Integrated Decision Support Tools for Disruption Management

Nikola Besinovic [a], Valentina Cacchiani [b], Twan Dollevoet [c], Rob M.P. Goverde [a], Dennis Huisman [c], Martin P. Kidd [b], Leo G. Kroon [d], Egidio Quaglietta [a], Joaquin Rodriguez [e], Paolo Toth [b], Lucas Veelenturf [f,1], Joris Wagenaar [d]

[a] Department of Transport and Planning, Delft University of Technology
P.O. Box 5048, 2600 GA Delft, The Netherlands

[b] Department of Electrical, Electronic, and Information Engineering "Guglielmo Marconi", University of Bologna,
Viale Risorgimento 2, 40136 Bologna, Italy

[c] Econometric Institute and ECOPT, Erasmus University Rotterdam
P.O. Box 1738, 3000 DR Rotterdam, The Netherlands

[d] Rotterdam School of Management, Erasmus University
P.O. Box 1738, 3000 DR Rotterdam, The Netherlands

[e] University of Lille Nord de France, IFSTTAR
P.O. Box 70317, 59666 Villeneuve d'Asq, France

[f] School of Industrial Engineering, Eindhoven University of Technology
P.O. Box 513, 5600 MB Eindhoven, The Netherlands

[1] Corresponding author. E-mail: l.p.veelenturf@tue.nl

Abstract
During railway operations unexpected events can require railway operators and infrastructure managers to adjust their schedules. In this research we investigate the disruption management process. More specifically, we come up with an architecture and algorithmic framework which railway operators could use for decision support during disruptions. The use of this framework results in a fully feasible timetable, rolling stock plan, and crew schedule to deal with the disruption, while minimizing the number of delayed and/or (partially) cancelled trains. We demonstrate the effectiveness of our framework on a disruption case on the Dutch Railway network, which is introduced within the EU FP7 project ON-TIME.

Keywords
Disruption Management, Railway Operations, Algorithmic Framework, Rescheduling.

1 Introduction

Railway systems face many unexpected events during the daily operations. Examples of such events are crew unavailability, train failures in a station, broken catenaries, strikes, or bad weather. We refer to an unexpected event as a disruption, if it renders any of the resource schedules (infrastructure, rolling stock, and/or crew) infeasible. Whenever a disruption occurs, the resource schedules should be adapted to the disrupted situation as quickly as possible. The goal of the disruption management process is to maintain the highest possible level of service for the passengers and freight carriers. In particular, the primary aim is to minimize the number of train services that have to be cancelled.

Many recovery models for rescheduling a single resource have been proposed in recent literature (see Cacchiani et al. [3] for a recent overview). Within the EU FP7 project ON-TIME (Optimal Network for Train Integration Management across Europe, see ON-TIME 2014 [6]), we have developed an integrated framework that reschedules all resources at once. Within ON-TIME, we focus on disruptions that block all tracks between two stations. This case occurs, for example, when catenaries break or when an accident happens on a level crossing. In these situations, the catenaries need to be repaired or the accident must be investigated. In both cases, the tracks will be blocked for a certain period of time. As a consequence, the timetable cannot be executed as planned, because trains that were supposed to run over these tracks need to be cancelled, short-turned, retimed, or rerouted. Especially short-turning or cancelling trains might render the rolling stock and crew schedule infeasible, too. If this is the case, the rolling stock and crew schedules need to be adapted as well.

In this paper, we describe the disruption management tool (DMT), developed within ON-TIME, which integrates the rescheduling of the timetable, the rolling stock, and the crew. The core of the tool is an iterative algorithm that first computes a timetable, and then determines a feasible rolling stock and crew schedule. The timetable is computed with a macroscopic optimization approach, using input data that is computed by a microscopic model. The DMT can be decomposed into the following four modules:

1) Macroscopic timetable module
2) Microscopic timetable module
3) Rolling stock rescheduling module
4) Crew rescheduling module

The iterative framework has been tested on a typical instance from the Netherlands. The experimental results show that the disruption management tool computes feasible resource schedules in a couple of minutes. Such solution times are acceptable in practice. This shows that the proposed iterative algorithm can be applied in a practical setting for the disruption management process.

The remainder of this paper is structured as follows. In Section 2, we discuss the iterative solution approach in more detail and show how it can be embedded in a real-world disruption management system. Section 3 presents a case study on a typical instance from Netherlands Railways. In Section 4, we draw conclusions and present possible directions for future research.

