

Optimization of supplements and buffer times in passenger robust timetabling

Sofie Burggraeve ^{a,1}, Pieter Vansteenwegen ^a,

^a KU Leuven Mobility Research Centre - CIB, KU Leuven
Celestijnenlaan 300, 3001 Leuven, Belgium

¹ E-mail: sofie.burggraeve@kuleuven.be, Phone: +32 (0) 16 37 20 23

Abstract

This paper proposes an iterative approach to construct a passenger robust railway routing plan and timetable from scratch that also takes supplement and buffer time allocation into account on the signaling level. Each iteration is based on four pillars, which are subsequently executed. First a routing plan that optimizes infrastructure usage is constructed. Secondly, a timetable is constructed that optimizes buffer times between trains while taking passenger numbers into account. Thirdly, simulation is used to evaluate the passenger robustness of the routing plan and the timetable. Finally, the simulation outcome is used to make a new supplement assignment for the next iteration. The algorithm is tested on a case study for the complex station area of Brussels (Belgium). The resulting timetable and routing plan are much more passenger robust than the existing combinations of timetables and routing plans for this case, developed during research or implemented in practice.

Keywords

Passenger robustness, timetabling, routing, simulation

1 Introduction

In many countries, railway passengers plan their trip based on train arrival and departure times in stations. An overview of these timings can be found in timetables published online or in the stations. In contrast to this simple overview, the operator needs a far more detailed timetable to safely and efficiently operate the railway system. Just as is required that only one train at a time can use a platform, this also holds for other parts of the network. These parts of the network are delimited by signals and are referred to as *sections*. The signal prohibits a train driver to pass if another train is using (a part of) its next section. So the train routes have to be carefully planned from signal to signal. A timetable is *conflict-free* if no two trains are planned to use overlapping sections at the same moment. Furthermore, a lot of choices have to be considered while constructing this detailed timetable. The planning will not be efficient if train drivers are planned to stop by signals outside the platform area. Even if such an inefficient stop is not planned, it may occur in practice if trains are scheduled too tightly. There are two important factors for avoiding delay propagation: supplements and buffer times. A *supplement* is extra time that can be added in the timetable to the technical minimum running or dwell time of a train at different places on its route. A *buffer time* is extra time added to the technical minimum time that is guaranteed between two trains at a shared part of the network. Supplements are typically included such that trains

can absorb their delays during operations. Buffer times can be provided such that a delayed train does not affect a next train using the same part of the network. However, the more supplements and buffer times that must be assigned, the more difficult it becomes to find a feasible conflict-free schedule. The inclusion of more supplements also constrains the buffer time that can be provided and the other way round. So, the inclusion of supplements and the scheduling of buffer times has to be wisely thought-out and the profit of both has to be balanced against each other to provide a good service. Even more so because supplements lengthen the travel time of the passengers, which is undesirable in case the train does not suffer from delays. Buffer times only have this disadvantage if they lengthen the transfer time between two trains.

Our objective is to construct a *passenger robust schedule*, which means that the total passenger travel time in practice in case of small daily delays is minimized (Dewilde, 2014). So travel times have to be short but also reliable in non-ideal circumstances.

Our methodology can be shortly summarized as follows. Starting from a given assignment of supplements, the buffer times between trains are optimized on the signaling level while constructing a feasible and conflict-free routing plan and timetable. This timetable is simulated and information on the ‘causers’ and the ‘victims’ of delay propagation are used to construct a new detailed assignment of supplements, which is the input for the next optimization round. The optimization stops when a balance between the achieved buffer times and the updated supplements is found that incurs a passenger robust timetable. The main contributions of this paper are:

- An iterative approach for computing passenger robust timetables for complex railway station areas on the signaling level.
- The heuristic improvement of both buffer times and supplements while striving for the optimal passenger travel time in practice.
- A procedure to evaluate and improve dwell and running time supplements in a timetable, based on simulation.

The focus in this paper is on the planning of complex railway station areas. The planning of such a bottleneck is at the same time complicated and crucial for the performance of the whole network (Burggraeve and Vansteenwegen, 2017). Like Burggraeve and Vansteenwegen (2017), we assume that the planning can thereafter easily be extended to the whole network. This paper proposes an approach that balances buffer times and supplements for these complex railway station areas. A *station area* consists of a number of stations close to each other, for example Brussels-South, Brussels-Central, Brussels-North and Brussels-Schaarbeek, and the grids connecting these stations with each other and with the rest of the network. We divide a station area in *platform areas*, which consist of the platforms in the stations, and *grid zones* in between these platform areas. Supplements provided in platform areas are referred to as *dwell time supplements*, while supplements provided in the grid zones between platform areas are referred to as *running time supplements*. If a train arrives with a delay on its platform, a dwell time supplement can make sure that the train leaves the station on time or with a smaller delay. In case the train is not delayed, it dwells longer, i.e. the train stands still for the minimal dwell time and the dwell time supplement. A running time supplement provides extra time for absorbing a train delay in a grid zone between two platform areas, for example when it is forced to stop at a signal because the next section is

not yet free. In case the train is not delayed, it lowers its speed below its normal speed, such that it does not arrive too early on its next platform.

In Section 2, we discuss existing approaches to construct a timetable while taking buffer times and supplements into account. Then, we elaborate on our methodology in Section 3. Thereafter, a case study is described and the results for this case study are presented and discussed in Section 4 and 5 respectively. The paper is concluded in Section 6.

2 State of the art

Timetables can be constructed on the microscopic scale (signaling level) and on the macroscopic scale. Planning on the macroscopic scale only provides arrival and departure times in stations, but does not guarantee that conflict-free train routes through the network can be assigned to guide the trains to their platform at the scheduled times. Cacchiani and Toth (2012) published a literature review on macroscopic timetable models, like the Periodic Event Scheduling Problem (e.g. Serafini and Ukovich, 1989; Liebchen and Möhring, 2007; Schmidt and Schöbel, 2015; Burggraeve et al., 2017) and the Train Timetabling Problem (e.g. Caprara et al., 2002; Cacchiani et al., 2016).

Planning on the microscopic scale guarantees a conflict-free timetable on the signaling level. This implies that the timetable can be plainly implemented in practice, but it is more complex to construct a microscopic timetable since much more constraints have to be taken into account. Existing literature focuses on how to check a macroscopic timetable for feasibility on the microscopic scale and how to optimally adapt this macroscopic timetable to obtain this feasibility in case it is not microscopically conflict-free (e.g. Bešinović et al., 2017). The existing approaches differ from our approach in the fact that we purely focus on the signaling level. This is because for complex railway station areas with many switches and route options, constructing a timetable on the signaling level seems the only way to avoid many trial and error iterations and to reach an optimal infrastructure usage of these areas. Also Caimi et al. (2009) focus on the construction of a conflict-free timetable and routing plan for networks that they divide in condensation zones, where capacity is limited, e.g. dense stations, and compensation zones, where the traffic is less dense, e.g. corridors. They remove time reserves to the compensation zones, which is not recommended for large and complex railway station areas which contain multiple platform areas close to each other and a very high number of routes. This is shown in Section 5.

We now first describe the work of Sels et al. (2016) and Bešinović et al. (2016a) which both constructed integrated approaches to plan routing and timetabling by using an interaction between the macroscopic and the microscopic scale. Other integrated approaches without feedback from the microscopic to the macroscopic level are for example described in Schlechte et al. (2011) and De Fabris et al. (2013). Thereafter we describe our own previous research on constructing a microscopic timetable and route plan (Burggraeve and Vansteenwegen, 2017) and we indicate the added value of the new research presented in this article. We end this section with approaches that purely focus on supplement allocation.

Sels et al. (2016) developed an approach to plan a timetable on the macroscopic scale and thereafter assign routes to the trains on the microscopic scale. During the timetabling phase the impact of daily delays on the performance of the timetable is taken into account by using the passenger travel time in practice as objective function. The passenger travel time in practice is estimated by a function that depends on the minimum possible travel time for each passenger and the added supplement to each passenger's trajectory. The shape of

this estimation function depends on the expected primary delay functions of the (involved) trains. The best amount of supplements to add is decided by the model based on the estimate of the passenger travel time in practice, i.e. the corresponding function value of the estimation function. However, this approach does not assure the existence of a conflict-free assignment of routes. Complex station areas form a bottleneck for this approach. Occurring conflicts have to be solved by trial and error with small adjustments to the timetable. Moreover, an efficient infrastructure usage is not taken into account in this approach. In Bešinović et al. (2016a), an integration of a microscopic and a macroscopic model in order to construct a feasible, stable and robust timetable is developed where the macroscopic parameters are iteratively updated by recomputing them at the microscopic level. The final timetable is microscopically conflict-free. The approach works well and fast for a network with 40 trains and 26 stations. However, in the considered network at most four tracks lie in parallel and the maximum occupation rate is 54.7%. Brussels-South and Brussels-North both have over 10 tracks in parallel and there pass 85 trains during peak hour. The percentage of time that the platforms are blocked is over 50% with a maximum of 77.3%. In fact, by restricting our network to the platform area, we calculated a lower bound on the maximum occupation rate of Brussels station area, since we did not take the interactions and differences in speed limits in the grid zones into account for compressing the timetable.

