Railway traffic optimization by advanced scheduling and rerouting algorithms

Andrea D'Ariano
Università Degli Studi Roma Tre
142 PUBLICATIONS **3,093** CITATIONS
SEE PROFILE

Francesco Corman
119 PUBLICATIONS **2,019** CITATIONS
SEE PROFILE

I.A. Hansen
Delft University of Technology
50 PUBLICATIONS **953** CITATIONS
SEE PROFILE

Proceedings of the 8th World Congress on Railway Research, WCRR 2008,
18-22 May 2008, Seoul, Korea

Railway traffic optimization by advanced scheduling and rerouting algorithms

[1]A. D'Ariano, [1]F. Corman, [1]I.A. Hansen
Department of Transport & Planning, Delft University of Technology, Delft, The Netherlands[1]

Abstract

In the last years, real-time railway traffic optimization experienced an increasing interest due to the expected growth of traffic and the limited possibilities of enhancing the infrastructure, which ask for a more efficient use of resources and the application of more advanced decision support tools. This paper presents a computerized train dispatching system, called ROMA (*Railway traffic Optimization by Means of Alternative graphs*), for supporting railway traffic controllers during operations. Innovative scheduling and rerouting algorithms have been developed in order to globally optimize disturbed railway traffic conditions. ROMA can anticipate the evolution of traffic, including the propagation of delays in a regional railway network, and can estimate the effects of different dispatching measures during a period of about 15 minutes ahead. Therefore, ROMA would enable traffic controllers to frequently perform incremental changes to the actual timetable to accommodate changes in traffic patterns due to disturbances, such as train delays and blocked tracks. An extensive computational study is carried out, based on a dispatching area of the Dutch railway network, to show the high potential of our train dispatching system as a support tool to improve punctuality.

Introduction

Regulation of railway traffic aims at ensuring safe, seamless and as much as possible punctual train operations. Due to the strict time limits for computing a new timetable in presence of disturbances, train dispatchers usually perform manually only a few timetable modifications (i.e. adjust train routes, orders and speeds) while the efficiency of the chosen measures is often unknown. Some computerized dispatching support systems have been developed, so far, which can provide good solutions for small instances and simple perturbations. Recent reviews on the related literature can be found e.g. in Oh et al. [14] and Törnquist [17]. However, most existing dispatching systems operate based on local information and decisions are taken locally, "on the spot and now". These systems are able to provide viable solutions only if few trains are delayed and the chosen traffic control actions are often sub-optimal. This kind of system cannot deal with heavy disturbances in larger networks as the actual train delay propagation is simply extrapolated and does insufficiently take into account the train dynamics and signaling constraints. Therefore, extensive control actions are necessary to obtain globally feasible solutions. For instance, Olsson and Hauglund [15] report the factors that affect train punctuality for the Norwegian Railways, Kauppi et al. [12] explain the limitations of the train dispatching process in Sweden and Geitz [10] gives an overview of the urgent need for automated railway traffic management tools in emerging economies.

Advanced real-time traffic management systems should take into account the whole traffic in a larger area, detecting future conflicts among train movements (that have direct impact on the level of punctuality), automatically calculating optimal traffic flow and suggesting possible change of orders or routes to the dispatcher, as well as displaying advisory speeds to the train drivers. This kind of systems must operate sufficiently in advance with the aim to correctly quantify the effects of different dispatching measures and enable traffic controllers to frequently perform incremental modifications of the actual timetable in order to adapt to sudden traffic disturbances.

In this paper, we compare two dispatching support systems. The first is based on local information, a common practice in railway real-time management, and is similar to the ARI system used in the Netherlands. The second makes use of advanced scheduling and rerouting algorithms that have been implemented in the recently developed ROMA (Railway traffic Optimization by Means of Alternative graphs) software, an optimization tool based on global information. ROMA is able to estimate and control the future evolution of the railway traffic considering actual train positions, signaling and safety operating rules and conditions, as well as dynamic train characteristics. ROMA computes a dispatching solution that minimizes train delays and their propagation by pro-actively detecting and solving train conflicts within short computation times. In our terminology, a conflict occurs whenever a train requires a block section which is not available, i.e., occupied by another train or temporarily blocked. The mathematical models, algorithms for conflict resolution and applications of ROMA software are described in detail in D'Ariano et al. [3-9].

