (17)\\
& z_{t_i,t_j} - y_{t_i,t_j,w} \leq 0 && \forall w \in W, \forall t_i, t_j \in T_w : i < j && (18)\\
& 0 \leq x_{t_i,w} - x_{t_j,w} + 60 v_{t_i,t_j,w} \leq 60 && \forall w \in W, \forall t_i, t_j \in T_w : i < j && (19)\\
& -x_{t_i,w} + x_{t_j,w} + y^{aux}_{t_i,t_j,w} - 60 v_{t_i,t_j,w} \leq 0 && \forall w \in W, \forall t_i, t_j \in T_w : i < j && (20)\\
& x_{t_i,w} - x_{t_j,w} + y^{aux}_{t_i,t_j,w} + 60 v_{t_i,t_j,w} \leq 60 && \forall w \in W, \forall t_i, t_j \in T_w : i < j && (21)\\
& 30 \leq y^{aux}_{t_i,t_j,w} + 30 y^{bin}_{t_i,t_j,w} \leq 60 && \forall w \in W, \forall t_i, t_j \in T_w : i < j && (22)\\
& y_{t_i,t_j,w} + y^{aux}_{t_i,t_j,w} - 30 y^{bin}_{t_i,t_j,w} \leq 60 && \forall w \in W, \forall t_i, t_j \in T_w : i < j && (23)\\
& y_{t_i,t_j,w} - y^{aux}_{t_i,t_j,w} + 60 y^{bin}_{t_i,t_j,w} \leq 60 && \forall w \in W, \forall t_i, t_j \in T_w : i < j && (24)\\
& x^{aux}_{t,w} - x_{t,s} = d_{t,s,w} && \forall t \in T, \forall w \in r_t \setminus \{s_t\} && (25)\\
& 60 \leq x^{aux}_{t,w} + 60 x^{bin}_{t,w} \leq 120 && \forall w \in W \setminus S, \forall t \in T_w && (26)\\
& x_{t,w} - x^{aux}_{t,w} - 60 x^{bin}_{t,w} \leq -60 && \forall w \in W \setminus S, \forall t \in T_w && (27)\\
& x_{t,w} - x^{aux}_{t,w} - 60 x^{bin}_{t,w} \geq -60 && \forall w \in W \setminus S, \forall t \in T_w && (28)\\
& 0 \leq x_{t,s} \leq 60 && \forall t \in T && \\
& z \in [0, 30] && && (29)\\
& z_{t_i,t_j} \in [0, 30] && \forall t_i, t_j \in T && (30)\\
& x_{t_i,w} \in [0, 60] && \forall w \in W, \forall t_i \in T_w && (31)\\
& x^{aux}_{t_i,w} \in [0, 120] && \forall w \in W \setminus S, \forall t_i \in T_w && (32)\\
& y_{t_i,t_j,w} \in [0, 30] && \forall w \in W, \forall t_i, t_j \in T_w : i < j && (33)\\
& y^{aux}_{t_i,t_j,w} \in [0, 60] && \forall w \in W, \forall t_i, t_j \in T_w : i < j && (34)\\
& x^{bin}_{t_i,w} \in \{0, 1\} && \forall w \in W, \forall t_i \in T_w && (35)\\
& y^{bin}_{t_i,t_j,w} \in \{0, 1\} && \forall w \in W, \forall t_i, t_j \in T_w : i < j && (36)\\
& v_{t_i,t_j,w} \in \{0, 1\} && \forall w \in W, \forall t_i, t_j \in T_w : i < j && (37)
\end{aligned}
$$