In Burggraeve and Vansteenwegen (2017) a microscopic routing and timetabling model are presented that strive for passenger robustness. These models take the microscopic layout of the network into account and assign blocking times to the trains and the sections on their routes such that the final timetable is conflict-free and the buffer times between the blocking times are maximized. The *buffer times* are defined per node as the time between the blocking times of two trains that block a section that contains that node. Thus the buffer time is the time distance between the release time of a first train that blocks (a section that contains) that node and a second train that blocks (a section that contains) that node. The *blocking times* are defined per section as in Pahl (2008): ‘The blocking time consists of the time for setting up, watching the signal, approaching the section, traversing and clearing the section and releasing the signal and depends on the type of the train, on the route that the train is assigned to and the specific section on this route.’ The blocking time of a train in a node equals the train’s blocking time of the train’s section that contains that node. An important note to make here is the distinction between the microscopic nature of the models in Burggraeve and Vansteenwegen (2017) and the information that is available for a certain case study. If all the correct microscopic data is available, the models assure conflict-freeness at the microscopic scale. However, the models in Burggraeve and Vansteenwegen (2017) can also be used for case studies for which not all microscopic data is explicitly available and normative or approximate values are used instead. In that case it is clear that conflict-freeness is not assured at the microscopic scale for that specific case study. An improved version of both models takes part in the current approach. The routing model is performed first and is not constrained by a timetable. It focuses on optimally using the available infrastructure. The routing model can be found in Burggraeve and Vansteenwegen (2017, equations (5)-(13)). We indicate briefly that the routing objective function of Burggraeve and Vansteenwegen (2017) takes only the amount of usages into account and not the usage times. However, if the routing model is assigned, the total node usage time for each node can be calculated. This total node usage time can be understood as follows. For each node on a train’s route, the train contributes to the total node usage time of that node with its blocking time of the section that contains that node.

For a detailed explanation of that routing model, we refer the reader to the description of the routing model in (Burggraeve and Vansteenwegen, 2017). In the current research an extra check is added to the routing model in order to find a solution for which the total usage time of a node does not exceed the available time. This is necessary to be able to construct a feasible timetable afterwards. Then the timetabling model is solved. It takes the microscopic routing of the trains as input and is thus restricted by these routes. The timetabling model can also be found in Burggraeve and Vansteenwegen (2017, equations (16)-(35)). Also here we refer the reader to Burggraeve and Vansteenwegen (2017) for a detailed explanation of the timetabling model. The objective function of that timetabling model optimizes the buffer times between the trains. In the research presented in this paper, the objective function has changed a little in order to weigh the buffer times between the trains according to the number of passengers that would be affected in case of knock-on delays. Apart from the mentioned improvements made to the routing and the timetabling model, the largest improvement compared to Burggraeve and Vansteenwegen (2017) is that the approach presented in this paper allows an extra degree of freedom to create a passenger robust schedule. While in Burggraeve and Vansteenwegen (2017) only the effect of buffer times is taken into account, we here also include the decision on the location and amount of supplements. The profit of buffer times and supplements is balanced against each other while creating a passenger robust schedule.

Apart from the integrated approaches, there are also approaches that focus on estimating, determining or improving the amount or placement of supplements in a timetable. In Rudolph (2003) is shown by simulation that the location of supplements affects the performance of the timetable. A supplement just before a station accommodating many connecting trains or a high passenger volume proves to be most effective in reducing delays. The authors indicate that their findings are not necessarily valid for railway bottlenecks before major stations, as considered in this paper. In Yuan and Hansen (2007) different distributions of the technical minimum dwell time of trains are considered to analyze the interaction between knock-on delays and scheduled buffer times. Here the focus is not on optimizing the supplements but on optimizing the station capacity utilization. Kroon et al. (2008) allocate running time supplements to a single train on a number of consecutive trips. As a result of the many interactions between trains in complex station areas this approach is not straightforwardly applicable for this input. Khan and Zhou (2010) propose a decomposition approach to optimize supplement allocation for individual trains, that starts from a two-stage recourse model. They accentuate that it is important to weigh the advantages of including supplements against the lengthening in planned travel time. They tested their approach on a high speed rail corridor. Niu and Meng (2014) also focus on supplement allocation in train timetables. Like the approach in this paper, they iterate between a timetabling phase and an evaluation phase, but their evaluation phase consists of a dispatching model which minimizes the total delay under a certain delay scenario. They tested their approach on high-speed passenger lines and concluded that slack time allocation highly affects the train timetable reliability.

3 Methodology

In this section, we elaborate on our methodology. An overview of the algorithm is presented in Figure 1. The approach works iteratively and consists of four pillars. These four pillars are performed consecutively in each iteration. First the input and the output are described,

then the four pillars are explained and how they interact with each other.

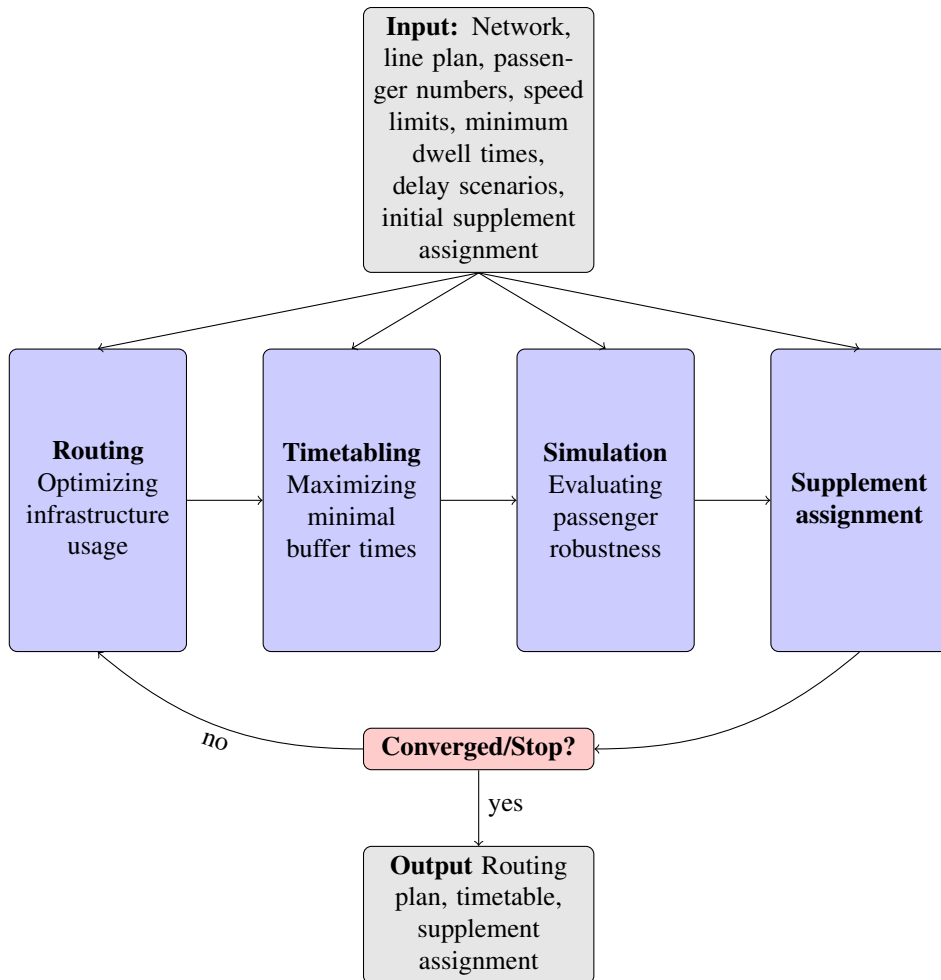


Figure 1: Algorithm lay-out.