The problem of defining train routes and orders at stations, junctions and passing points can be viewed as a job shop scheduling problem with additional constraints. An alternative graph formulation (Mascis and Pacciarelli [13]) is used to model the problem and a truncated branch and bound procedure (D'Ariano et al. [6]) finds a train schedule that is optimal at a network scale. A rescheduling procedure computes a deadlock-free and conflict-free schedule, being fixed the route of each train.

A better use of rail capacity and a further improvement of punctuality are achieved by an iterative adjustment of train orders and routes (D'Ariano et al. [4]). An innovative rerouting algorithm has been added recently to determine the best routes and train sequences in corridors and stations. The rerouting actions are evaluated and an iterative rescheduling procedure is then performed within strict time limits. The effectiveness of extensive rerouting strategies is explored by incorporating the search for new routes in a tabu search scheme, in order to escape from local minima (D'Ariano et al. [5]).

The blocking time theory is adopted to compute the minimum required separation between consecutive trains (as described e.g. in Pachl [16]) and refers to the actual Dutch signaling system (as described e.g. in Goverde [11]). The resulting solution minimizes train delays with respect to a disrupted timetable, ensures minimal safety distances between any pair of trains while respecting the actual signaling constraints, and is compatible to the actual train positions and speeds.

Computational experiments based on the Dutch rail network are presented. The railway area is around 50 km long and consists of the Utrecht - Den Bosch link and the Den Bosch station. We consider a cyclic and hourly timetable, which contains around 40 trains each hour. In this paper, the railway traffic simulation includes both passenger and freight trains but optimal dispatching solutions are computed without introducing priorities between the running trains. Given a disturbance, i.e., entrance delay or blocked track, a new feasible plan has to be produced within a short computation time. In our experiments, for each perturbed situation we generate several feasible schedules by using different configurations of our dispatching support system. This allows us to quantify the effects of various traffic management strategies.

The next section describes the architecture of ROMA dispatching support system, while our computational experience is shown in a successive section. A concluding section discusses the implementation status of the proposed system and indicates future research directions.

ROMA dispatching support system

We have developed a laboratory version of dispatching support tool that is designed to work during operations. The proposed dispatching system is tested on an off-line data set and does not include coupling to actual train monitoring data. We therefore assume that the exact speed and location of each train can be updated in real-time. Hence, the impact of inaccurate location and speed data is supposed negligible. A discussion of the real-time applicability of the dispatching system and the necessary communication links between the running trains and the traffic control centers can be found in D'Ariano et al. [8] and in the conclusions of this paper.

A human dispatcher should interact with the system by adding/removing constraints or changing the existing timetable. Figure 1 presents the overall ROMA dispatching support system architecture, which is composed by interrelated modules. We now describe the function of each module and how the four introduced subproblems are solved:

- Data loading module (subproblem (i)): Collect data from the field such as the current infrastructure status, the existing timetable, the actual position and speed of all running trains, and forecast the time needed to complete the next scheduled operations.
- Disruption recovery module (subproblem (ii)): Given a default route and a prioritized set of rerouting options, find a passable routing for each train by avoiding, eventually, the tracks blocked in the studied railway area.
- Real-time railway traffic optimization module (subproblem (iii)): Given a set of dynamic traffic management strategies, i.e., flexible orders, routes and departure times, find a new deadlock-free and conflict-free schedule by rescheduling and/or rerouting trains. In this phase the minimum traversing time of a block section for a train is considered fixed.

- Train speed coordination module (subproblem (iv)): Given the schedule computed by the previous module, check if the solution is consistent with the train dynamics and if the blocking time of each train in each block section overlaps with those of the following trains, i.e., if the distance headway between two or more trains is not respected. In case of an overlap of blocking times, perform an iterative procedure to compute acceptable train speed profiles on the basis of the actual signal aspects, infrastructure and rolling stock characteristics.