2 Framework

The main focus of the current paper is the disruption management tool (DMT). Nevertheless, we now first describe how the tool could be embedded in a disruption management system. This description also includes how all relevant input can be gathered from the field and how the decisions can be implemented, given the real-time nature of its application. We have developed all modules in the framework and specified their interfaces, but have not yet fully tested the complete integration of all modules.

The framework we propose for managing railway traffic disruptions in real-time is illustrated in Figure 1. The realistic simulation model HERMES is used as a surrogate of the real railway traffic. This is a microscopic environment which reproduces all the events

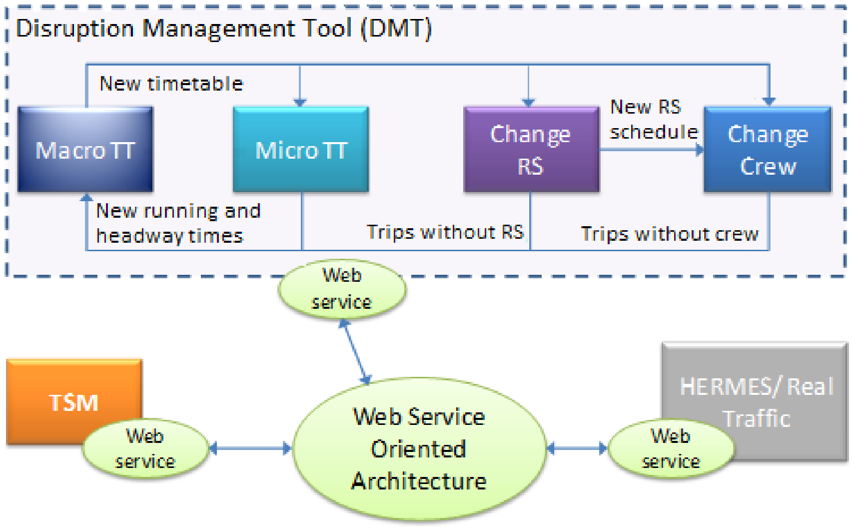

Figure 1: Proposed framework for the optimal real-time management of railway traffic disruptions.

and the interactions among trains, infrastructure, signalling and interlocking, typical of real railway systems. The Traffic State Monitoring (TSM) continuously collects traffic information deriving from HERMES such as track occupation/release events. This information is necessary to estimate the current traffic state, i.e. position and speed of every train on the network. In case a disruption event occurs during the simulation (e.g. the failure of a signal or a track) this event is sent together with the estimated current traffic state to the DMT.

The DMT is the core of the framework since it automatically computes a new emergency timetable and corresponding rolling stock and crew schedules which minimize the impact of the disruption on traffic performances. The DMT is composed of four sub-modules which iteratively interact among each other to produce a timetable which is conflict-free at the level of block sections and respects rolling stock and crew constraints. The current traffic state produced by the TSM is used by the Microscopic timetable module (*MicroTT*) to determine accurate train running times and headway times. Both running and headway times are used as input to the macroscopic timetable module (*MacroTT*). This macroscopic model identifies the timetable which minimizes the number of (partially) cancelled and delayed trains. It considers the capacity of the infrastructure and the number of available rolling stock units. The produced timetable is conflict-free at the macroscopic level but it is not said that it is also conflict-free at the level of block sections. For this reason this timetable is transferred to the *MicroTT* which verifies the microscopic feasibility of the macroscopic timetable, that is to say the absence of track conflicts at the level of block sections. The iterations between *MicroTT* and *MacroTT* only stop after the timetable has been proved to be feasible at both macroscopic and microscopic levels. The optimal and feasible timetable is then sent as input to the Rolling Stock (*Change RS*) and the Crew (*Change Crew*) rescheduling modules. These two modules ensure that rolling stock and crew are available for each scheduled train service. At this stage we assume that the timetable remains feasible if we (partially) cancel a train service in the timetable if no rolling stock or crew is available for that (part of the) train service. The *Change RS* module assigns the rolling stock to the train services. All (parts of the) train services for which no rolling stock is found are cancelled before the framework continues with *Change Crew*. If *Change Crew* cancels a train service because of unsatisfied crew constraints, the rolling stock schedule might consequently become infeasible. In this case the rolling stock and crew schedule must be adjusted by means of an iterative process that stops only after a mutually feasible solution is found. The final output of the DMT is therefore a new timetable which is conflict-free at the level of block sections and respects rolling stock and crew constraints. This timetable is implemented into the simulation environment for the whole duration of the disruption. Train services will therefore operate according to this timetable until the disruption ends. The TSM, the DMT and the simulation environment HERMES exchange input/output data among each other by means of a web service-oriented architecture. The architecture is flexible and scalable and builds on a standard RailML communication interface.