3.1 Input

The input consists of the infrastructure for which a timetable and a routing plan has to be constructed. This includes the locations of the signals, the switches and the platforms. The algorithm starts from a fixed line plan that contains information on the incoming and outgoing tracks, the stops and the frequency of each train on the considered (part of the) network. The passengers numbers on these lines are considered to be known, including the changes in passenger numbers in the different platform areas. Also the speed limits on all sections and the minimum dwell times are input to the algorithm. The speed limits are used to calculate the minimum running time of the trains based on the train type, e.g. high speed train, intercity train, etc. To accommodate for the speeding up and slowing down in the

platform areas, fixed times, based on the platform area, the train direction and train type, are also considered as input and used for calculating the running times. In order to evaluate the constructed timetable, delay scenarios are fixed beforehand. In case historical data is available, delays can be drawn from distributions estimated from these historical data, e.g. exponential distributions. Furthermore, an initial supplement assignment can be included to start the algorithm.

3.2 Output

The output consists of the best performing routing plan and timetable constructed during the iterations, i.e. with the best passenger robustness which is the smallest total passenger travel time in practice. Both the output routing plan and the timetable are specified on the microscopic level. For each train and for each node and each section on its route, the timetabling model outputs the reservation and release times (begin and end times of the blocking times). Another output is the final amount and distribution of the supplements per platform area and per grid zone. The blocking times can be represented in a diagram with the nodes on the vertical axis, the time on the horizontal axis and where the trains are represented with colors. However, this diagram is not easy to interpret for a dense network with a complex infrastructure lay-out since the links (i.e. tracks) between the nodes are not visible on this diagram. That is why we restrict this diagram for presenting the output of our case study later in this paper, to an interesting part of the network, namely the platform area Brussels-Central.

3.3 Algorithm

This section elaborates on the four pillars of the algorithm and how they interact with each other. These four pillars are executed in each iteration until a stop criterion is satisfied. The input that changes in each iteration is the assignment of dwell and running time supplements for each train in each platform area and grid zone. In the first pillar, trains are assigned to routes. Routes are defined as a sequence of nodes in the network, where a node can be either a switch, a platform or a border node of the network. If the routes are assigned, it is exactly clear which signals the train will pass and which sections the train has to block when traveling throughout the network. This information is output of the routing model and is used as input in the timetabling model. As mentioned earlier, if the routing model is assigned, the total node usage time for each node can be calculated. Unless the total node usage time for at least one node is not feasible anymore, the routing plan will remain the same over the iterations. The total node usage time can increase and become infeasible due to the new supplement assignment in the current iteration. In that case, a new routing plan is constructed for which the total node usage time does not exceed the threshold value for either of the nodes. In the second pillar, a timetable is constructed. The timetable (and thus also the train orders and the blocking times) will change in each iteration since it is directly affected by the new supplement assignment. In the third pillar, the passenger robustness of the constructed timetable and routing plan for the current iteration are computed. This is done by simulation. Since our idea is to use probability distributions of historic delays, the same delay scenario is used during the entire algorithm. In the fourth pillar, a new supplement assignment for the next iteration is made based on the non-used supplements and the propagated delays. After executing the fourth pillar, the stop criteria are checked.

The four pillars and the stop criteria are now explained in detail.

3.3.1 Routing

The routing model of Burggraeve and Vansteenwegen (2017, equations (5)-(13)) constructs a routing plan which optimizes the infrastructure usage within a station area. The underlying logic is that the less trains use a node the easier it will be to spread these usages in time. Not the duration of each usage is taken into account, but the number of usages. We use this model to construct a routing plan. Once the trains are assigned to a route, the total node usage time for each node is calculated. As we already mentioned in the literature review, we improved the routing by adding a check if none of these total node usage times exceeds the available time, for example the period length of the timetable. Since if the node usage time exceeds the period length, this means that the sum of the times that trains block this node within one period is larger than the period length, which implies overlapping blocking times and thus conflicts! If so, an extra constraint is included in the model for each of the critical node usage durations and the model is solved again. This extra constraint is now explained. Let T be the set of trains, R be the set of routes and W be the set of nodes. Route $r \in R$ contains node $w \in W$ is indicated as $w \in r$. Let $x_{t,r}$ be the binary decision variable that equals 1 if train $t \in T$ is assigned to route $r \in R$ and 0 otherwise. Let $b_{t,r,w}$ be the time that train $t \in T$ blocks node $w \in W$ in case train t is assigned to route $r \in R$, including the technical minimum blocking time and the dwell or running time supplement. Let P be the available time, for example the period length of the timetable, or perhaps more realistically, the value that may not be exceeded to stay below the maximum allowed occupation rate. For each node w for which the node usage time exceeds the allowed time, the following constraint is included in the routing model and the routing model is solved again.

$$\sum_{t \in T, r \in R, w \in r} b_{t,r,w} x_{t,r} < P. \quad (1)$$

The construction and check are repeated until the final routing plan optimally spreads the trains over the available infrastructure without exceeding the available time of any node. Since the routing model is solved in less than a minute, even when an additional constraint is added a few times, its computation time remains small compared to the construction of a timetable.

3.3.2 Timetabling

The timetabling model of Burggraeve and Vansteenwegen (2017, equations (16)-(35)) constructs a timetable which optimizes the buffer times between trains. Including sufficient buffer time between trains avoids the propagation of delays and favors shorter and more reliable travel times in practice for passengers. To minimize the passenger travel time in practice, sufficient buffer times precursory to highly crowded trains are even more important. Indeed because a delay affects the travel time of all the on-board passengers. That is why we here alter the objective function of Burggraeve and Vansteenwegen (2017). Their objective function is twofold. First the overall minimum buffer time is maximized. We kept this first part. Secondly, the sum of the minimum buffer times between train pairs is maximized. The passengers that are affected by an insufficient buffer time between two trains are the passengers on the second train, since a delay of the first train larger than the buffer time propagates to the second train. That is why weights are now included in the second part to account for the number of passengers that may be affected in case the buffer time is

insufficient for avoiding delay propagation. So instead of treating all trains equally, trains that transport a high number of passengers are prioritized. The weights have to satisfy the next two conditions. First, the more passengers can be affected by delay propagation, the more impact this buffer time must have on the objective function. Secondly, the smaller the buffer time, the more impact this buffer must have on the objective function. This second condition is fulfilled by the original objective function of Burggraeve and Vansteenwegen (2017) without weights. For the first condition, a first intuition could be to use the number of passengers that is affected if the buffer time is insufficient, directly as weight for this buffer time. However, this has not the desired effect. This is because of the fact that the second part of the objective function then maximizes the sum of minimum weighted buffer times between train pairs. The more affected passengers, the higher the weight, the less probable that this weighted buffer time is also the actual minimum weighted buffer time between these two trains and vice versa. By contrast, using one over the number of passengers that is affected works well. The more passengers may be affected the smaller the weight and thus the larger the impact on the objective function, which maximizes the minimum weighted buffer times. Nonetheless, to avoid fractional coefficients in the model, the buffer times are split into K groups based on the number of passengers that may be affected. This is done by splitting the set from zero up to the maximum number of passengers that may be affected into K disjoint, but consecutive intervals. Group 1 contains the buffer times that may affect the least number of passengers and group K contains the buffer times that may affect the highest number of passengers. Let C be the smallest common multiple of $\{1, \dots, K\}$ and let the weight for a buffer time B be $C / \text{group}(B)$, where $\text{group}(B)$ is an integer referring to the number of the group of buffer times in which buffer time B is ranked. These weights are integers by construction and satisfy the two conditions. In case the K groups are intervals of equal length, the weighted buffer time decreases linearly with the group number, i.e. the number of passengers that are affected. The weighted buffer times also increase linearly with the magnitude of the buffer time itself.