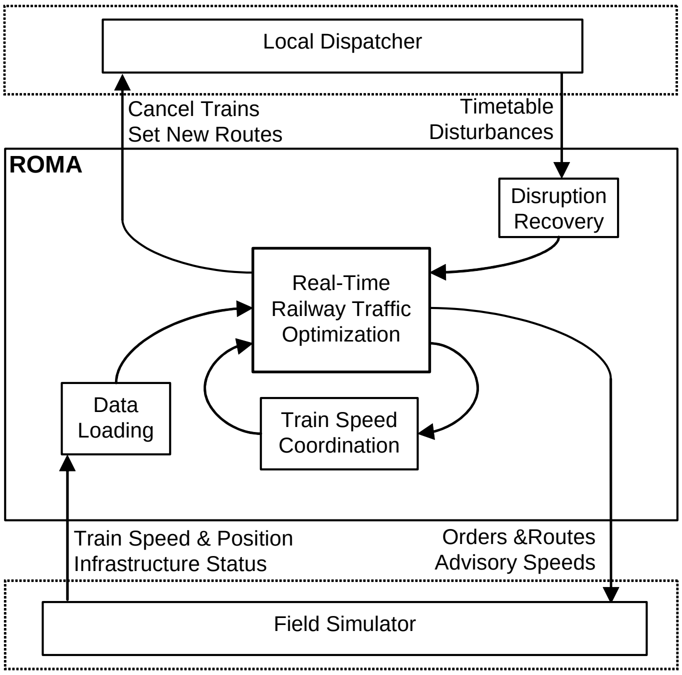

Figure 1: Architecture of the ROMA dispatching support system

The data loading module (i.e., subproblem (i)) is in charge to gather all the information, which is required by the other modules. This module loads static data (off-line data such as infrastructure and timetable information) and dynamic data (train detection and other real-time information that varies in time) from the field. The operational timetable contains a list of arrival/departure times for a set of relevant points in the network, including all the station platforms visited by each train. The infrastructure consists of a set of available block sections delimited by signals. Infrastructure data includes the status and length of each block section and other characteristics, such as speed limitations and the traversing direction. The data associated with each train includes speed and position at its entrance of the network, acceleration and braking curves (calculated on the basis of traction force/speed diagrams and maximum speeds) and a prioritized list of routing options (the most evident and frequently used alternative routes are selected by the dispatcher and given to the dispatching support system). At any time a route, i.e., a sequence of block sections, is feasible if none of its block section is blocked. Finally, the blocking time for each pair (train, block section) is computed by this module on the basis of current rolling stock characteristics and infrastructure data.

After the loading phase, the disruption recovery module checks if there are blocked zones in the network, which make infeasible some route (i.e., subproblem (ii)). This module is used to set a passable route for each train based on an evaluation of a number of prioritized routing options. Infeasible routes are discarded from the list of available routes for each train, and the feasible route with highest priority is assigned to the corresponding train. If no passable route is available for a train the system asks for an external support by the human dispatcher that is asked to set up new routes or cancel some trains. In such cases, emergency timetables may be adopted and train routes are strongly modified, e.g. enabling a train to reverse the running direction.

The dispatching support system must be able to efficiently detect and solve the conflicts arising in the rail network during perturbed operations. After a passable route has been computed for each train, a conflict detection procedure identifies the potential headway and route conflicts with a high level of accuracy while considering all trains simultaneously. The potential conflicts are determined by predicting the future location of trains based on information about the actual state of the rail network.

In first instance, an alternative graph formulation (Mascis and Pacciarelli [13]) of the conflict detection and resolution problem with fixed train speed profiles (i.e., subproblem (iii)) is adopted. Since a train must traverse the block sections in its route sequentially, a route is modeled, in the alternative graph, with a chain of precedence constraints. Since a block section cannot host two trains at the same time, a potential conflict occurs whenever two or more trains require the same resource. In this case, a passing order must be defined between the trains, and is modeled introducing in the graph a suitable pair of alternative arcs for each pair of trains traversing a block section. A conflict-free schedule is next obtained by selecting one of the two alternative arcs from each pair, in such a way that there are no positive length cycles in the graph. Precisely, a positive length cycle in the graph corresponds to a deadlock situation. In order to evaluate a schedule, we use as performance index of a solution the maximum consecutive (knock-on) delay, which is the maximum delay introduced when solving conflicts in the dispatching area, i.e., the objective function is the minimization of the largest conflict in the network (D’Ariano et al. [6]). This is caused by the propagation of the input delays of late trains to the other trains in the railway area. The average consecutive delay is therefore the average delay due to conflicts between trains running in the simulation period at all the relevant points for which a planned arrival time is specified in the timetable. Relevant points include all the stations and the exit point of each train from the network.