2.1 Architecture

The web service-oriented architecture enables the communication among the different modules of the framework. This is an event-based architecture where the data to transfer from a module to another are considered as events. The architecture identifies every event by means of a unique "type" identifier. Each module of the framework can be linked to the architecture as a "subscriber" or "publisher" for one or more types of events. A module is a publisher for a given event type when it produces that event as output. Instead it is a subscriber when it needs to receive that type of event as input. If a module is subscriber to a given event type it cannot receive events of different types unless it is also subscribed to them. We give a simple example to clarify the concept. The current traffic state produced by the TSM is seen by the architecture as an event of a given type, say type A. In this case the TSM is a publisher for the event type A. The DMT instead is a subscriber for type A events since it takes the current traffic state as input. If an event of a given type cannot be dispatched to one or more of its subscribers, it is stored in a queue and transferred as soon as this is possible. The storage is realized by using the open-source massage broker Rabbit MQ [8]. The architecture is scalable since it works independently of the number of modules connected and the amount of data exchanged. It is moreover flexible given that it allows replacement of a module with another with similar characteristics and interfaces. The main characteristic is that it is standardized by using RailML to express both static and dynamic railway data. This means that all the modules exchange input/output data expressed in standard RailML. Such a feature allows immediate applicability of the framework to any network whose data are expressed in RailML format.

2.2 Modules

In this section we introduce the models we have used in the modules that compose the disruption management tool. However, since our approach is modular, for each module it is possible to replace the mathematical model or algorithm behind it. This means that the use of the disruption management tool is not restricted to the models discussed below.

Macroscopic Timetabling

The algorithm used for macroscopic timetable rescheduling is the one introduced by Veelenturf et al. [9]. This algorithm consists of solving a mixed integer programming (MIP) formulation by a general purpose solver. In our experiments this solver is CPLEX 12.5. The MIP formulation is based on an event activity network. It models the capacity within stations and between stations by means of the number of available tracks. Details about switches and signals are neglected. Furthermore, the formulation ensures that a feasible rolling stock circulation exists for the resulting timetable. For our framework, it means that using this model ensures that in the rolling stock rescheduling module we will not have additional cancellations due to lack of rolling stock. The algorithm is able to find the timetable which (partially) cancels and delays as few train services as possible. The advantage of using such an automated approach instead of the current practice of contingency plans is that it can handle all situations, where contingency plans are only available for a limited set of scenarios. In addition, the contingency plans do not consider the delays of the trains at the time the disruption starts.

Microscopic Timetabling

The microscopic model for timetabling consists of several algorithms that: a) convert macroscopic timetable input to the microscopic level, b) compute occupation times for each train, c) evaluate the conflict-freeness of the macroscopic timetable, and d) perform rerouting in stations where necessary (i.e., in short-turning stations). The first three models are established by Besinovic et al. [2], while the latest is newly developed. The microscopic timetable incorporates a high level of detail: infrastructure is modelled on a microscopic level including static speed limits, section lengths and gradient changes. Also, the detailed rolling stock characteristics comprise engine power, resistance forces, mass, and length that are necessary to accurately compute train trajectories. The input for the microscopic timetabling module is a macroscopic timetable, i.e. the event times computed by the macroscopic timetable rescheduling algorithm.

First, we perform the computation of operational speed profiles based on the event times from the macroscopic timetabling module. The operational running time is defined as the running time scheduled by the macroscopic timetable between two infrastructure points (e.g. stations or junctions). This model represents an improvement of the running time computation model defined in [1]. The associated running time supplements are exploited by applying cruising with a speed lower than the maximum speed and to do so, we implemented the customised bisection algorithm. Afterwards, the infrastructure occupation for each train is computed using the well-established blocking time theory [4].

Before the conflict detection analysis, it is necessary to perform the local rerouting in short-turning stations. This is needed because a planned platforming schedule is most likely to be infeasible when the concept of short-turning is introduced. Therefore, we need to assign new platforms and the corresponding routes for trains. For this purpose, a simple heuristic approach is developed to mimic the dispatchers' decisions. For example, a departure (arrival) platform of a train in the short-turning station corresponds to the scheduled platform for the train of the same line running in the opposite direction in the regular timetable plan (i.e., in the timetable with no disruptions). This aim of the rerouting algorithm is to minimize necessary changes in a station.