The objective part in the timetabling model, formula (15) in (Burggraeve and Vansteenwegen, 2017),

$$\max \sum_{\substack{t_i, t_j \in T: i < j \\ \wedge r^{t_i} \cap r^{t_j} \neq \emptyset}} \min\{\text{buf}_{t, t', w} | t, t' \in \{t_i, t_j\} : t \neq t', w \in r^{t_i} \cap r^{t_j}\}, \quad (2)$$

is replaced with

$$\max \sum_{\substack{t_i, t_j \in T: i < j \\ \wedge r^{t_i} \cap r^{t_j} \neq \emptyset}} \min\left\{\frac{C}{\text{group}(\text{buf}_{t, t', w})} \text{buf}_{t, t', w} | t, t' \in \{t_i, t_j\} : t \neq t', w \in r^{t_i} \cap r^{t_j}\right\}. \quad (3)$$

In the linearized model, only constraint (18) changes to

$$\min \text{buf}_{t_{\min\{i, j\}}, t_{\max\{i, j\}}} \leq \frac{C}{\text{group}(\text{buf}_{t, t', w})} \text{buf}_{t_i, t_j, w} \quad \forall t_i, t_j \in T : t_i \neq t_j, w \in r^{t_i} \cap r^{t_j}. \quad (4)$$

3.3.3 Simulation

The simulation tool of Dewilde et al. (2014) is used to measure the performance of a railway timetable and routing plan and to compare the performance of different schedules. The

simulation tool is described in detail in Dewilde et al. (2014) and Burggraeve and Vansteenwegen (2017). The main performance criterion that it measures, is the passenger robustness of a timetable and routing plan. The simulation tool is a discrete event Monte Carlo simulation where events consists of trains blocking their next section and in which (small) initial train delays are drawn from a probability distribution in each simulation run. Like for the construction of the timetable for Brussels, the simulation uses the same available data from Infrabel to calculate the blocking times of the trains. For each train, its first reservation time in the network (taking its initial delay into account) is its first event time. The simulation handles events chronologically. At the event time of a certain train is checked if it can block its next section. If that is the case, then the train's next event time is the instant at which it wants to block the section thereafter. If its next section cannot be reserved, since (part of) it is blocked by another train, its next event time is the end of the blocking time of that other train. The simulation tool performs 10 000 simulation runs. The probability distribution that is used in both Dewilde et al. (2014) and in this paper is the exponential distribution. The exponential distribution is known to be a good approximation of historical delays (Goverde, 1998). Initial delays are generated by a simulation draw. These initial train delays are the delays with which the trains enter the station area. Adding the initial delay to the assigned start times of the trains can cause that two trains that enter the station area along the same track will be conflicting. In that case, a priority rule decides which of the two trains is further delayed (compared to its input delay). These initial delays cause therefore that train trips cannot pass off as planned. Entering the network later than planned can cause that trains encounter red signals. This results in knock-on delays and prolonged real travel times. In the simulation tool knock-on delays are registered for each event when a train cannot reserve its next section when requested. The knock-on delay is then equal to the time that this train has to wait before it can block its next section. The number of passengers that enter, leave or remain in a train in a certain station is assumed to be known in advance for all trains and all stations of the considered network and is assumed to be independent of the timetable. The passenger numbers that are used represent the passenger demand for one hour during the peak hours in the morning, which is assumed to be fixed during this period. Based on this information the passenger robustness of a timetable and routing plan can be calculated in each simulation run. As defined earlier, the passenger robustness is a weighted sum of the travel times in practice over all passengers. The weights reflect how passengers perceive travel time (waiting in a station, driving in a train at full speed, etc.). The weights that are used to calculate the passenger robustness are listed in Table 1 (Dewilde, 2014).

Table 1: Passenger robustness: weights (Dewilde, 2014).

Kind	Weight
Nominal travel time	1
Out delay	3
Exit delay	3
Missed transfer	3
Used supplement	1
Non-used supplement	2

The out delay refers to the delay of passengers that leave a train inside the considered network. The exit delay refers to the delay of passengers that leave the considered network

inside a train. The weight for a missed transfer is applied to the time that passengers have to wait for the next connecting train when they miss a planned transfer. A planned transfer is a transfer that is accounted for during timetable optimization. A used supplement refers to a supplement that a train used to absorb (part of) its delay. A non-used supplement refers to a supplement that appeared to be superfluous, meaning the train was not delayed.

The travel time in practice can easily be calculated for each simulation run, since each simulation run results in a value for the arrival time in the network and on the platforms, the departure time on the platforms and the exit time out of the network for all trains.

The simulation tool differentiates between the planned weighted travel time, the passenger travel time in practice and the nominal travel time. *The planned weighted travel time* is the sum of the weighted travel times of all passengers in ideal circumstances, implying all trains drive according to schedule. The passenger travel time in practice is the sum of the weighted travel times of all passengers in non-ideal circumstances. *The nominal travel time* is the sum of the travel times of all passengers for a schedule in ideal circumstances, without weighing the travel times. This is the technical minimal travel time (running and dwell times) summed for all passengers. The supplements are measured with respect to the technical dwell and running minimal times.

3.3.4 Supplement assignment

We distinguish between two kinds of supplements that can be assigned: dwell time supplements and running time supplements. A dwell time supplement of 7% means that the blocking time of the section that contains the train's platform now contains 1.07 times the technical minimum dwell time of that train in that station instead of just the technical minimum dwell time. If the running time supplement in a grid zone is 7%, then the running time supplement provided to a section in this grid zone, is 7% of the technical minimum blocking time of that section.

To assign new supplements, it is useful to keep track of several criteria during the simulation: the total knock-on delay per train, the total knock-on delay that is incurred per signal, i.e. the total amount of time that trains have to wait at this signal when being halted at this signal because their next section is still blocked, the total knock-on delay that is incurred per node, i.e. the total amount of time that trains have to wait in a section containing this node when being halted at the next signal because their next section is still blocked, the total delay incurred per train per node, i.e. the amount of time that a certain train has to wait in the section that contains that node when being stopped by its next signal and the total amount of non-used supplements. These values are calculated for each of the 10 000 simulation runs and the average for each criterion is reported. Furthermore, the distribution of the delays incurred per train per node over the 10 000 simulation runs and the distribution of used supplements per train per platform area and grid zone over the 10 000 simulation runs are determined.

Some or all of these values can be used in various combinations to determine a better assignment of the supplements. For example, the knock-on delay per train can be used to estimate the amount of supplement to be assigned to each train. The knock-on delay incurred at a certain node or signal can be used to estimate the amount of supplements to be assigned in the platform area or grid zone of that signal or node. After some preliminary tests, we selected the following approach to calculate the next running and dwell time supplements. For each train, the switches and platforms on its route are analyzed in chronological order up to the switch or platform where (i) this train propagates (part of) its delay to the next

train or (ii) this train is delayed for the first time. Both cases are now described in detail.

(i) In case the train propagates (part of) its delay to the next train, we determine the delay that corresponds to the x -th percentile (e.g. x equals 50, 75 or 95) of all simulation runs. We set the supplement to a value that prohibits the delay to propagate in these $x\%$ of the simulation runs. We add the determined value as a dwell time supplement to the dwell time at the train's first platform area with an actual stop. So the train's dwell time in that platform area in the next iteration is the dwell time of the current iteration plus this new dwell time supplement. The dwell time in the current iteration consists of the minimum dwell time and any dwell time supplements that are added as original input or in one of the previous iterations. If a train arrives on its platform with a delay, a dwell time supplement can make sure that the train leaves the station on time or with a smaller delay. Note that a supplement increases the planned occupation rate, but avoids conflicts and so reduces the occupation rate compared to a conflict situation. In case the train is not delayed, the train stands still for the minimum dwell time and its (total) dwell time supplement in that platform area. In our case study, we assume that primary delays are only present when entering the network and do not occur while the trains are already inside the network. So a train that propagates a delay to another train, without first getting a knock-on delay from another train, already suffered from this delay when first entering the considered network. Note that assigning a supplement to the train's first platform area does not avoid delay propagation from this train to other trains in case the delay propagation occurs before the first platform area. However, a supplement may also be assigned to the train that suffers from the knock-on delay, such that the knock-on delay can be immediately absorbed. This is explained in (ii). The advantage of the assigned supplement to the first train is that (part of) its initial delay can already be absorbed in the first platform area without lengthening the blocking times in the (first) grid zone.

(ii) In case a train encounters a delay from another train, we determine the delay that corresponds to the x -th percentile of all simulation runs. We set the supplement to the value that compensates the propagated delay in these $x\%$ of the simulation runs. There is no upper bound restriction. Nevertheless, in the actual results there is no huge increase in the amount of supplements. This can be explained, because in each iteration we give at most one supplement per train and this supplement is based on the train's first secondary delay. So at this point, there is no accumulation of secondary delays yet. If the train encounters this first delay in a station area, we add this value as a dwell time supplement to the dwell time in this station area, otherwise, we add this value to the running time of the train's current grid zone. A running time supplement provides extra time for absorbing a train delay in a grid zone between two platform areas, for example when it is forced to stop at a signal because the next section is not free yet. In case the train is not delayed, it lowers its speed below its normal speed, such that it will not arrive too early on its next platform.

