CASPT 2015

Optimal train rescheduling after conflict detection

S. Van Thielen · S. Burggraeve ·
P. Vansteenwegen

Abstract The increasing importance of public transport incurs a growing need for accurate trains. Though timetabling can account for possible delays, in practice, external events regularly lead to delays. Once trains are deviating from their schedule, conflicts can occur. A conflict implies that (at least) two trains require the same part of the infrastructure at the same time. Conflicts need to be resolved quickly in a way that disturbs the system as little as possible. Therefore, the impact on the whole network should be taken into account when solving conflicts. This paper discusses and evaluates three heuristics that solve conflicts, based on a detailed simulation. The first two are based on common dispatching rules, the last one is a self-designed strategy that incorporates the secondary conflicts in a way that it tries to limit the duration of all conflicts in the next hour. Given delay scenarios, the three different strategies are compared based on secondary delays and computation time.

Keywords Conflict resolution · Delay management · Dispatching · Real-time scheduling

1 Introduction

Increasing the punctuality of a train network is an important problem the Belgian railway infrastructure manager Infrabel has to deal with. Since 2007, the punctuality without neutralisation has dropped below 90 % (Infrabel, 2014). Many of these delays are knock-on delays caused by train conflicts. Detecting and resolving conflicts in real-time is a crucial step in increasing the punctuality and obtaining a more robust railway system.

KU Leuven, Leuven Mobility Research Centre - CIB
Celestijnenlaan 300, Box 2422
3001 Leuven
Tel.: +32-16-376504
Fax: +32-16-322986
E-mail: Sofie.VanThielen@kuleuven.be

On a daily basis, conflicts are detected when two trains require the same part of the infrastructure at the same time. This is mostly caused by a primary delay of at least one of the two trains involved. Dispatchers have to solve these conflicts as soon as possible by delaying or rerouting or in some cases even cancelling trains. In many cases, the solution of a conflict will cause secondary delays that might propagate through the entire network. Hence, one conflict can lead to many delays or even cancellations of several trains. Especially on the Belgian rail network, being the densest in the world, these propagations easily occur (Infrabel, 2015). Therefore, it is required to find optimal solutions that minimize the total delay such that the dispatcher can take the best decision (Hansen and Pachl, 2008).

In the next section, the problem description is discussed in more detail. In Section 3, the literature regarding conflict resolution is elaborated. Section 5 then introduces the three heuristics that are evaluated in Section 6.

2 Problem description

Soon, Infrabel will implement a new traffic management system (TMS) that will perform conflict detection. The TMS will predict whether a conflict will occur around 15 minutes before it would actually take place. Therefore, conflict resolution in the form of rescheduling can be carried out. In this paper we only consider changes in the timetable, rerouting is not considered as a solution yet. Since we will only consider small disturbances leading to conflicts, there is no need to consider reallocating rolling stock or crew duties yet (Cacchiani et al., 2014).

A rail network consists of lines of tracks that have signals on the side. The part of the infrastructure between two signals is called a block section. These signals ensure safety by blocking parts of the infrastructure. If a train occupies a block section, then no other train is allowed to enter the same block section. Figure 1 depicts a simplified area where the numbers indicate the block sections. The yellow blocks represent station platforms, the green lines are signals.

The network can be divided into two parts: station areas and inter-station areas. Typically station areas include many switches and signals, enabling trains to move from one side of the area to the complete other side. The station areas have manual signals such that any movement of trains is always controlled by a signaller. Inter-station areas are mostly double track lines where bi-directional traffic is allowed. This implies that there are two major movements along tracks: normal track regime (where the trains drive on the left side) and counter-track regime (where trains drive on the right side). These parts use Automatic Route Setting (Teshima et al., 2014).

A simulation tool has been built which can be used to compare in detail different strategies for conflict resolution. Additional to two well-known heuristics one new heuristic will be evaluated in order to perform conflict resolution.

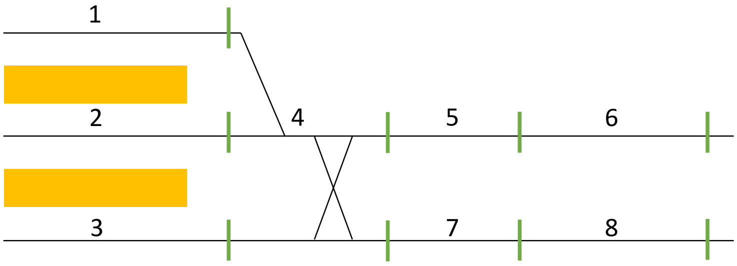

Fig. 1 Small artificial network showing station platforms (yellow blocks), block sections (numbers) and signals (green vertical lines)

3 Literature

The European deregulation of the rail network makes passenger and freight transport companies want to be competitive. Therefore interest in punctual trains and good service for customers is growing and potential conflicts should be solved in an optimal way. During the last decades many recovery models have been created (Cacchiani et al., 2014).