The main value of the alternative graph is the level of detail that can be included in the model. In fact, this graph incorporates a description of the network topology at the level of railway signal aspects and operational rules. Moreover, it can easily include other constraints relevant to the railway practice, such as minimum required time for rolling stock and passenger connections (D’Ariano et al. [4]), and flexible arrival/departure times at scheduled stops (D’Ariano et al. [7]).

The real-time railway traffic optimization module is the decisional kernel of the traffic management system that allows to take train rescheduling and rerouting decisions considering all the train speed profiles fixed as scheduled and aims at the minimization of train delays in the network. To alter the original timetable as little as possible, we focus here on the development of conflict resolution actions based, in first instance, on local dispatching measures. Large timetable modifications and cancellation of train routes are among the possible dispatching measures but not performed automatically.

This module is dedicated to perform the following sub-tasks: (a) given a feasible route for each train, define a schedule for each train, i.e., define its entrance time on each block section; (b) given the train schedules, search for routing options potentially leading to better schedules. The two sub-tasks are executed iteratively until no improvement is possible within a time limit of computation (Figure 2).

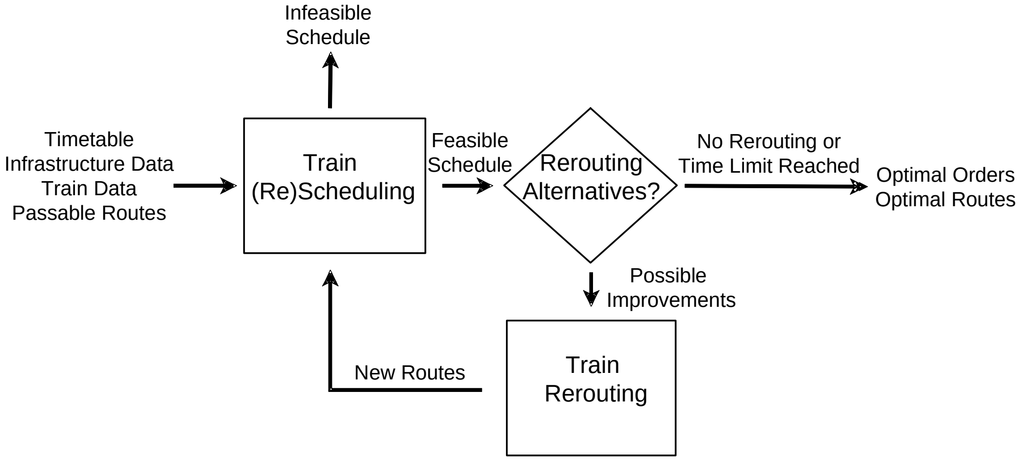

Figure 2: Architecture of the real-time railway traffic optimization module

As for solution algorithms in ROMA dispatching system, task (a) can be achieved by two optional scheduling algorithms. Both algorithms require that a routing for each train is given and a fixed traversing time of each block section is known in advance, except for a possible additional waiting time between operations in order to solve train conflicts. The first scheduling algorithm is the branch and bound algorithm (BB) described in (D'Ariano et al. [6]), which is able to sequencing instances of practical size within short computation times. This is an advanced scheduling algorithm with the aim of minimizing the propagation of train delays. The second scheduling algorithm, described in (D'Ariano et al. [4]), simulates the practice of traffic management adopted in the Netherlands. This algorithm is based on the ARI system (Berends and Ouburg [1]), a semi-automated system which detects and solves train conflicts one at a time. Our implementation is a completely automated version of the ARI system, simulating the behavior of the human dispatchers using priority rules.