If the determined supplement for the x -th percentile of scenarios is zero, independently of case (i) or (ii), check the next switch or platform on the route of the train until a node in which a delay is encountered or propagated and the determined value for the x -th percentile is not zero.

Until now, only extra supplements are added. Furthermore, the higher x the larger the supplement that will be added and the more difficult it will become to find a feasible timetable in a limited amount of time for very dense networks. Therefore, we will also remove unnecessary supplements. We check for each train and each grid zone and platform area that it crosses, which amount of the provided supplement remains unused in $y\%$ of the

simulation runs. We remove this supplement. The lower y , the more supplement is removed and the less delays can be absorbed. In case y is chosen equal to 100, even supplements that proved useful in only one simulation run, are not removed. In fact, y has to be chosen such that the scenarios in which a delay could be absorbed are weighted against the scenarios in which the supplement is a superfluous lengthening of the passenger travel time. In the case study, x varies over 50, 75 and 95 and y is chosen equal to 95.

3.3.5 Stop criteria

The iterations over these four pillars stop if one of the following stop criteria is satisfied. First, the approach terminates if the supplement assignment converges, i.e. in two succeeding iterations the supplement assignment is equal. This is the first stop criterion. We choose to execute at most ten iterations, which limits the total duration of the algorithm. This is the second stop criterion. We put a time limit on the timetabling phase, since the time needed for the other phases is small compared to the time needed for the construction of an optimal or even a near optimal timetable. This optimization time depends of course on the network characteristics and on the number of trains. The more complex the network or the more trains, the more difficult it becomes to find a feasible timetable. In case of a dense complex network, a feasible solution could be far from optimal. Anyway, for our case study, in case a feasible, let alone an optimal timetable, is found within three hours, the algorithm proceeds with the simulation, otherwise, without finding a feasible solution for this iteration, the algorithm stops. This is the third stop criterion. If in two successive iterations the passenger robustness is worse than the best found value, no further iterations are done. This is undesired but possible nevertheless, since we do not directly optimize passenger robustness. This is the fourth stop criterion.

4 Case study

We use the same case study as in Dewilde (2014); Burggraeve et al. (2015) and Burggraeve and Vansteenwegen (2017). Also the delay scenario that is used to simulate the performance on this case study is adopted from the above research. The entire station area considered in this case study is visualized in Figure 2. An overview of the core of the network (Brussels-South, Brussels-Central and Brussels-North) can be found in Figure 3. The platform areas Brussels-South, Brussels-Central and Brussels-North are the three busiest stations of Belgium.¹ In total the network contains four platform areas (Brussels-South, Brussels-Central, Brussels-North and Brussels-Schaarbeek), with respectively 22, 6, 12 and 12 platforms, and five grid zones. The network contains 481 nodes. During peak hour between 7 a.m. and 8 a.m. 85 trains pass through this station area. The initial supplement assignment is 7% of each minimum running and dwell time. This amount is based on Caimi et al. (2009) and Bešinović et al. (2016a).

Concerning the calculation of the blocking times, like in Burggraeve and Vansteenwegen (2017), we note that the available data does not explicitly contain values for the time for setting up, watching the signal, approaching the section and releasing the signal, that is why we assume that these times are included in the available data. Due to this uncertainty in the data, conflict-freeness at the microscopic level cannot be assured for our case study. However, we can assure that the same data and principles to calculate the lengths of the

¹Source: <https://pvmagazine.nl/brussel-noord-nu-het-drukste-station-van-belgie/>, consulted in July 2016

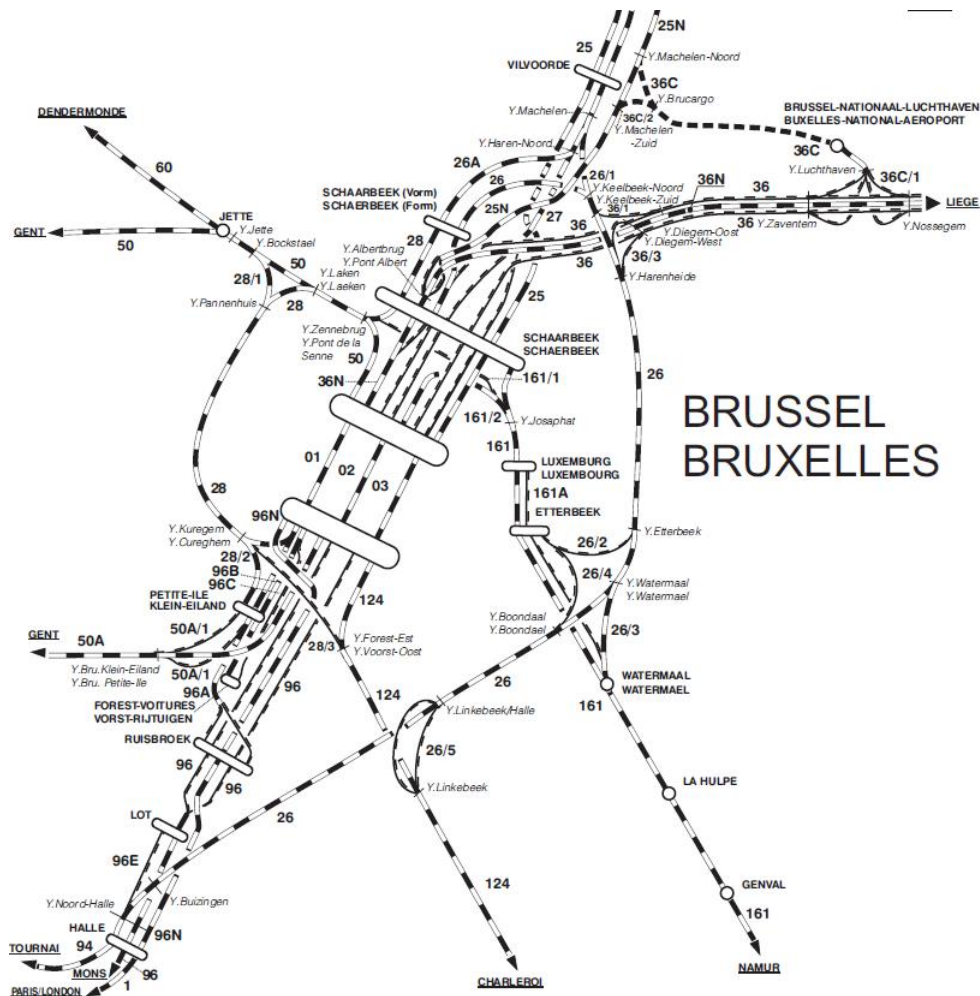


Figure 2: A schematic overview of the entire station area of Brussels. The tracks indicated by 01, 02 and 03 are a simplification of the core of Brussels dense railway area presented in Figure 3.

blocking times are used by the Infrabel and Dewilde et al. (2014) when planning complex station areas, so it still makes sense to compare the resulting routing plans and timetable (cited from Burggraeve and Vansteenwegen (2017)).

For taking the passenger numbers into account in the objective function of the timetabling model, we divided the number of passengers present on trains in six groups, such that $\text{group}(\text{buf}_{t,t',w}) \in \{1, \dots, 6\}$ and the smallest common multiple C equals 60.

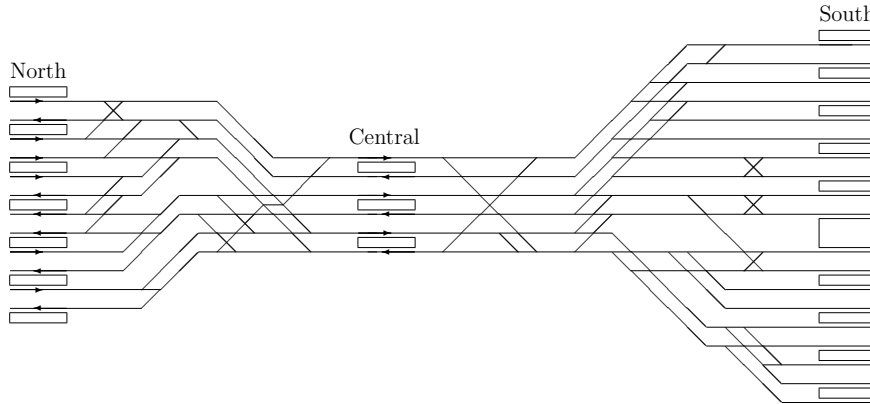


Figure 3: The core of the dense railway area of Brussels.