The optimisation problem arising from the dispatching problem can be seen as a job-shop problem with no-store (or blocking) constraints and is mostly represented using the alternative graph formulation (Mascis and Pacciarelli (2002); Mascis et al. (2004)). The problem is NP-hard and therefore needs to be solved using algorithms such as a branch and bound algorithm (D'Ariano et al. (2007b); D'Ariano et al. (2008)), a tabu search (Corman et al. (2010); Törnquist and Persson (2005)), a genetic algorithm (Wegele and Schnieder, 2004), etc.

Mostly, the problem is regarded on the microscopic level of the infrastructure, which gives much detailed information about the solution directly. The disadvantage is that the problem easily becomes huge and therefore is hard to solve. It will need a lot of computation time. However, due to the fact that the problem is situated in real-time, there is not much time available.

In order to limit the computation time it can be more interesting to consider models or approaches on the macroscopic level. Though this results into faster methods (Kecman et al., 2013), the dispatchers cannot directly use the solution that this method gives due to incompatibilities with the microscopic infrastructure.

Some research has also focused on splitting the network up into smaller local parts, where conflict resolution can be performed for distinct regions (Corman et al., 2012a). However, due to the dense Belgian star-shaped network with a large central bottleneck, it is very difficult to distinguish different regions inside the network.

The final goal of our research project is to implement a Decision Support System (DSS) that helps the dispatchers in making the best decision. Such DSS should be complementary to the TMS. Typically it contains three parts: 1) predicting the movement of the trains, 2) detecting conflicts and 3) propose how to solve the detected conflicts. This is a tool for real-time train dispatching. When looking at complete DSSs, several have already been proposed (Mazzarello and Ottaviani (2007); D'Ariano et al. (2007a); Mascis et al. (2004)). This paper focusses on the third part and discusses a simulation tool to evaluate different conflict resolution heuristics.

4 Study area

The large study area considered in this paper and on which the simulation is performed, is Brugge-Gent-Denderleeuw. This rail network is approximately 91 km long (Oostende-Denderleeuw) and 32 km wide (Waregem-Gent). It consists of 84 stations ensuring 232 platforms. The area includes 8850 block sections. The largest stations in this area are Gent-Sint-Pieters, Oostende and Brugge, each having ten platforms. Note that the study area also includes shunt yards. The network is regarded on a microscopic scale, considering switches, (automatic and manual) signals, block sections, etc.

The simulation covers both passenger and freight trains. A schematic view of the study area is depicted in Figure 2. Four important stations are marked on this blind map: Oostende, Gent-Sint-Pieters, Brugge and Denderleeuw.

5 Simulation

The discrete event simulation runs trains on the microscopic study area for the duration of two hours. The time window (7-9 am) was chosen during rush hour, ensuring many passengers and high capacity usage and includes 82 trains. Each run includes a different delay scenario (see 5.2) that is then tackled by every heuristic. The performance criteria to evaluate the heuristics are further discussed in 5.3.

5.1 Heuristics

This section describes three possible strategies for a dispatcher to solve conflicts in real-time.

5.1.1 First Come, First Serve (FCFS)

The first and easiest heuristic is based on the FCFS principle, which is often used in dispatching. If, due to a delay of at least one train, two trains need the