Constraints (19) - (21) represent if-then constraints to find buffer times (but not necessarily shortest buffer times) between trains that make use of the same switch. They express that if train t_i passes switch w earlier than train t_j in the interval $[0, 60]$, the duration between these passages, $x_{t_j,w} - x_{t_i,w}$ ($\in [0, 60]$), is a buffer time between those two trains and vice versa. Remark, however, that this buffer time is not necessarily the shortest. The order of t_i and t_j at switch w in $[0, 60]$ is described by the binary variable $v_{t_i,t_j,w}$, which equals 1 if t_i passes switch w before t_j in the interval $[0, 60]$ and 0 otherwise. The variable $y^{aux}_{t_i,t_j,w}$ is introduced in the model to represent this (not necessarily shortest) buffer time between t_i and t_j in switch w, and thus is obtained by subtracting the passage time of the train that arrives first at switch w in the interval $[0, 60]$ from the passage time of the other train. Remark that this value is always positive. We use these variables $y^{aux}_{t_i,t_j,w}$ again in constraints (22) - (24) to find the shortest buffer time between the two trains. In fact, constraints (19) - (21) define the variable $y^{aux}_{t_i,t_j,w}$:

IF $x_{t_j} - x_{t_i} > 0$, THEN $y^{aux}_{t_i,t_j,w} = x_{t_j,w} - x_{t_i,w} \in [0, 60]$
and
IF $x_{t_i} - x_{t_j} > 0$, THEN $y^{aux}_{t_i,t_j,w} = x_{t_i,w} - x_{t_j,w} \in [0, 60]$.

We can rewrite these if-then conditions by using the theory in footnote [iii]. Because we are maximizing the buffer times, it is enough to put an upper bound for the $y^{aux}_{t_i,t_j,w}$, i.e. we use $x_{t_j,w} - x_{t_i,w} - y^{aux}_{t_i,t_j,w} \geq 0$ for the first if-then condition and $x_{t_i,w} - x_{t_j,w} - y^{aux}_{t_i,t_j,w} \geq 0$ for the second if-then condition. In our MILP, we then get:

$$x_{t_j,w} - x_{t_i,w} \leq P v_{t_i,t_j,w} \tag{40}$$
$$-(x_{t_j,w} - x_{t_i,w} - y^{aux}_{t_i,t_j,w}) \leq P(1 - v_{t_i,t_j,w}) \tag{41}$$

$$x_{t_i,w} - x_{t_j,w} \leq P'(1 - v_{t_i,t_j,w}) \tag{42}$$
$$-(x_{t_i,w} - x_{t_j,w} - y^{aux}_{t_i,t_j,w}) \leq P' v_{t_i,t_j,w}, \tag{43}$$

where we still have to fix a value for P and P'. We can use the same binary decision variable for both if-then representations, because these if-statements are disjoint and complementary. In case $v_{t_i,t_j,w} = 0$, then $x_{t_j,w} - x_{t_i,w}$ is bounded below by -60, and in case that $v_{t_i,t_j,w} = 1$, we equally have that $x_{t_i,w} - x_{t_j,w}$ is bounded below by -60, such that we can choose P and P' equal to 60. This leads to constraints (19) - (21), where we put constraints (40) and constraints (42) together.

[iii] The condition that 'if constraint $f(x_1, \cdots, x_n) > 0$ is satisfied, then constraint $g(x_1, \cdots, x_n) \geq 0$ must be satisfied', where f and g are linear expressions in the decision variables $x_1, \cdots, x_n$, can be included into the IP model as follows:

$$f(x_1, \cdots, x_n) \leq P(1 - y) \tag{38}$$
$$-g(x_1, \cdots, x_n) \leq Py \tag{39}$$

where y is a binary decision variable and P is a large positive number such that $f \leq P$ and $-g \leq P$ hold for all values of $x_1, \cdots, x_n$ that satisfy the other constraints in the problem. Indeed, if $f(x_1, \cdots, x_n) > 0$, then constraint (38) imply that y must equal 0, such that (39) implies that $-g(x_1, \cdots, x_n) \leq 0$ or $g(x_1, \cdots, x_n) \geq 0$. The other way around, if $f(x_1, \cdots, x_n) \leq 0$, then y can be either 0 or 1, such that $g(x_1, \cdots, x_n)$ is only bounded below by $-P$, instead of 0. (Winston, 2004)