Once the train trajectories are computed and blocking times are obtained, we execute the conflict detection and resolution algorithm. The aim of the conflict detection (CD) is to verify the feasibility of the macroscopic timetable by checking the existence of conflicts between trains. A track conflict occurs when blocking times of two or more trains at the same track section partially or fully overlap for a period of time. The CD is conducted over the complete infrastructure including both stations and open track lines (i.e., nodes and arcs of the macroscopic network model), and for all the trains running in the network. The conflict resolution (CR) part computes for each pair of conflicting trains new minimum headway times that satisfy safety constraints and ensure conflict-free operations.

The output of the microscopic timetabling module is: a) new (locally corrected) train routes with computed running times and b) updated headway times.

Rolling stock Rescheduling

The model introduced by Nielsen et al. [5] is used for rescheduling the rolling stock in our framework. The problem is formulated as a multi-commodity flow model, where the nodes in the graph are represented by stations at specific times, and the arcs correspond to the trips that have to be performed in the timetable. Individual rolling stock units can be coupled to each other to form a composition. In this way, compositions can be appointed to a trip, such that there is enough capacity to cover passenger demand.

Several constraints are taken into account to guarantee a feasible rolling stock circulation. The first set of constraints ensures that exactly one composition is appointed to every trip. A so-called empty composition is introduced that represents the possibility to cancel a trip. Secondly, constraints are used to handle composition changes. At stations with a shunting yard, it is allowed to couple or uncouple units from a composition. So, constraints are required to handle this correctly. Furthermore, constraints are required to keep track of the inventory of each train unit type at stations. This inventory may never be negative. Then, constraints are used to determine the end of day balance of each rolling stock type at a station. Finally, constraints are used to ensure that rolling stock compositions on trips that have already started at the time of rescheduling are not changed.

The objective is to minimize the costs and to maximize the passenger service. Penalties are put on additional shunting movements, deviations from the end of day balance, capacity shortage, carriage kilometres, and the largest penalty is set for trips to which no composition is assigned. If no composition could be assigned to a trip it means that the trip has to be cancelled. The problem is solved with a rolling horizon approach and by using the general purpose solver CPLEX 12.5.

Crew Rescheduling

Crew rescheduling is solved by a heuristic proposed by Potthoff et al. [7]. The crew rescheduling problem is formulated as a set covering problem and solved by a heuristic based on a combination of column generation and Lagrangian relaxation. No commercial solver is necessary. The idea of the formulation is to assign duties to crew members. While assigning the duties to the crew members, the aim is to ensure that every task has at least one crew member assigned to it. If not, the train has to be cancelled. As a side objective the new duty of a crew member should not deviate too much from his original duty. For example the new duty should not take much longer than the original duty. Depending on the labour rules, it can even be imposed that the new duty cannot end later than the original duty. Other labour rules, like meal breaks and transfer times are considered as well. Column generation is used to generate promising duties that should be added to the formulation. Adding all possible duties for a crew member in the model beforehand is impossible because of the huge number of duties that are possible.

3 Case Study

We have tested the disruption management tool on a typical case from the Netherlands. We have obtained infrastructure data from ProRail and a timetable with the associated rolling stock and crew schedules from Netherlands Railways (NS).

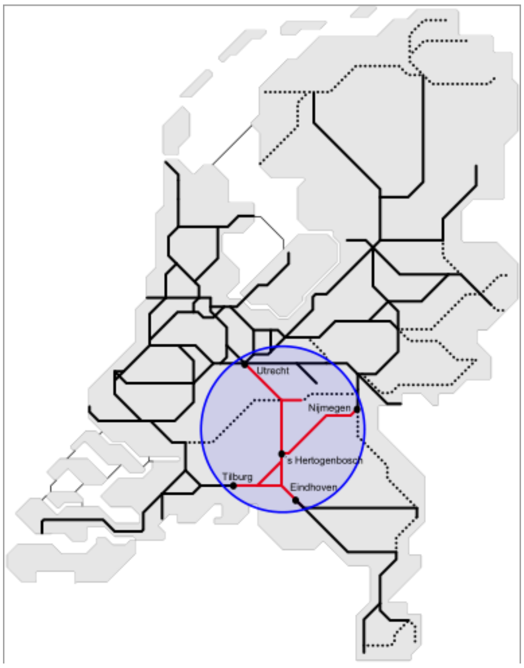

Figure 2: Railway network in the Netherlands

3.1 Case description

A picture of the railway network in the Netherlands can be found in Figure 2. We consider a disruption between ‘s-Hertogenbosch and Oss that lasts from 6:30 until 8:30 in the morning. During the disruption, both tracks are blocked. The station of Oss is halfway along the line from Nijmegen to ‘s-Hertogenbosch. Between Oss and Nijmegen, there is a bridge with only one track and several minor stations. There are double tracks between Oss and ‘s-Hertogenbosch with some minor stations.