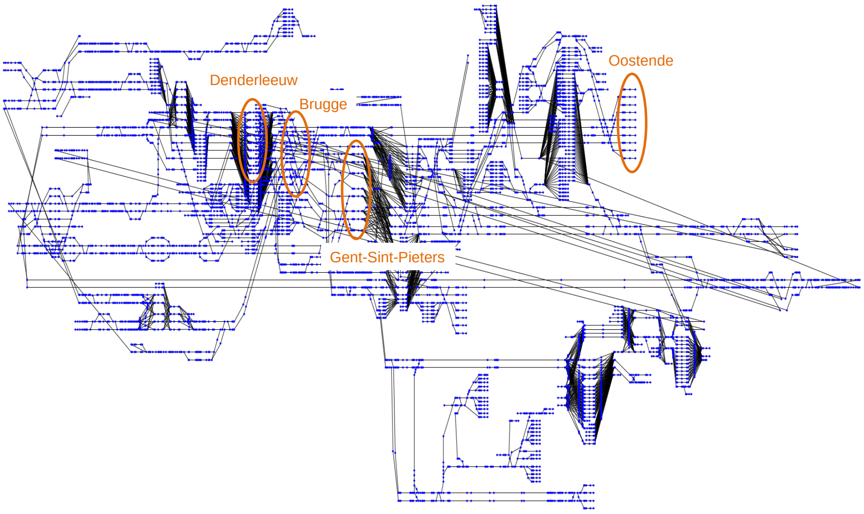

Fig. 2 A schematic overview of the study area

same part of the infrastructure at the same time, the first train that arrived, is the first one to operate on this part of the infrastructure. An important shortcoming of this strategy is that it is possible for a fast and/or crowded train to get stuck behind a slow train for as long as there is no station or an area for overtaking. Furthermore, extra 'secondary' conflicts that may be caused by this decision are not taken into consideration. This might be a serious shortcoming in congested areas.

5.1.2 Priority to type of train (PTT)

To counter one of the previous mentioned problems, the second heuristic takes the type of the trains into account. The trains' type is an important characterisation. It resembles the stopping pattern and the speed of the train. The type can be for example a Thalys (THA), which is a high-speed train, or a Peak hour train (P), a train that only drives during rush hour and mostly stops at every possible station.

This heuristic makes use of an ordering of the types: the highest type corresponds to the train with the highest speed and the least amount of stops along the way (THA), whereas the lowest type is a train with low maximal speed and many stops. In order to incorporate both passenger and freight trains, it is assumed that the economic cost of delaying a passenger is higher then delaying freight transport. Therefore the lowest type of trains are the freight trains.

If a train wants to start using a block section, it is examined first whether the block section is currently available. If so, it is also examined whether another train wants to use this block section as well during the expected running time of this first train. The first train will then only be delayed, waiting for the other train to pass first, if the other train has a higher priority. The train is only delayed if a train with a higher type requires the block section.

5.1.3 Conflict limitation (CL)

The third heuristic studied here is based on limiting the number and size of extra 'secondary' conflicts caused by giving one train priority over another train. This heuristic is self designed.

If a conflict is detected, the progress of both trains causing the conflict is examined for the two possible situations. Consider two trains A and B that cause a conflict. First, train A is given priority and gets to use the block section first. This immediately implies that train B is delayed. Subsequently the progress of train A is examined. Specifically the duration of extra 'secondary' conflicts that train A will cause during the next hour are summed up. Then the propagation of train B during the next hour is examined in the same way. The sum of secondary conflicts caused by giving A priority over B is then compared to the situation where B is given priority over A. The decision that generates the least minutes of extra 'secondary' conflicts is executed. In order to limit the computation time to determine this decision, only secondary conflicts are considered and no further conflicts. Finally the train that generates the shortest total duration of conflict is given preference to enter the block section first.

5.2 Delay scenarios

In order to compare the effectiveness of these three strategies, their performance is evaluated for different delay scenarios. Every scenario assumes a percentage of trains (20, 40, 60 or 80 %) that enter the network with a certain delay. This delay is based on an exponential distribution with an average delay of two minutes and a maximal delay of fifteen minutes.

5.3 Performance criteria

The performance criteria that are regarded here, are the maximal and average delay. In order to measure the delay, the delay of all trains is measured when the simulation ends (at 9 am) or when trains leave the study area (or when they reach their final destination in the study area).

Next to the general delay, the weighted delay is determined as well. The weighting is based on the type of train (see 5.1.2).

The average computation time, which equals the time it takes to find a solution for one delay scenario based on one strategy, is also reported.

6 Results

The heuristics described in Section 5.1 are implemented in C++. The different methods are compared by looking at average delay and the computation time (see 5.3). Because PTT and CL methods perform train movement prediction and conflict detection, it is expected that these methods will take significantly more computation time than the FCFS principle.