Task (b) can be achieved by a tabu search algorithm (TS) in order to select an alternative route for some trains with the aim of further reducing train delays (D'Ariano et al. [5]). Given the schedule of task (a), the procedure analyzes all the feasible routes of each train, searching for a train route that enables a reduction of train delays and a better use of the available rail capacity in presence of timetable disturbances. Whenever a better schedule is found, the new route is set as default route and the search is repeated, until no improving routes are found or a given time limit is reached.

In the ROMA dispatching support system, the solution to subproblem (iii) presents deterministic blocking and waiting times and thus does not model explicitly the dynamic consequences of braking and subsequent acceleration imposed to avoid headway conflicts between trains running in the network. For the Dutch signaling system, this means that the traversing time of each train on each block section is computed assuming green signal aspects.

A variable-speed model for train dispatching (D'Ariano et al. [8]) is adopted to compute feasible solutions in terms of safe space headway constraints between consecutive trains (i.e., subproblem (iv)). An iterative strategy coordinates a set of procedures to build a feasible solution for the train scheduling problem with variable-speed profiles. On the basis of the blocking time theory, a feasibility check procedure verifies whether the schedule is compatible with the actual train dynamics and signal aspects or not, which is the case when a train traverses a block section with yellow (or red) signal aspect. In the latter case, a speed updating procedure adjusts the train speed profiles according to typical driver behaviors and to the dynamics of the rolling stock, i.e., computes an acceptable speed profile for each running train. This check-update activity is performed for increasing time values until a schedule with admissible train dynamics is obtained. New train speed profiles are obtained, which comply with the existing signaling system and rolling stock characteristics. Optimal energy-efficient running times can also be computed afterwards (D'Ariano and Albrecht [3]).

The dispatching system solution obtained by the variable-speed dispatching model can be finally suggested to the railway traffic controllers before its actual implementation. After checking the suggested solution, he could confirm the proposed actions or choose other dispatching measures. The dispatcher's decision would then be communicated to the interlocking system, that sets the interconnected switches and signals, and to the drivers in the train cabins by means of radio data transmission and indication of the new target speed on display. Consequently, we suppose that trains are equipped with on-board computers for automatic train control. Due to the synchronization time, i.e., the time to react to changing conditions, speed and location modifications may happen while the dispatching support system is computing a solution. However, since the proposed variable-speed dispatching support system is able to compute a feasible solution in a few seconds, depending on the time period of traffic prediction, we assume that such real-time variations would not affect the principal validity of the rescheduling solution. Therefore, the solution generated provides accurate train dynamics and describes a realistic traffic plan over the intended time horizon.

Computational experiments

This section reports on our experiments on a large sample of practical size instances. Our test case is the dispatching area of Utrecht - Den Bosch, a bottleneck of the Dutch railway network. We study the network simulating several disturbed traffic conditions. Dispatching algorithms are implemented in C++ language and executed on a PC equipped with a 3 GHz Pentium D processor, 1 GB Ram and Linux operating system. Computational times and delays are always expressed in seconds. Each run of the BB algorithm is truncated after 10 seconds of computation, while the overall time allowed to the real-time railway traffic optimization module to compute a solution is limited to 120 seconds. This choice makes the code compatible with real-time railway traffic management.

The dispatching area under study is shown in Figure 3. This railway includes the Den Bosch station and the line connecting Utrecht to Den Bosch, which is around 50 km long. There are 2 main tracks, a dedicated stop for freight trains and 7 intermediate passenger stations. There are several potential conflict points along the corridor due to mixed train traffic, and three critical crossings, due to the two tracks per direction of Geldermalsen station, Dordrecht corridor, Betuweroute corridor and Den Bosch station. However, the current infrastructure offers a number of possibilities of train reordering and local rerouting. For each train a default route and a set of local rerouting options are given. Rerouting options can be applied along corridors or within a station, in which a train may be allowed to stop at different nearby platforms.