In the planned timetable, two train lines run over the tracks between ‘s-Hertogenbosch and Oss. The first one consists of regional trains that run from ‘s-Hertogenbosch to Nijmegen and vice versa. The second one consists of long distance trains, running from the South-West to the North-East and vice versa, stopping at the stations of Tilburg, ‘s-Hertogenbosch, Oss, and Nijmegen.

3.2 Timetable rescheduling

We start the solution process with the timetabling modules. For the timetabling modules, we consider only that part of the railway network that is enclosed within the circle in the picture. For the rolling stock and the crew rescheduling modules, we consider the complete network. First, the microscopic timetable module computes running and headway times based on the planned timetable. These parameters are used by the macroscopic timetable module to compute a timetable that is macroscopically feasible and takes the disruption into account. In order to find such a timetable, it is possible to cancel, retime, or short-turn trains.

The macroscopic timetable module decides to short-turn the long distance trains in Oss and in ‘s-Hertogenbosch. This means that trains arriving from Tilburg in ‘s-Hertogenbosch terminate there and return into the direction of Tilburg. Similarly, at Oss, trains arriving

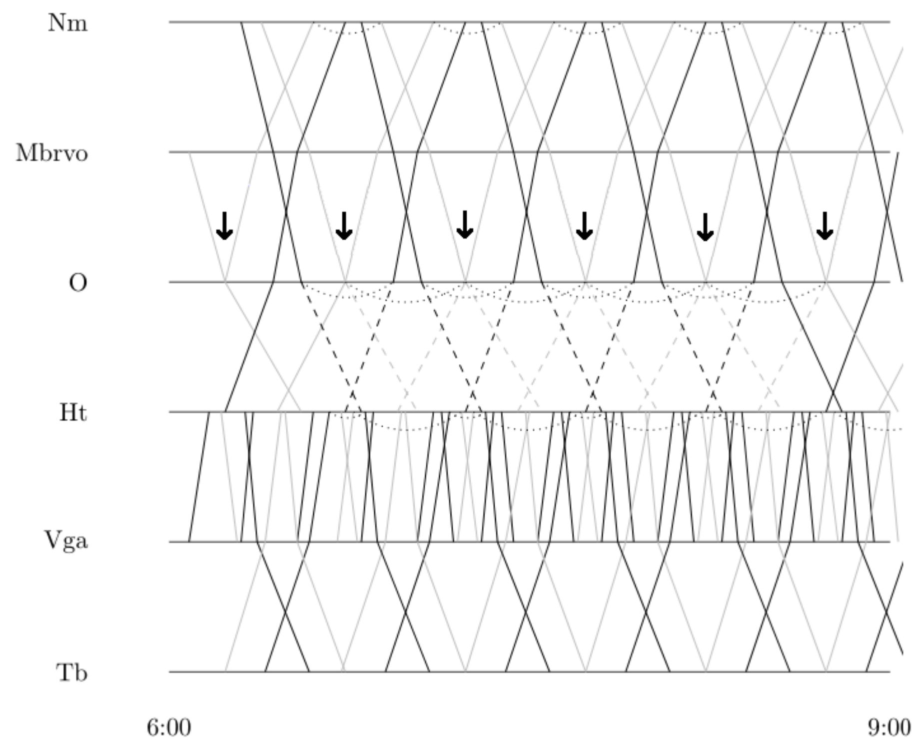

Figure 3: Time space diagram of the macroscopic timetable

from Nijmegen are short-turned and return in the direction of Nijmegen. The regional trains, that were supposed to run from ‘s-Hertogenbosch to Nijmegen, can only run between Oss and Nijmegen. They are also short-turned in Oss. The parts of the train lines between ‘s-Hertogenbosch and Oss are cancelled during the disruption. In total, 4 long distance trains and 4 regional trains per direction are partially cancelled.

The macroscopic timetable module also takes into account the availability of rolling stock. There is no shunting yard at Oss, so all trains arriving in Oss must be parked at the station until their next departure. Given that the station of Oss only has two platform tracks, the amount of trains that can be parked in Oss is limited.