Constraints (22) - (24) are also translations from if-then constraints into the MILP-model. These constraints are included to determine the shortest buffer time between two trains that make use of the same switch on the basis of the buffer time that we fixed in constraints (19) - (21) in the variable $y^{aux}_{t_i,t_j,w}$. If $y^{aux}_{t_i,t_j,w} \in [0,30]$, then it is the shortest buffer time between train t_i and train t_j in switch w. On the other hand, if $y^{aux}_{t_i,t_j,w} \in]30,60]$, then it is not the shortest buffer time between train t_i and train t_j in switch w, but then $60 - y^{aux}_{t_i,t_j,w}$ is. The decision variable $y_{t_i,t_j,w}$ is introduced to define this shortest buffer time between trains t_i and t_j in switch w. We have the following conditions:

IF $y^{aux}_{t_i,t_j,w} - 30 > 0$, THEN $y_{t_i,t_j,w} = 60 - y^{aux}_{t_i,t_j,w}$,
and
IF $30 - y^{aux}_{t_i,t_j,w} > 0$, THEN $y_{t_i,t_j,w} = y^{aux}_{t_i,t_j,w}$.

Again by using the theory in footnote [(iii)], we find constraints

$$y^{aux}_{t_i,t_j,w} - 30 \leq P(1 - y^{bin}_{t_i,t_j,w}) \tag{44}$$

$$-(y^{aux}_{t_i,t_j,w} - y_{t_i,t_j,w}) \leq P y^{bin}_{t_i,t_j,w} \tag{45}$$

$$30 - y^{aux}_{t_i,t_j,w} \leq P' y^{bin}_{t_i,t_j,w}) \tag{46}$$

$$-(60 - y^{aux}_{t_i,t_j,w} - y_{t_i,t_j,w}) \leq P'(1 - y^{bin}_{t_i,t_j,w}), \tag{47}$$

where the right-hand-side values P and P' can here be set to 30, because $y^{aux}_{t_i,t_j,w}$ is bounded above by 60 by constraints (19) - (21). Since the if-statements are again disjoint and complementary, the same binary variable, $y^{bin}_{t_i,t_j,w}$, can be used in both representations. Moreover, also here we only set an upper bound on the variable $y_{t_i,t_j,w}$, because we are maximizing buffer times. Remark that for the case that $y^{aux}_{t_i,t_j,w} = 30$, then $y^{bin}_{t_i,t_j,w}$ is not uniquely defined by constraints (44) - (47). However, $y_{t_i,t_j,w}$ is in that case uniquely determined with a value of 30. This leads to constraints (22) - (24), where we again put the if-constraints together, in this case constraints (44) and constraints (46).

Constraints (25) fix the time between the moment at which the train enters the network and the moment at which the train passes a certain switch on its route. Here, $d_{t,s,w}$ is a fixed value determined in advance based on known distances and speed limits (and if necessary supplements and waiting times). Remark that $x^{aux}_{t,w}$ can be assigned a value outside $[0,60]$, that is why it has the superscription *'aux'* and why constraints (26) - (28) are necessary. However, we assume that $x^{aux}_{t,w}$ never exceeds the interval $[0,120]$.

Constraints (26) - (28) also represent if-then constraints to find the time instants in the interval $[0,60]$ at which a certain train passes each switch on its route that is different from the start switch of that train. We need to know these time instants to be able to find the buffer times $y^{aux}_{t_i,t_j,w} \in [0,60]$. These time instants are represented by the variables $x_{t,w}$. More specifically, if $x^{aux}_{t,w} \in [0,60]$, then we want that $x_{t,w}$ equals $x_{t,w}^{aux}$, otherwise if $x_{t,w}^{aux} \in [60, 120]$, then $x_{t,w}$ has to be $x_{t,w}^{aux} - 60$. Remark that $x_{t,w}$ is not uniquely defined if $x_{t,w}^{aux}$ equals 60. However, this does not cause any problems in constraints (19) - (21). The if-then statements are here:

IF $x_{t,w}^{aux} - 60 > 0$, THEN $x_{t,w} = x_{t,w}^{aux} - 60$,
and
IF $60 - x_{t,w}^{aux} > 0$, THEN $x_{t,w} = x_{t,w}^{aux}$.