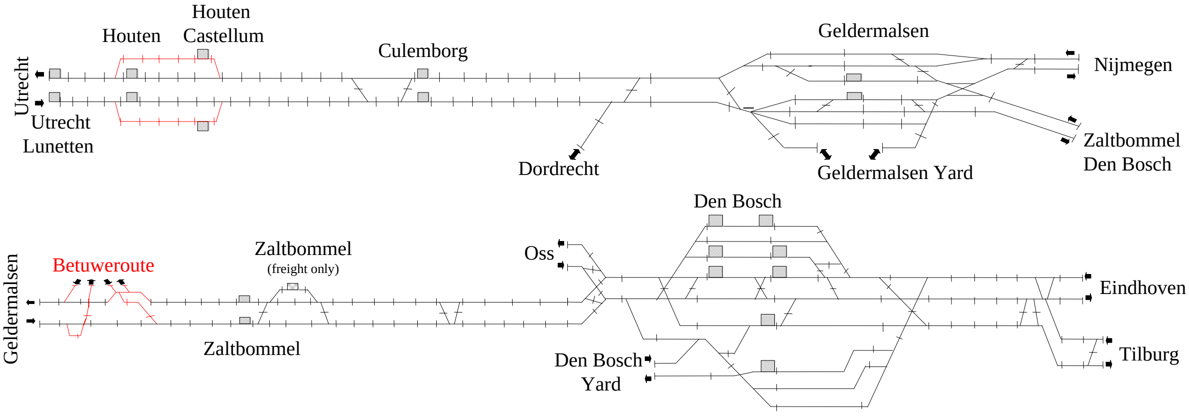

Figure 3: Railway dispatching area between Utrecht and Den Bosch

We consider a clockface hourly timetable for 2007 with minimum transfer time between connected train services. During peak hours, 26 passengers and freight trains in both directions are scheduled for the area around Geldermalsen. A more complex situation occurs at Den Bosch station, where up to 40 trains are scheduled each hour. Rolling stock connections are located in Zaltbommel and Den Bosch stations. Passenger connections are modeled at Den Bosch station for the traffic directions from Oss to Utrecht and vice versa. The minimum time for passenger connections varies from two to five minutes, depending on the distance between the arrival and departure platforms.

The ROMA dispatching support tool is tested under disturbed traffic conditions. In total, 32 test cases are defined, each of those for a simulation period of one hour. We look at 16 instances with timetable perturbations (i.e., train delays at the network entrance) and 16 instances with timetable perturbations plus infrastructure disruptions (i.e., randomly generated blocked tracks). Table 1 presents the average values of the two types of disturbances. Specifically, the second and third columns show the maximum and average delays at the entrance of the network. These are randomly chosen in a time window of typical train delays for this railway area. The fourth column shows the number of block sections that are blocked, while the fifth column indicates the percentage of train routes that have been filtered by the disruption recovery module since they were unavailable due to track blockage. We also include the relevant case in which the double track corridor is blocked in one of the two traffic directions and requires the trains running in opposite directions to share the only track available (see e.g. the double track corridor between Zaltbommel and Den Bosch).

Traffic Disturbances	Max Entrance Delay	Avg Entrance Delay	Block Sections Blocked	% Disrupted Train Routes
Timetable Perturbations	1350	295	-	-
Additional Disruptions	1350	295	5	67

Table 1: Description of the disturbed traffic conditions

Table 2 reports the performance of the ROMA tool when dealing with the timetable perturbations. The first row of this table presents the computational results by using the ARI-like system. We consider this algorithm as the base case that simulates the current rail operations. The second and third row refer to the average results by using the advanced scheduling algorithm (BB) and the rerouting optimization algorithm (TS), respectively. Each column of this table shows the average results on the 16 timetable perturbations. The second and third columns report the maximum and average consecutive delays (that are due to conflicts between consecutive trains in the studied railway area), while the fourth column gives the total computational times. The fifth column presents the percentage of train routes that have been changed by the real-time railway traffic optimization module starting from the set of default routes given by the disruption recovery module. The last column indicates the percentage of train orders that have been changed with respect to the timetable.