The macroscopic timetable computed by the macroscopic timetable module is depicted as a time-space diagram in Figure 3. The horizontal axis represents time. The vertical axis represents space. From top to bottom, we have Nijmegen (Nm), the single track bridge (Mbrvo), Oss (O), ‘s-Hertogenbosch (Ht), a junction south of ‘s-Hertogenbosch (Vga), and Tilburg (Tb). Dark lines in the diagram represent long-distance trains, while light lines represent regional trains. The dashed lines represent partial train services that cannot be operated as a consequence of the disruption. As can be seen in the figure, no trains cross the disrupted area between 6:30 and 8:30.

The dotted arcs in the diagram connect two train services that are operated using the same rolling stock. Trains arriving in Oss are connected to train departures at a later time, indicating that the same rolling stock is assigned to both trains. As can be seen, the first train arriving in Oss that cannot continue is a long distance train from Nijmegen. This train

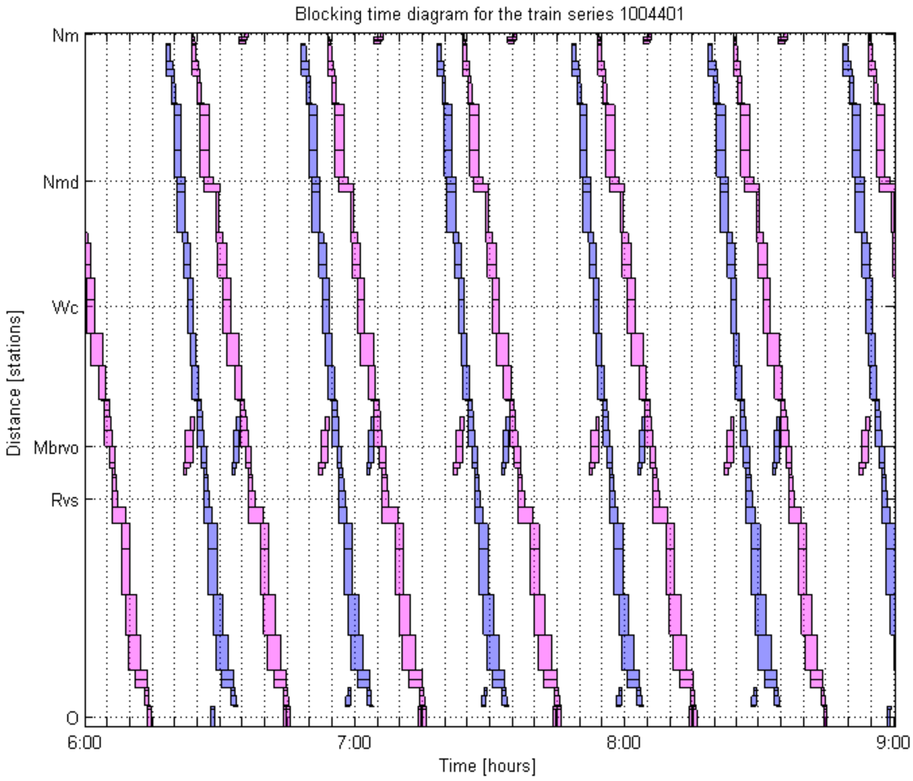

Figure 4: Blocking times diagram for the short turned train line 4401

is parked at a platform in Oss and later returns to Nijmegen. Between this arrival and the corresponding departure, one of the tracks is blocked.

As indicated by the arrows in the figure, regional trains from both directions are supposed to arrive at the same time in Oss. However, because one of the tracks is blocked by the long distance train, they should now use the same platform. In the macroscopic timetable module, it is initially assumed that the headway time between those trains is zero. The reason is that the trains do not use the same platform in the planned timetable. However, when the new timetable is validated in the microscopic timetable module, it turns out to be impossible to schedule the trains so closely together. This is reflected by a positive headway time from the departure of the first train to the arrival of the second that is computed by the microscopic timetable module in the first iteration. In the second iteration of the framework, the macroscopic timetable module takes this positive headway time into account, and retimes the arrival of the regional train at Oss during the disruption, making sure that enough time is available between the departure to Nijmegen of the first train and the arrival from Nijmegen of the second. In the time-space diagram in Figure 3, the arrivals at the second, third, fourth, and fifth arrow are shifted to later times. Because the headway times depend on the speed profiles of the trains, which depend on their turn on the exact arrival times, several iterations are needed before the iterative approach converges.