To translate these if-then constraints to our MILP-problem, we cannot use the same trick to only bound the decision $x_{t,w}$ below or above, as its relation to the objective function is different. That is why we include two constraints to catch the equality in the 'then'-part. We use again the theory in footnote [iii]:

$$x_{t,w}^{aux} - 60 \leq P(1 - x_{t,w}^{bin}) \tag{48}$$
$$-(x_{t,w} - x_{t,w}^{aux} + 60) \leq P x_{t,w}^{bin} \tag{49}$$
$$-(-x_{t,w} + x_{t,w}^{aux} - 60) \leq P x_{t,w}^{bin} \tag{50}$$

$$60 - x_{t,w}^{aux} \leq P' x_{t,w}^{bin} \tag{51}$$
$$-(x_{t,w} - x_{t,w}^{aux}) \leq P'(1 - x_{t,w}^{bin}) \tag{52}$$
$$-(-x_{t,w} + x_{t,w}^{aux}) \leq P'(1 - x_{t,w}^{bin}). \tag{53}$$

The right-hand-side values P and P' can be set to 60, because $x_{t,w}^{aux}$ is bounded above by 120. Remark that constraint (52) is stronger than constraint (49):

$$-x_{t,w} + x_{t,w}^{aux} - 60 x_{t,w}^{bin} \leq 60 \tag{49'}$$
$$-x_{t,w} + x_{t,w}^{aux} + 60 x_{t,w}^{bin} \leq 60, \tag{52'}$$

because $x_{t,w}^{bin}$ is a binary decision variable. Also constraint (50) is stronger than constraint (53):

$$x_{t,w} - x_{t,w}^{aux} - 60 x_{t,w}^{bin} \leq -60 \tag{50'}$$
$$x_{t,w} - x_{t,w}^{aux} + 60 x_{t,w}^{bin} \leq 60. \tag{53'}$$

Indeed, if $x_{t,w}^{bin} = 1$, then both constraints reduce to the same constraint and if $x_{t,w}^{bin} = 0$, then the upper bound of $x_{t,w} - x_{t,w}^{aux}$ in constraint (50) is lower. If we combine constraints (48) and (51), then this leads to constraints (26) - (28).

The last constraints (29) enforce that the times at which the trains enter the network are included in the interval $[0, 60]$. This concludes the explanation of timetable determination model.

3.3 Small case study

We demonstrate our approach on the fictive and schematic network in Fig. 1. The bold circles represent switches and the numbers next to the arrows are time durations to get from one end of the corresponding link to the other end. We consider 6 trains on our network for which the origin and destination in the network and all the possible routes that link these points are represented in Table 2. These routes are visualized in Fig. 2. We implement both the route choice model and the timetable model in C_{++} and they are solved by CPLEX 12.6 on an Intel Core i7 with 8 cores. For this small case study, both models are solved in less than one second.

The output of the route choice model is presented in Fig. 3. The maximum number of times that a switch is used is 3, which already implies that the minimum buffer time over all switches can never be higher than 20. The sum of the quadratic number of usages over all switches is 63. The output of the timetable determination model is presented in Fig. 4. The table on the left presents the minimal timespan between every train pair that has at least one switch in common. In the figure on the right, each colour represents a train. The minimal timespan between two trains in the optimal solution is 20 minutes. The sum of the minimal timespans between every train pair is 145 minutes.

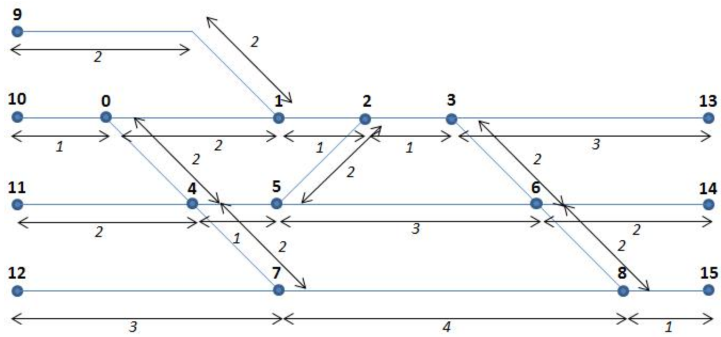

Fig. 1 Network of the small case study

Table 2 Train information

Train	Origin	Destination	Possible routes
0	15	10	5, 6, 7, 8
1	15	12	9
2	13	10	3, 4
3	11	14	1, 2
4	11	14	1, 2
5	9	13	0