5 Results and discussion

This section presents the results. We first schedule 40 trains on the network of Brussels, 20 trains driving from Brussels-North to Brussels-South and 20 the other way round. Thereafter we show the results for all 85 trains. In case of 40 trains, the network is not densely occupied, such that trains can more easily be spread. This will have an impact on the results.

The delay scenario that is considered for this research is one in which approximately 50% of the trains are delayed according to an exponential distribution with an average in the interval [0 min, 8 min]. The approximately 50% of the trains and the averages of the exponential distributions are fixed in advance. The average initial delay of the 50% of delayed trains for each timetable of Table 3 equals 3.87 minutes (for the case of 85 trains). The average is different for each train. The remainder of the trains is supposed to be on schedule when entering the network. This delay scenario coincides with the delay scenario used in Burggraeve and Vansteenwegen (2017).

In Table 2 and 3 the passenger robustness, presented in the column '*Passenger robustness*', is the most important performance indicator.

The column '*Impro*' presents the percentage improvement in passenger robustness compared to the outcome constructed with the initial supplement assignment for the case with 40 trains, Table 2. For the 85 trains case, Table 3 gives the percentage improvement compared to the reference timetable and routing plan from the Belgian railway infrastructure manager Infrabel. The column '*Supplements*' presents the amount of supplements present in the timetable and the column '*Non-used supplements*' presents the average amount of supplements that is not used in the 10 000 simulation runs. The column '*Knock-on delays*' shows the average amount of delays that is propagated from one train to the next train over all trains. The next column '*% Delay out*' is the average number of trains that leaves the network with a delay and the column '*% Absorb delay*' gives the average number of trains that absorbs its initial delay before leaving the network. The column '*Deadlocks*' is the number of simulation runs in which two trains are in each other's next section such that they both cannot proceed anymore on their assigned route. This is called a deadlock situation and then the simulation needs to start over again. Only successful runs are counted in the 10 000 simulation runs. It is difficult to interpret the amount of deadlocks since the simu-

lation gives no indication on how much (or how little) extra delay would be necessary to avoid them or if it would be possible to reroute the trains. Furthermore, (i) it can be one or a small number of train pairs that only cause a deadlock in the case of larger delays or (ii) it can be a high number of train pairs that already cause a deadlock in case of a small delay or (iii) it can be a combination. This problem is caused by and only related to the simulation. The problem is that the simulation cannot solve or prevent deadlocks, it can only report them. From preliminary tests, in which deadlocks are prevented with a first come first serve strategy, can be concluded that a high or low number of ‘deadlock runs’ in the simulation, gives no indication about the quality of a given timetable. The results of these tests show that the actual value of the passenger robustness changes when using different strategies to handle deadlocks, but the trend in improvement compared to a reference timetable remains. The last column ‘*Minimum buffer time*’ is the minimum buffer time that is achieved in one hour of optimization time for the case with 40 trains computed with CPLEX 12.6 on an Intel Core i7-5600U CPU @ 2.60 GHz. Note that the restriction of one hour optimization time is only put on the construction of the timetabling, since this phase is by far the most time consuming. For the case with 85 trains, we use three hours of optimization on a compute node Xeon E5-2680v2, IB-QDR, 20 cores with 16 GB memory.

5.1 Results for limited train occupation

We now discuss the results presented in Table 2. To determine the supplements of the next iteration $x = y = 95$ is used (see Section 3.3.4). The algorithm stops after ten iterations, since two stop criteria are then satisfied. On the one hand, the maximum number of iterations is reached, but on the other hand, the passenger robustness deteriorates in each of the last two iterations. The final outcome are the routing plan, timetable and supplement assignment of iteration 8, since there the passenger robustness is the best. During the first six iterations the passenger robustness improves with 10.3%. In the seventh iteration the passenger robustness deteriorates, then it improves again in the eighth iteration and thereafter deteriorates again in the next two iterations. The amount of included supplements and non-used supplements both decrease in the first iteration, but then increase. The average percentage of *used* supplement, equaling 32% ($= \frac{59.8-40.4}{59.8}$) for the initial timetable, lies in between 40% and 45% in the first six iterations and then decreases again. The average knock-on delay decreases first with its lowest value in the fourth iteration. It then increases, but achieves another low value in the eighth iteration. The average percentage of trains that leaves the network with a delay decreases over all the iterations and the average percentage of trains that absorbs their initial delay increases.

These results imply that the passenger robustness can be highly improved by wisely adding supplements. Remember, however, that the network is not densely occupied for this case. Since the minimum buffer time is about the same for all iterations, the improvement can be ascribed to the better amount and assignment of supplements. Also the amount of trains that absorb their initial delay and leave the network with a delay can be highly improved. However, we observe that while the amount of trains that leave the network with a delay keeps improving, this is not the case for the passenger robustness. A possible explanation here could be that we did not take passenger numbers into account while adding supplements. So extra supplements that avoid train delay only have a small effect on the passenger robustness if only a low number of passengers are affected by this delay. Note also that supplements lengthen the travel times of the trains in the network which increases the

Table 2: Simulation results for 40 trains, computed in one hour of optimization time with CPLEX 12.6 on an Intel Core i7-5600U CPU @ 2.60 GHz. *: 12 hours of computation time.

Iter	Passenger robustness ($\cdot 10^6$ min)	Impro (%)	Supplements (min)	Non-used supplements (min)	Minimum buffer time (min)
0	1.16		59.8	40.4	2.7
1	1.12	3.4	55.9	33.1	2.8
2	1.10	5.2	62.2	35.9	2.9
3	1.07	7.6	65.2	37.7	2.8
4	1.05	9.6	66.4	37.6	2.9
5	1.05	9.7	70.1	39.1	2.8
6	1.04	10.3	79.3	46.5	2.8
7	1.04	10.2	89.9	54.1	2.8
8	1.02	12.1	105.4	68.3	2.7
9	1.03	11.8	105.4	67.4	2.4
10	1.06	8.7	112.3	74.7	2.6
8*	1.01	13.2	105.4	68.9	3.1

Iter	Knock-on delays (min)	% Delay out (min)	% Absorb delay (trains)	Deadlocks
0	6.3	29.3	17.0	0
1	6.5	28.3	18.7	238
2	5.0	24.4	21.9	0
3	3.9	21.6	23.9	0
4	2.9	19.5	25.2	493
5	4.4	19.7	26.9	771
6	4.9	18.2	27.8	0
7	8.3	17.8	29.0	0
8	3.7	13.8	31.7	170
9	6.1	14.4	31.6	258
10	3.4	13.2	32.5	581
8*	2.5	13.3	31.2	0

occupation of the infrastructure. If supplements are well placed, then this higher occupation of the network does not increase the probability on knock-on delays in the network. This is because a train can immediately absorb its delay in case a supplement is provided, such that this delay will not be propagated. Knock-on delays are only caused if the delay is larger than the supplement and the subsequent buffer time.

The last row of Table 2 presents the simulation results for iteration 8, where we use a computation time of 12 hours instead of only one hour for constructing an optimal timetable.

This is indicated as iteration 8*. We see that the minimum buffer time in the timetable improves to 3.1 minutes and that the passenger robustness improves even further (12.1% to 13.2%). Note that all other performance indicators also improve, except for the average percentage of trains that absorbs its initial delay. This difference in performance between the timetable for iteration 8 and 8* because of a difference in computation time is discussed further on.