Table 1 Average delay (in s) for the three strategies and gap with FCFS

	Percentage delayed	Average delay
FCFS	20 %	27.98
	40 %	30.48
	60 %	41.96
	80 %	40.26
PTT	20 %	34.28 (+ 22.5 %)
	40 %	35.44 (+ 16.3 %)
	60 %	53.43 (+ 27.3 %)
	80 %	44.94 (+ 7.1 %)
CL	20 %	33.37 (+ 19.3 %)
	40 %	41.20 (+ 35.2 %)
	60 %	53.93 (+ 28.5 %)
	80 %	46.01 (+ 9.7 %)

For every percentage delayed trains 25 delay scenarios are created.

Table 1 shows the average delay for the different heuristics and for different delay scenarios. FCFS performs the best of all three purely looking at average delay. This is due to the fact that a limited two-hour window is examined. FCFS typically reduces delays on short-term, whereas the other two strategies try to achieve less conflicts on the long-term. The gap with FCFS, which is also shown in Table 1, is clearly the smallest at the highest percentage of delayed trains.

PTT and CL can be seen as more elaborate heuristics, since they predict the movement of trains for a certain amount of time. Because no rerouting is taken into consideration, these two methods are only capable of giving delays to trains and every interference directly has a negative influence on the delay.

Due to the fact that the evaluation criteria are solely based on the delays, it is expected that FCFS outperforms the other two strategies.

When interpreting these results, it should be noted that in many cases the delay at the end of the simulation or of trains leaving the study area is smaller then the delay that was imposed when entering the study area (20-80 %). This means that this network is far from saturated and that a lot of buffer is foreseen for each train, allowing most trains to catch up their delay in the study area. As a result, also the number of initial conflicts to solve is very limited (between 1 and 40 over all delay scenarios). The most important conclusion at this stage is that the simulation works on complex networks and can be used to evaluate different dispatching heuristics.

On a large network as is used here, it is not likely that many extra 'secondary' conflicts are caused. For a more thorough analysis of the different heuristics, more challenging case studies should be considered, with more trains and higher average initial delays.

Table 2 Average weighted delay (in s) for the 3 heuristics and gap with FCFS

	Percentage delayed	Average weighted delay
FCFS	20 %	27.72
	40 %	29.57
	60 %	40.30
	80 %	37.93
PTT	20 %	33.38 (+ 20.42 %)
	40 %	34.15 (+ 15.49 %)
	60 %	50.61 (+ 25.6 %)
	80 %	41.51 (+ 9.4 %)
CL	20 %	32.92 (+ 18.8 %)
	40 %	39.92 (+ 35 %)
	60 %	51.27 (+ 27 %)
	80 %	43.33 (+ 14.2 %)

Another evaluation criterion is the average weighted delay, where the weights are calculated based on the type of train. This criterion should give better results for the PTT strategy. Looking at Table 2 it is clear that the average delay remains approximately the same for the three strategies, probably again because of not enough primary and secondary conflicts occurring in the simulation.

Table 3 Computation time (in s)

	Average computation time
FCFS	2.51
PTT	141.65
CL	77.19

Table 3 shows the average computation time in seconds for the three different strategies, indicating the average time a two-hour run takes when entrance delays are given. The computation time of the FCFS strategy clearly outperforms the two other strategies. CL performs slightly better when it comes to computation time than the PTT, although CL predicts movements over the next hour, whereas PTT only predicts for the running time of the next block section.

7 Conclusion and future work

This paper describes and evaluates three different dispatching heuristics. All of them only consider rescheduling, no rerouting. FCFS, one of the most classic strategies, performs well due to the limited time-window and the objective function. By predicting future train movements, the CL strategy on average gives more delays than in the FCFS case, but it takes into account the running path of the trains. Due to the low saturation and high buffer times, many trains are capable to catch up the delays that were given entering the study area. This feature also entails a small amount of initial conflicts. Therefore the more advanced strategies (PTT and CL) cannot influence the solutions strongly.

Though the CL method should be tested on a network with more extra 'secondary' conflicts, it already outperforms the PTT strategy based on computation time.

The most important conclusion is that our simulation approach works on complex networks, is directly connectable to the infrastructure and timetable from Infrabel and that different dispatching strategies can be evaluated.

Further research should be dedicated to further improving heuristics such as the CL strategy. It can be interesting to study the propagation of conflicts, which can give an insight on the complexity of the network such that different approaches can be used when dealing with particular networks.

Also taking into account the passengers and creating a heuristic such that passenger delays are minimised can be studied.

Acknowledgements

This research is made in close cooperation with the Belgian railway infrastructure manager Infrabel.