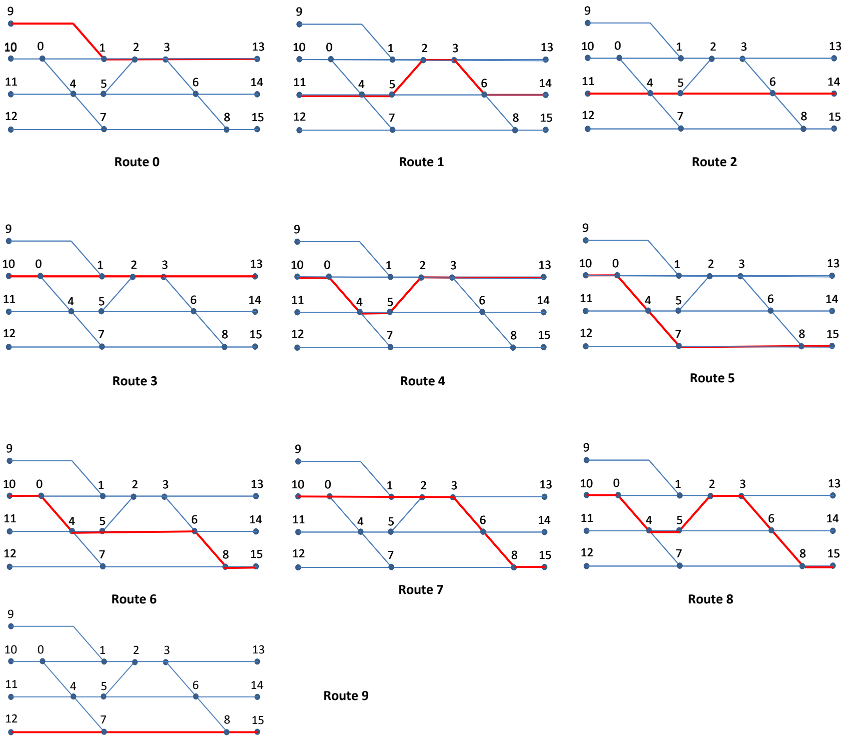

Fig. 2 Train routes

Train	Assigned route
0	5
1	9
2	3
3	2
4	2
5	0

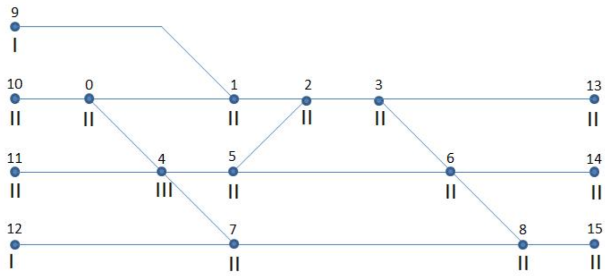

Fig. 3 Output of the route choice model: route assignment and switch usage

Train pair	minimal time span
0 - 1	30
0 - 2	30
0 - 3	20
0 - 4	20
2 - 5	25
3 - 4	20

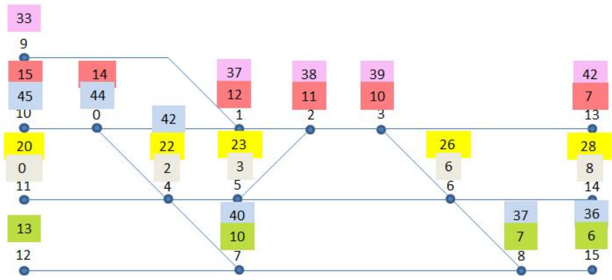

Fig. 4 Output of the timetable determination model: minimal time spans between train pairs (in minutes) and passage times (train 0, train 1, train 2, train 3, train 4, train 5)

4 Case study

We perform a case study on the dense railway area of Brussels, which contains 3 out of 5 of Belgium's busiest stations. It also includes the beginning of the open tracks, the outer grids, and the entrances to the shunt yards. The core of this area is presented in Fig. 5 and a schematic overview of the entire station area is presented in Fig. 6.(iv) We consider every border point, every point where two tracks cross and every platform as a switch in the network. We then have 481 switches. The network contains 6 station areas with 22, 6, 6, 6, 12 and 12 platforms. The total number of routes that starts at a border point or platform track and ends at a border point or platform track amounts 28 970 248! We consider 85 trains that pass through this network during the morning peak hour between 7 and 8 o'clock. In a first (preprocessing) step we generate for each train all the routes that connect its origin and destination in the network.