Finally, Figure 4 gives the corresponding blocking time diagram which shows that the obtained timetable is indeed microscopically feasible. The corridor between Nijmegen

(Nm) and Oss (O) is depicted. The x and y-axes represent time and stations, respectively. Note that the figure shows only the infrastructure used by a short-turned, regional train service in order to be able to indicate possible conflicts. The microscopic module computed blocking times with a high level of detail, therefore the open track stops are also presented: Nijmegen Dukenburg (Nmd), Wijchen (Wc) and Ravenstein (Rvs). Note that Mbrvo represents a short portion of a single-track line. Therefore, all trains running through Mbrvo use the same infrastructure. This explains why trains running in both directions appear in the blocking diagram in the region around Mbrvo. It is clear that all conflicts, particularly in O and Mbrvo, are resolved in the last micro-macro iteration and the rescheduling timetable is conflict-free at the microscopic level.

Table 1 presents the microscopic conflicts in the macroscopic timetable at the end of each iteration. In the first iteration, there are 6 conflicts that add up to 160 seconds of overlapping blocking times. In the second iteration, only 2 conflicts remain with a conflict time of 4 seconds. In the subsequent iterations, a final conflict is resolved. It can be seen from the table that the approach can deal with severe conflicts rather easily, but that it takes several iterations to find a timetable that is completely feasible both macroscopically and microscopically.

Table 1: Characteristics of the macroscopic timetable after each iteration

Iteration	**Number of conflicts**	**Time of conflicts (s)**
1	6	160
2	2	4
3	1	1
4	1	7
5	1	1
6	1	1
7	0	0

3.3 Rolling stock and crew rescheduling

When a timetable has been obtained that is both macroscopically and microscopically feasible, the rolling stock and crew schedules have to be adapted. We first reschedule the rolling stock. To this end, we first import the (macroscopic) departure and arrival times of trains at all stations where it is allowed to change rolling stock compositions. This results in a set of so-called trips to which rolling stock must be assigned. Special attention is required for the partially cancelled trips between ‘s-Hertogenbosch and Nijmegen, which are short-turned in Oss. Given that it is not allowed to change the rolling stock composition in Oss, the trips from Nijmegen to Oss that are short-turned onto trips from Oss to Nijmegen are interpreted as trips from Nijmegen to Nijmegen. This holds both for the regional and long-distance trains. When the set of trips has been determined, the rolling stock module is applied to reschedule the rolling stock. The model that we use for macroscopic timetabling already ensures that a feasible rolling stock circulation exists, so the resulting rolling stock schedule should not contain trips without rolling stock. However, some trips might be operated by rolling stock compositions different from those planned.

We first consider the long distance trains. Here, for 3 trips, a rolling stock composition is chosen that has less capacity than originally assigned. This means that less seats are available on this trip causing inconvenience for the passengers. For 6 trips, a rolling stock composition is selected with more capacity than planned. In total, only for 9 trips the rolling

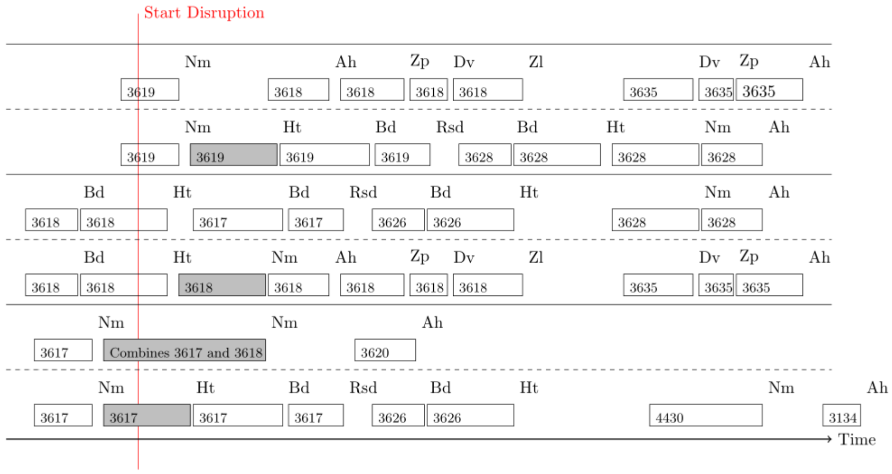

Figure 5: Rescheduled duties

stock composition had to be changed. Furthermore, only 3 new shunting operations are introduced.

For the regional trains, the capacity is reduced for 6 trips. The number of seats is reduced roughly by 30 percent. For 6 trips, the capacity is increased. The number of newly introduced shunting movements is 7. For these newly introduced shunting movements, local crew must be instructed to execute the shunting movements.

Also when rescheduling the crew, no additional train services need to be cancelled due to lack of crew. This means that we do not need to perform a second iteration to check the timetable and rolling stock schedules again. There are 17 duties that had to be changed to cope with the disruption. Among these 17 duties, there are 3 reserve duties. Reserve duties are crew duties that are specifically included in the schedule to be able to deal with disruptions.