References

Cacchiani, V., Huisman, D., Kidd, M., Kroon, L., Toth, P., Veelenturf, L. and Wagenaar, J., An overview of recovery models and algorithms for real-time railway rescheduling, Transportation Research Part B, 63, 15–37 (2014)

Corman, F., D'Ariano, A., Pacciarelli, D. and Pranzo, M. , A tabu search algorithm for rerouting trains during rail operations, Transportation Research Part B, 44, 175–192 (2010)

Corman, F., D'Ariano, A., Pacciarelli, D. and Pranzo, M., Optimal inter-area coordination of train rescheduling decisions, Transportation Research Part E, 48 (1), 71–88 (2012)

Corman, F., D'Ariano, A., Pacciarelli, D. and Pranzo, M., Bi-objective conflict detection and resolution in railway traffic management, Transportation Research Part C, 20, 79–94 (2012)

Dewilde, T., Sels, P., Cattrysse, D. and Vansteenwegen, P., Defining Robustness of a Railway Timetable, Proceedings of 4th International Seminar on Railway Operations Modelling and Analysis (IAROR): RailRome 2011, 4th International Seminar on Railway Operations Modelling and Analysis (IAROR) RailRome2011, Rome, 16-18 Februari 2011, 1–20 (2011)

D'Ariano, A., Pranzo, M. and Hansen, I., Conflict Resolution and Train Speed coordination for Solving Real-Time Timetable Perturbations, IEEE Transactions on intelligent transportation systems, 8 (2), 208–222 (2007)

D'Ariano, A., Pacciarelli, D. and Pranzo, M., A branch and bound algorithm for scheduling trains in a railway network, European Journal of Operational Research, 183 (2), 643–657 (2007)

D'Ariano, A., Corman, F., Pacciarelli, D. and Pranzo, M., Reordering and Local Rerouting Strategies to Manage Train Traffic in Real Time, Transportation Science, 42 (4), 405–419 (2008)

Hansen, I. and Pachl, J., Railway timetable and traffic: analysis, modelling, simulation. Eurail Press (2008)

Infrabel, Annual Reports Infrabel, Reported on 21/11/2014 from Infrabel: Right On Track: `http://www.infrabel.be/en/infrabel/annualreports`

Infrabel, Rail transport at the heart of economic challenges, Reported on 20/05/2015 from Infrabel: Rail Acces: `http://www.infrabel.be/sites/default/files/documents/INFRABEL_Industrie_EN.pdf`

Kecman, P., Corman, F., D'Ariano, A. and Goverde, R.M.P., Rescheduling models for railway traffic management in large-scale networks, Public Transportation, 5, 95–123 (2013)

Lamorgese, L. and Mannino, C., The track formulation for the Train Dispatching problem, Electronic Notes in Discrete Mathematics, 41, 559–566 (2013)

Mazzarello, M. and Ottaviani, E., A traffic management system for real-time traffic optimisation in railways, Transportation Research Part B, 41 (2), 246–274 (2007)

Mascis, A. and Pacciarelli, D., Job-shop scheduling with blocking and no-wait constraints, European Journal of Operations Research, 143 (3), 498–517 (2002)

Mascis, A., Pacciarelli, D and Pranzo, M., Scheduling Models for Short-Term Railway Traffic Optimisation, In Proceedings of the 9th International Conference on Computer-Aided Scheduling of Public Transport, San Diego CA (USA), August 9-11, 2004

Teshima, H., Hori, S., Shimura, A. and Sato, N., Railway Track Layout Modelling And Its Application To An Automatic Route Setting System, In: Computers in Railways XIV. Railway Engineering Design and Optimization (2014)

Törnquist, J. and Persson, J.A., Train traffic deviation handling using Tabu Search and Simulated Annealing. In: IEEE Proceedings of the 38th Annual

Hawaii International Conference on System Sciences (HICSS'05), January 2005, Hawaii, USA

Törnquist, J., Computer-based decision support for railway traffic scheduling and dispatching: A review of models and algorithms, L.G. Kroon and R.H. Mohring(Eds.), Proceedings of the 5th Workshop on Algorithmic Methods and Models for Optimization of Railways (2006)

Törnquist, J. and Persson, J.A., N-tracked railway traffic rescheduling during disturbances, Transportation Research Part B, 41, 342–362 (2007)

Wegele, S. and Schnieder, E., Automated dispatching of train operations using genetic algorithms, Computers in Railways IXWIT Press, Southampton, 775–784 (2004)