We apply some simplifications to make the models manageable on this complex network. To limit the number of routes (and thus the number of decision variables) in the route choice model, we consider for every feasible platform combination only one route per train, for example the shortest route or the route that contains the smallest amount of switches. Since most border points are connected to different platforms, this still leaves many possible routes per train from its origin to its destination (median: 361, minimum: 15). However, one could argue that this converts the problem from a routing problem to a platforming problem. This routing problem can be solved within 20 minutes up to a gap of 0,2%:

$$\text{gap} = \frac{\text{best found integer solution - best found lower bound}}{\text{best found lower bound}}. \tag{54}$$

For the timetable determination problem, we reduce the number of switches in which buffer times have to be optimized. We only maximize the buffer time

(iv) Source: http://www.infrabel.be/sites/default/files/documents/ns_c-01-map-net-1045901_1.pdf, consulted in September 2014.

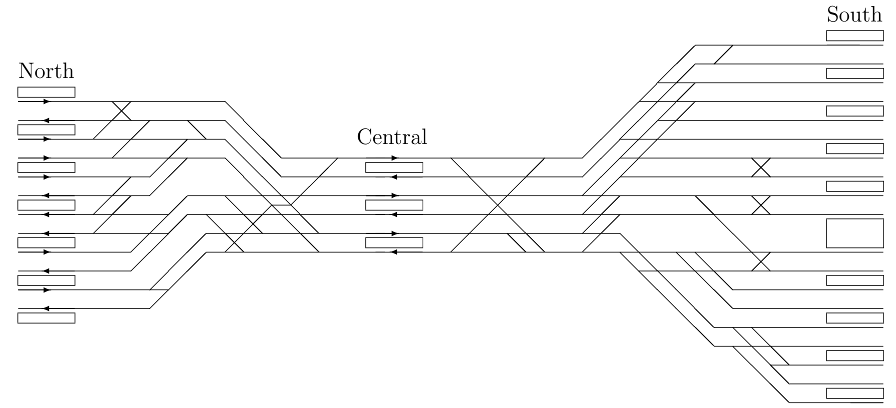

Fig. 5 The core of Brussel's dense railway area

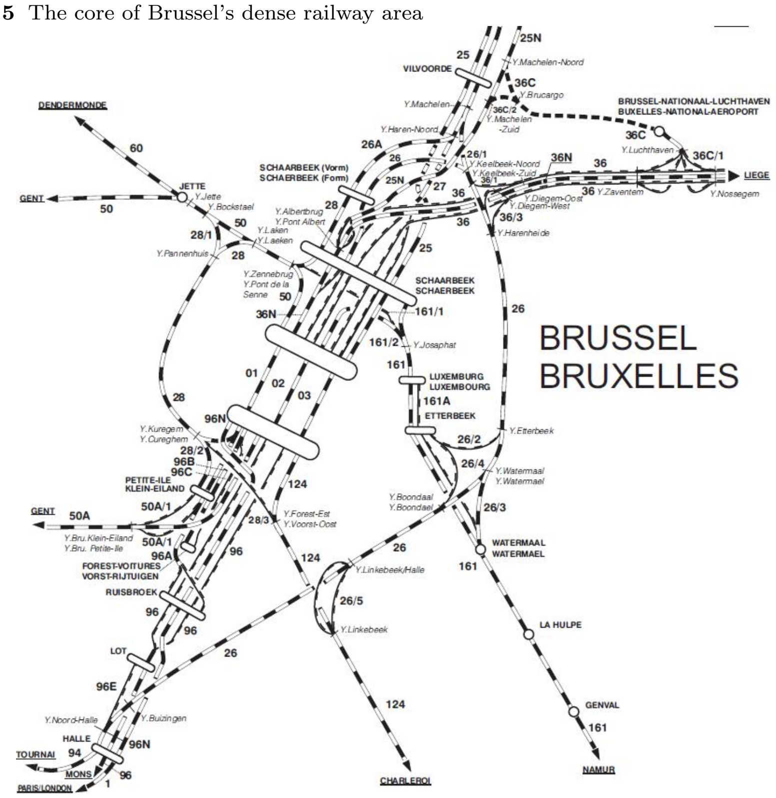

Fig. 6 A schematic overview of the entire station area of Brussels

in every switch with a switch usage (strictly) higher than 8 and we impose a buffer time of at least five minutes in switches with a switch usage of 8 or less:

$$y_{t_i,t_j,w} \geq 5 \qquad \forall w \in W, t_i, t_j \in T_w : g_w \leq 8. \tag{55}$$

We take 8 usages as a boundary on the switch usage in constraint (55) to significantly reduce the number of switches in which the spreading has to be optimized during the timetable construction (reduction from 482 to 114 switches). We use 5 minutes as a lower bound on the minimal buffer in the switches with usage lower than 8 to provide a workable minimal buffer in practice and to leave enough flexibility in constraints (55).
As the timetable is cyclic, a solution to the timetabling problem can be adapted to another solution by adding a fixed value to all the passage times. To cut off all these equivalent solutions, we fix the start time of one of the trains, for example train 0:

$$x_{0,s} = 0. \tag{56}$$

Furthermore, the route choice model provides an upper bound on the minimum buffer time. As the maximum usage of a switch equals U, then the minimum buffer time will not be higher than $\frac{60}{U}$. Thus, we can add the constraint

$$z \leq \frac{60}{U}. \tag{57}$$