As an example, Gantt charts of 3 rescheduled duties are depicted in Figure 5. Information about different crew members is separated by solid lines. For each of the duties, the original duty is shown at the bottom and the rescheduled duty at the top. Trips that are cancelled are represented by solid grey rectangles. The numbers in the rectangles are train numbers and indicate, among others, to which train series the train belongs. For example, train number 3617 belongs to train series 3600. This train series contains the long distance trains that run between the South-West and the North-East and cross the area where the disruption occurs. For the first (lowest) crew member, the trip from Nijmegen to 's-Hertogenbosch (with number 3617) is partially cancelled. Recall from the previous paragraph that this train runs to Oss, and is short-turned to Nijmegen there. As a consequence, this crew member does not arrive in 's-Hertogenbosch and cannot operate the subsequent trips from 's-Hertogenbosch to Breda (Bd) and further. Instead, this crew performs a new task that combines driving from Nijmegen to Oss (3617) and driving from Oss to Nijmegen (3618). Then, having returned in Nijmegen, this crew member executes a task on a later train (with number 3620) from Nijmegen to Arnhem (Ah). The second crew member was supposed to run a trip from 's-Hertogenbosch to Nijmegen and further (with number 3618). Because of the disruption, it is impossible to perform the work between Nijmegen and Zwolle (Zl). Instead, this crew member takes over the tasks on trains 3617 and 3626 from the first and the last two tasks of train 3628 from the third crew member. Similarly, the original work from the second crew member is performed by the third. This example demonstrates how the work is redistributed among the crew members.

4 Conclusions

In this paper a disruption management tool that can be used by railway operators during the disruption management process is proposed. The tool implements an iterative algorithm that consists of four modules: a macroscopic timetable module, a microscopic timetable module, a rolling stock rescheduling module, and a crew rescheduling module.

The iterative algorithm has been tested on an instance of Netherlands Railways, where a track is completely blocked for two hours. A feasible timetable, rolling stock circulation, and crew schedule are found. Several iterations are required between the macroscopic and microscopic modules before it converges to a feasible solution. The resulting solution shows that the iterative approach can be used by railway operators as a decision support tool during disruptions.

In future research more computational experiments will be conducted to test the effectiveness of our algorithmic framework. In order to evaluate the quality of the iterative approach, the solution will be imported in the simulation environment. Furthermore, the tool will be integrated in the web-service oriented architecture and tested in a close loop with the simulator.

Acknowledgements

The research leading to this paper has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) in the ON-TIME project under Grant Agreement SCP1-GA-2011-285243.

References

[1] Besinovic, N., Quaglietta, E., Goverde, R.M.P., 2013. “A simulation-based optimization approach for the calibration of dynamic train speed profiles”, *Journal of Rail Transport Planning and Management*, vol. 3, pp. 126-136.

[2] Besinovic, N., Quaglietta, E., Goverde, R.M.P., 2014. “Supporting tools for automated timetable planning”, In: Brebbia, C.A., Tomii, N., Tzieropoulos, P., Mera, J.M. (eds.), *Computers in Railways XIV,* pp. 565-576, WIT Press, Southampton.

[3] Cacchiani, V., Huisman, D., Kidd, M., Kroon, L., Toth, P., Veelenturf, L., Wagenaar, J., 2014. “An Overview of Recovery Models for Real-Time Railway Rescheduling”, *Transportation Research Part B: Methodological*, vol. 63, pp. 15-37.

[4] Hansen, I.A., Pachl, J., 2014. “Railway timetabling and operations”, Eurail press, Hamburg.

[5] Nielsen, L.K., Kroon, L.G., Maróti, G., 2012. “A rolling horizon approach for disruption managament of railway rolling stock”, *European Journal of Operational Research*, vol. 220, pp. 496-509.

[6] ON-TIME, http://www.ontime-project.eu, last accessed 8th November 2014.

[7] Potthoff, D., Huisman, D., Desaulniers, G., 2010. “Column Generation with Dynamic Duty Selection for Railway Crew Rescheduling”, *Transportation Science*, vol. 44, pp 493-505.

[8] RabbitMQ, www.rabbitmq.com, last accessed 5th November 2014.
[9] Veelenturf, L.P., Kidd, M.P., Cacchiani, V., Kroon, L.G., Toth, P., 2014. *A railway timetable rescheduling approach for handling large scale disruptions*, ERIM Report Series Research in Management, ERS-2014-010-LIS.