5.2 Results for dense train occupation

Table 3 shows the simulation results of the here presented approach applied to the case where all 85 trains are planned. The performance is compared to the performance of the routing plan and timetable of Dewilde (2014), presented in row '*Dewilde et al.*', the performance of the Belgian railway infrastructure manager Infrabel, presented in row '*Infrabel*', and the performance of the routing plan and timetable constructed by the routing and timetable model described in Burggraeve and Vansteenwegen (2017), presented in row '*Burggraeve et al.*' The latter routing plan and timetable differ from the routing plan and timetable constructed in the first iteration ('Iter 0') of the approach by the changes made to these optimization models, see Section 3.3. The routing model now includes a feasibility check and the timetabling model takes passenger numbers into account. The parameters of Section 3.3.4 can be interpreted as follows. The higher x , the more simulation runs in which delays early in the network can be avoided, since the amount of supplements matches or exceeds the amount of delays in more simulation runs. The higher y , the less supplements that proved useful in some simulation runs, are removed. As for the case with 40 trains, we first used $x = y = 95$. The resulting passenger robustness of the first iteration was incredibly good, see row '*95% - 1*'. It improved the passenger travel time in practice of Infrabel with over 17%. However, during the second iteration, no feasible timetable could be determined anymore within a three hour computation period, given the added supplements. For this timetable, we calculated for each train the ratio of its supplements and its technical minimum blocking time of the sections on its route. On average this gives 8.58% with a minimum of 3.30% and a maximum of 41.72%. This maximum is very high, but can be explained because this train terminates its trip in Brussels-South and is thereafter reused for a next trip. As a result the approach provides a lot of supplement in this trip to assure that the next trip of this train can punctually take off. The maximum percentage for trains that do not take part in a splitting or re-usage is 20.60%. Furthermore, the amount of dwell time supplement in platform blocking times (running time supplement in grid zones) is on average 7.18% (9.88%), with a minimum of 0% (0%) and a maximum of 19.58% (31.45%), where the trains that take part in a re-usage of splitting are omitted for determining this maximum. For $x = 75$ and $y = 95$, the passenger robustness in the first iteration deteriorates, but improves in the second iteration, see row '*75% - 1*' and '*75% - 2*'. The third iteration is however not feasible anymore within three hours of computation time. For $x = 50$ and $y = 95$, the passenger robustness improves in the first iteration, but then deteriorates in the next two iterations, see row '*50% - 1*', '*50% - 2*' and '*50% - 3*'. The minimum buffer time in Table 3 is approximately zero for all the presented timetables. Note however that (passenger) robustness is a result of the interaction of the amount of passengers on the trains, the routes of the trains, the buffer times and the supplements. Such that the appearance of a zero buffer time cannot be interpreted on its own.

Table 3: Simulation results for 85 trains, computed in three hours of optimization time with CPLEX 12.6 on a compute node Xeon E5-2680v2, IB-QDR, 20 cores with 16 GB memory.

Iter	Passenger robustness ($\cdot 10^6$ min)	Impro (%)	Supplements (min)	Non-used supplements (min)	Minimum buffer time (min)
Infrabel	2.89		203	122	0.0
Dewilde et al.	2.61	9.4	203	140	0.0
Burggraeve et al.	2.56	11.4	206	105	0.0
0	2.66	8.0	123	61	0.0
95% - 1	2.38	17.6	177	100	0.0
75% - 1	2.84	1.5	133	57	0.0
75% - 2	2.63	8.9	170	83	0.0
50% - 1	2.57	10.9	118	54	0.0
50% - 2	2.69	6.7	127	53	0.0
50% - 3	3.05	-	128	51	0.0

Iter	Knock-on delays (min)	% Delay out (trains)	% Absorb delays (trains)	Deadlocks
Infrabel	150	60.2	9.7	1 687
Dewilde et al.	87	49.2	12.8	5 174
Burggraeve et al.	117	43.0	18.3	6 796
0	105	55.7	10.1	39 377
95% - 1	83	48.2	15.6	26 651
75% - 1	149	60.9	10.6	17 635
75% - 2	128	53.9	14.0	22 140
50% - 1	102	54.5	11.6	17 797
50% - 2	120	55.1	12.7	40 698
50% - 3	178	57.0	12.7	31 955

Figure 4 shows the blocking times in Brussels-Central station for the reference timetable from Infrabel and Figure 5 for the reference timetable from Dewilde (2014). The labels on the x -axis are expressed in minutes. The y -axis labels the six platforms of Brussels-Central. Each train has its own color and the train number is presented inside the colored rectangle that presents its blocking time. Since each train passes at most once in Brussels-Central per period, at most one blocking time per train is shown.

Observe that only one train in the timetable from Dewilde (2014) has a different platform in Brussels-Central compared to the timetable from Infrabel (train 63). Furthermore, there are several trains that changed order in the timetable from Dewilde (2014) compared to the timetable from Infrabel, for example train 56 and 61.

In Figure 6, the blocking times in Brussels-Central station are shown for the timetable of row '95% - 1' in Table 3. The platforms are used 11 up to 15 times. None of the blocking times overlap, so the constructed timetable is conflict-free on the signaling level.

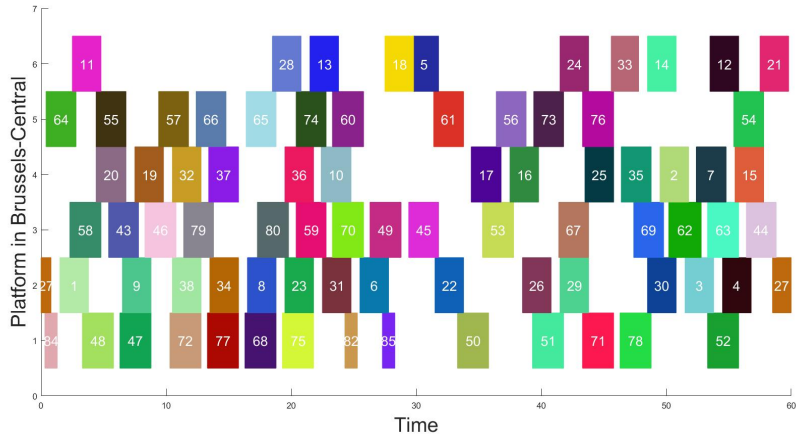


Figure 4: The blocking times of the reference timetable of the Belgian railway infrastructure manager Infrabel for all trains passing Brussels-Central during morning peak hour in between 7 a.m. and 8 a.m.

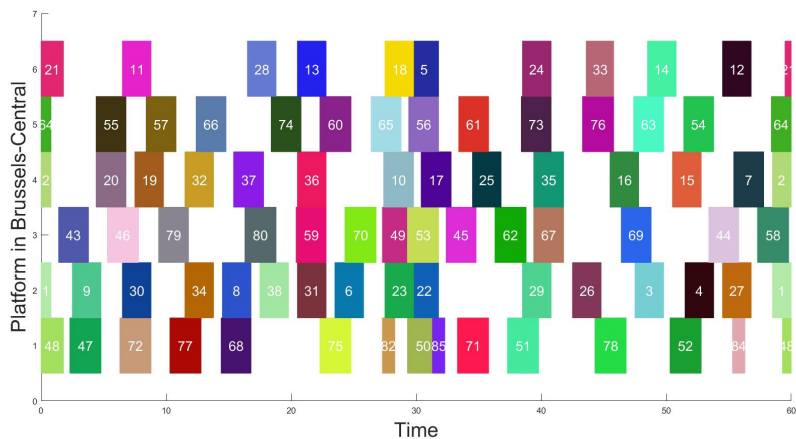


Figure 5: The blocking times of the timetable of Dewilde (2014) for all trains passing Brussels-Central during morning peak hour in between 7 a.m. and 8 a.m.

Furthermore, we observe that between most blocking times there is a positive buffer time.

At first sight, the timetables from Infrabel and Dewilde et al. (2014) seem to have a better spreading in Brussels-Central. However, an important remark to make is that these timetables are not entirely cyclic. This cannot clearly be observed from the above diagrams, but it can be noticed when evaluating other nodes in the network. Our timetable is entirely cyclic and it is more robust for the whole station area.

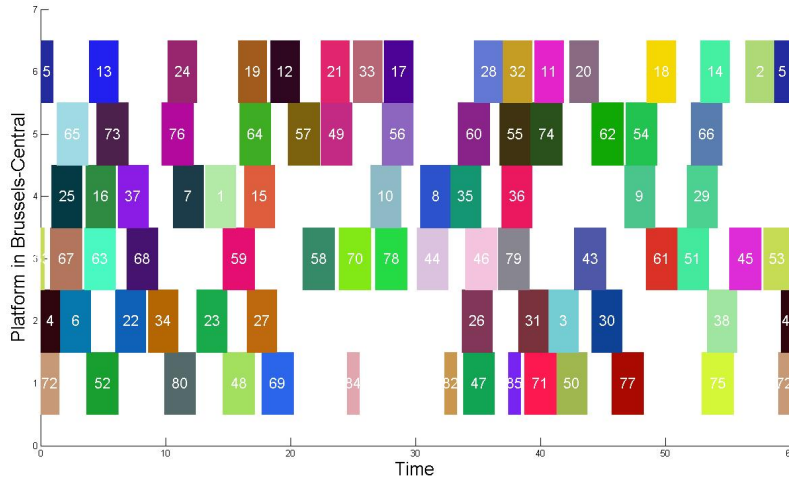


Figure 6: The blocking times of the timetable of row ‘95% - 1’ for all trains passing Brussels-Central in between 7 and 8 a.m. during morning peak hour.