We compare some performance criteria of different routing plans. A first routing is one that was implemented by the Belgian railway companies on the network in 2012. We will refer to this routing plan as the reference. A second routing plan is the result from applying the optimization algorithm of Dewilde et al. (2013, 2014) on a reference timetable (from Infrabel) and the reference routing plan. Dewilde et al. (2014) maximize the sum of the minimal buffer times between (release and reserve times of) every two trains by an iterative approach. They optimize the routing for a fixed timetable and platform assignment on the one hand and they make small changes to the timetable and platform assignment on the other hand. The third routing plan is the result from applying the extended version of the optimization algorithm of Dewilde et al. (2014) on the same reference timetable and the reference routing plan (Burggraeve et al., 2015). In this approach, the selection of the buffer times that are enlarged depends on the probability of causing delay propagation due to recurring delays and on the number of passengers that will be affected in case of delay propagation through this buffer. The fourth routing plan is constructed by the algorithm presented in this paper.

Table 3 compares the switch usage of the different approaches. In the first row the maximum switch usage is given. We see that our model was not able to improve this maximum usage. The observation that this number is the same for all of the different approaches can be explained by the fact that the origins and destinations of the trains in the case study dictate this maximum switch usage. The second row presents the quadratic penalizations of the switch usages, which is the second criterion that is optimized in the route choice model. We see that this quadratic penalization decreased with 7,6% compared to the reference routing plan and with 8,0% compared to the routing plan of Burggraeve et al. (2015). The third and fourth row present the number of switches with a switch usage strictly higher than 6 and 12 respectively. Remark that the number of switches with a switch usage equal to or higher than 0 is 481 and with a switch usage strictly higher than 16 is 0 for all approaches. The number of switches with more than 6 or 12 usages is decreased from 196 to 191 and 55 to 48 respectively for the reference routing plan and from 215 to 191 and 62 to 48 respectively for the routing plan of Burggraeve et al. (2015). The output of the route choice model gives us a detailed view on the planned switch usage on our network. This information can be useful for disruption management.

Table 3 Switch usage in the routing plans

Approach	Reference	Dewilde et al. (2014)	Burggraeve et al. (2015)	Route choice model
$\min \max_{w \in W} g_w$	16	16	16	16
$\sum_{w \in W} g_w^2$	27565	27410	27698	25481
$\#w : g_w > 6$	196	198	215	191
$\#w : g_w > 12$	55	62	62	48

We cannot directly compare the output from the timetable determination model with the timetables from the other approaches, because our model maximizes buffer times between train passages in switches, but does not take reserve and release times into account yet.

5 Conclusion and future work

This paper presents a model to build a railway timetable and a routing plan for a busy railway station area from scratch. This approach focusses on the area where designing a robust timetable is the most difficult, i.e. in the largest bottleneck. We assume that the resulting timetable can be made feasible outside this bottleneck without many changes since much less constraints are present there. In order to improve the total travel time of all passengers in practice in case of frequently occurring small delays, we optimally spread the trains in space and time. The maximum switch usage and the switch usage of each individual switch is minimized by the route choice model. The timetable determination model maximizes the minimal buffer time over all switches, as well as the buffer time between every two trains that share a part of the infrastructure.

Based on a small case study we illustrated that our approach is indeed effective. Furthermore, we showed that our route choice model can decrease the switch usage in Brussel's railway station area with 7-8% compared to a reference routing plan and the routing plan of Burggraeve et al. (2015). We also have a detailed view on the planned switch usage in our network. This information can be useful for disruption management.

The first thing to do next is including reserve and release times of switches in the timetabling model such that the constructed timetable not only takes the passage times in switches but also the occupation times into account and becomes useful in practice. This will enable a full validation of the timetable determination model on the complex case study. It would also be interesting to further adapt the presented models or further improve the solution method in order to make them more efficient for large networks.

References

Borndörfer, R., Erol, B., Graffagnino, T., Schlechte, T., Swarat, E., Aggregation Methods for Railway Networks, Journal of Rail Transport Planning & Management 1(1), 38-48 (2011).

Burggraeve, S., Dewilde, T., Sels, P., Vansteenwegen, P., Improving passenger robustness by taking passenger numbers and recurring delays into account on the tactical level, Proceedings of the 6th International Conference on Railway Operations Modelling and Analysis, 1-15 (2015).

Dewilde, T., Sels, P., Cattrysse, D., Vansteenwegen, P., Defining robustness of a railway timetable, In: Proceedings of 4th International Seminar on Railway Operations Modelling and Analysis (RailRome2011), Rome, Italy (2011).

Dewilde, T., Sels, P., Cattrysse, D., Vansteenwegen, P., Robust railway station planning: An interaction between routing, timetabling and platforming, Journal of Rail Transport Planning & Management, vol. 3, pp. 68-77 (2013).

Dewilde, T., Improving the robustness of a railway system in large and complex station areas, doctoral dissertation, KU Leuven, Leuven (2014).

Cacchiani, V., Toth, P., Nominal and robust train timetabling problems, European Journal of Operational Research 219, pp. 727-737 (2012).

Goldratt, E.M., Cox, J., The Goal: A Process of Ongoing Improvement, North River Press (1986).

Liebchen, C., Möhring, R., The modeling power of the periodic event scheduling problem: Railway timetables - and beyond, Algorithmic methods for railway optimization, Ed. by F. Geraets, L. Kroon, A. Schöbel, D. Wagner, D. Zaroliagis, vol.4359, Lecture Notes in Computer Science, Springer Berlin Heidelberg, pp. 3-40 (2007).

Serafini, P., Ukovich, W., A Mathematical model for periodic scheduling problems, SIAM Journal on Discrete Mathematics 2, pp.550-581 (1989).

Winston, W.L., "If-Then Constraints", Operations Research Applications and algorithms, 4th Edition, pp. 490 (2004).

Zwaneveld, P.J., Kroon, L.G., Romeijn, H.E., Salomon, M., Dauzre-Prs,S., van Hoesel, S.P.M., Ambergen, H.W., Routing trains through railway stations: Model formulation and algorithms, Transportation Science, vol. 30, pp. 181-194 (1996).

Zwaneveld, P.J., Kroon, L.G., van Hoesel, S.P.M., Routing trains through a railway station based on a node packing model, European Journal of Operational Research, vol. 128, pp. 14-33 (2001).