5.3 Discussion

For the case with 40 trains, where the occupation rate is only moderate, we observe that the passenger robustness can be highly improved by our proposed method, up to 13.2% compared to an initial supplement assignment corresponding to common practice. Note that the timetable of the first iteration contains less supplements than the initial timetable and a 3.4% better value for the passenger travel time in practice. This shows that a good implementation of supplements is worthwhile. Furthermore, we see that the minimum buffer time over all iterations is more or less equal, so the improvement in passenger robustness is due to the smart allocation of supplements.

For the case with 85 trains, we also observe that a smart allocation of supplements can highly improve the passenger robustness. However, due to the restriction on the computation time of the timetable, the advantage of a better supplement allocation is counteracted by a bad buffer time assignment. Either the timetable optimization is broken off too early or a feasible timetable is not found. One solution would be to extend the time limit. Another option is to replace the exact timetabling model with a fast heuristic to overcome this problem. Nevertheless, the algorithm constructed a timetable and routing plan that improved the passenger robustness of a reference planning from practice with up to 17.6%.

To do a thorough analysis of the performance of the algorithm, the output timetable may not depend on the whims of the optimization process in the allowed time. So we propose to use a heuristic for constructing the timetable. This heuristic must have the same objective, but must be faster and the quality of the output timetable must be less sensitive in the available time. It would also be interesting to have some error estimation on how far the constructed planning is from the real optimal solution. Even if for a complex network with a high occupation rate, the optimization gap for the timetable construction remains large, it would be interesting to have an idea on how far the best found feasible solution and the

upper bound are from the real optimum.

Furthermore, to assure the same performance on other networks, further research must consist of an efficiency and convergence analysis. Realistically, this analysis is only achievable if the computation time for the timetable construction is reduced without reducing the solution quality. However, there are lots of interesting ideas to include in this efficiency and convergence analysis. For example, the assignment of supplements for the next iteration in the algorithm could be based on the simulation output of more than one delay scenario in case no historical delays are available. Some restrictions on the addition or removal of supplements could be inserted, for example the amount of supplement may increase with at most 10% compared to the previous iteration, the addition or removal of supplement in the previous iteration is tabu in the next iteration, etc. The supplement assignment can also be made dependent on the amount of passengers which are affected. It would also be useful to choose $x(y)$ in Section 3.3.4 based on a substantiated assessment of propagated delays (non-used supplement) in some scenarios and no delay propagation (useful supplements) in the other scenarios.

6 Conclusion

This paper proposes an iterative approach to construct a railway routing plan and timetable from scratch with a smart allocation of buffer times and supplements. In each iteration the supplement assignment is refined based on simulation results and the buffer times are optimized in an exact optimization model. The results show that the passenger robustness can be highly improved. However, for very dense and complex networks, a time constraint to construct a well performing timetable can counteract the positive effects of a better supplement assignment. The timetable quality has to be weighted against the computation time.

7 Acknowledgements

This research was supported in part through computational resources provided by the KU Leuven high performance cluster. We thank the Belgian railway infrastructure manager Infrabel for their commitment during the research and for providing data.

References

- Bešinović, N., Goverde, R.M.P., Quaglietta, E., Roberti, R., 2016a. ‘An integrated micro-macro approach to robust railway timetabling’, *Transportation Research Part B: Methodological*, vol. 87, pp. 14–32.
- Bešinović, N., Goverde, R.M.P., Quaglietta, E., 2017. ‘Microscopic models and network transformations for automated railway traffic planning’, *Computer-Aided Civil and Infrastructure Engineering*, vol. 2, pp. 89–106.
- Burggraeve, S., Dewilde, T., Sels, P., Vansteenwegen, P., 2015. ‘Improving passenger robustness by taking passenger numbers and recurring delays explicitly into account on the tactical level’, In: *Proceedings of the 6th International Seminar on Railway Operations Modelling and Analysis (RailTokyo2015)*, Tokyo, Japan.
- Burggraeve, S., Vansteenwegen, P., 2017. ‘Robust routing and timetabling from scratch in complex railway stations’, *Transportation Research Part B: Methodological*, vol. 101, pp. 228-244.

- Burggraeve, S., Bull, S., Vansteenwegen, P., Lusby, R.M., 2017. 'Integrating robust timetabling in line plan optimization for railway systems', *Transportation Research Part C: Emerging Technologies*, vol. 77, pp. 134–160.
- Cacchiani, V., Caprara, A., 2012. 'Nominal and robust train timetabling problems', *European Journal of Operational Research*, vol. 219, pp. 727–737.
- Cacchiani, V., Furini, F., Kidd, M.P., 2016. 'Approaches to a real-world Train Timetabling Problem in a railway node', *Omega*, vol. 58, pp. 97–110.
- Caimi, G., Burkolter, D., Herrmann, T., Chudak, F., Laumanns, M., 2009. 'Design of a new railway scheduling model for dense services', *Networks and Spatial Economics*, vol. 9, pp. 25–46.
- Caprara, A., Fischetti, M., Toth, P., 2002. 'Modelling and solving the train timetabling problem', *Operations Research*, vol. 50, pp. 851–861.
- De Fabris, S., Longo, G., Medeossi, G., Pesenti, R., 2013. 'Application and validation of a timetabling algorithm to a large Italian network', In: *Proceedings of the 5th International Seminar on Railway Operations Modelling and Analysis (RailCopenhagen2013)*, Copenhagen, Denmark.
- 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., 2013. 'Robust railway station planning: An interaction between routing, timetabling and platforming', *Journal of Rail Transport Planning & Management*, vol. 3, pp. 68–77.
- Dewilde, T., Sels, P., Cattrysse, D., Vansteenwegen, P., 2014. 'Improving the robustness in railway station areas', *European Journal of Operational Research*, vol. 235, pp. 276–286.
- Dewilde, T., 2014. 'Improving the robustness of a railway system in large and complex station areas', *Doctoral dissertation*, KU Leuven, Belgium.
- Goverde, Rob, 1998. 'Optimal Scheduling of Connections in Railway Systems', *Proceedings of the 8th WCTR, Antwerp, Belgium*.
- Khan, M.B., Zhou, X., 2010. 'Stochastic optimization model and solution algorithm for robust double-track train-timetabling problem', *Transactions on Intelligent Transportation Systems*, vol. 11(1), pp. 81–89.
- Kroon, L.G., Dekker, R., Vromans, M.J.C.M., 2008. 'Cyclic Railway Timetabling: A Stochastic Optimization Approach', *Algorithmic Methods for Railway Optimization*, vol. 4359 of the series of Lecture Notes in Computer Science, pp. 41–66.
- Liebchen, C., Möhring, R., 2007. 'The modeling power of the periodic event scheduling problem: Railway timetables - and beyond, Algorithmic methods for railway optimization', In: Geraerts, F., Kroon, L., Schöbel, A., Wagner, D., Zaroliagis, D., *Lecture Notes in Computer Science*, vol. 4359, pp. 3–40, Springer Berlin Heidelberg.
- Niu, Y., Meng, L., 2014. 'Optimizing slack time allocation in train timetable: A two-stage stochastic recourse model', In: *Proceedings of the 2013 International Conference on Electrical and Information Technologies for Rail Transportation (EITRT2013)*, pp. 245–252.
- Pachl, J., 2008. 'Timetable design principles', In: *Railway timetable & traffic: Analysis, modelling and simulation*, Ed.: Hansen, I. and Pachl, J., Eurailpress, Hamburg, pp. 9–42.
- Rudolph, R., 2003. 'Allowances and margins in railway scheduling', In: *Proceedings of WCCR 2003*, pp. 230–238, Edinburgh.
- Schlechte, T., Borndorfer, R., Erol, B., Graffagnino, T., Swarat, E., 2011. 'Micro-macro

- transformation of railway networks', *Journal of Rail Transport Planning and Management*, vol. 1, pp. 38–48.
- Schmidt, M., Schöbel, A., 2015. 'Timetabling with passenger routing', *OR Spectrum*, vol. 37, pp. 75–97.
- Sels, P. and Dewilde, T. and Cattrysse, D. and Vansteenwegen, P., 2016. 'Reducing the passenger travel time in practice by the automated construction of a robust railway timetable', *Transportation Research Part B: Methodological*, vol. 84, pp. 124–156.
- Serafini, P., Ukovich, W., 1989. 'A mathematical model for periodic scheduling problems', *SIAM Journal on Discrete Mathematics*, vol. 2, pp. 550–581.
- Yuan, J., Hansen, I.A., 2007. 'Optimizing capacity utilization of stations by estimating knock-on train delays', *Transportation Research Part B: Methodological*, vol. 41, pp. 202–217.