Roma Configurations	Max Cons Delay	Avg Cons Delay	Total Comp Time	% Train Route Changed	% Train Orders Changed
Automatic Route Setting (ARI-like)	342.0	38.7	4.8	-	12.3
Scheduling Algorithm (BB)	246.4	27.8	3.9	-	16.9
Rerouting Algorithm (TS)	238.7	24.6	127.9	15.5	12.2

Table 2: Performance of the ROMA configurations in case of timetable perturbations

In Table 2, the BB algorithm based on global information provides by far better results with respect to the ARI-like system (around 28% in terms of both average and maximum consecutive delays). For these two algorithms, the computation time is less than 5 seconds. When using the TS algorithm, consecutive delays can be further minimized with respect to the BB algorithm (more than 11% in terms of average consecutive delays and 3% in terms of maximum consecutive delays). The delay reduction using the TS algorithm is even larger when compared with the ARI-like system (more than 36% in terms of average consecutive delays and 30% in terms of maximum consecutive delays). However, the computation time of the compound scheduling and rerouting module becomes more critical for a real-time use of the decision support system and a simulation period of one hour.

Table 3 presents the performance of the ROMA tool when dealing with more serious traffic disturbances. Each column of this table shows the average results of the 16 timetable perturbations plus the infrastructure disruptions. The obtained dispatching solutions have larger delays, especially the maximum consecutive delay computed by the ARI-like system. The computation time also increases considerably, and the BB and TS algorithms often reach the given time limit of computation. In fact, less rerouting alternatives are available due to the presence of blocked tracks but the search for optimal conflict-free and deadlock-free solutions is more challenging. As a result, the percentage of train routes and orders changed is more relevant compared to the results of Table 2.

Roma Configurations	Max Cons Delay	Avg Cons Delay	Total Comp Time	% Train Routes Changed	% Train Orders Changed
Automatic Route Setting (ARI-like)	470.0	42.4	9.7	-	17.9
Scheduling Algorithm (BB)	287.5	32.7	8.1	-	17.5
Rerouting Algorithm (TS)	276.2	31.6	133.0	21.8	18.6

Table 3: Performance of the ROMA configurations in case of disrupted traffic conditions

Conclusions

A real-time train dispatching is in charge of solving disturbances in the traffic flow adjusting the timetable of each train in terms of routing and timing, and by resequencing the trains at the entrance of each merging/crossing point. This paper presents the architecture of our laboratory dispatching support system designed for the real-time optimization of train scheduling, routing and speed co-ordination. The main goal is to minimize consecutive delays (i.e., the difference between the arrival time at each relevant point in the new schedule and that in the timetable) while satisfying the traffic regulation constraints and the compatibility with the real-time position of each train. The computational results show the effectiveness of using real-time railway traffic optimization algorithms with respect to simple and local dispatching procedures. We now point out the potential and limitations of the proposed dispatching support system.

The future evolution of railway traffic should be predicted using real-time data gathered from the field (subproblem (i)). However, the technical implementation issues concerning the practical operation, such as data transmission, communication of delays and the realization of the proposed dispatching measures, have not been discussed in this paper. We are aware that the required data may not always be readily available. The existing train describer system records automatically the occupation and release of each signal block and the train number and its actual passing time at the critical block signals of the network. This information is used by many railways for comparison with the scheduled passing times. Thus, the difference between the scheduled and the measured times is computed and could be used, too, as input data for the dispatching support system (Daamen et al. [2]).

ROMA is able to optimize railway traffic also even if the timetable is not deadlock-free and conflict-free, and can be used to test the schedule robustness. This enables managing railway traffic even in case of severe traffic disturbances, including the presence of completely blocked tracks and routes (subproblem (ii)), such as when emergency timetables are required and dispatchers need efficient support to solve conflicts.

The proposed approach combines rerouting and rescheduling strategies for solving real-time disturbances, which require extensive timetable modifications to reach a feasible schedule (subproblem (iii)). All trains are taken into account simultaneously (global dispatching actions). This enables a better use of infrastructure capacity and an improvement of train punctuality.

By using detailed information on the actual infrastructure status and rolling stock characteristics, the ROMA dispatching support system can calculate accurate running times and blocking times that identify the effective remaining distances and running times until the next conflict points. An iterative procedure to coordinate the train speed profiles is therefore adopted (subproblem (iv)). However, in the current implementation of our dispatching support system the train dynamics are computed separately from the scheduling problem.

Future research should address the integration of the proposed system into a larger framework including several dispatching areas. To this end, it is challenging to address the decomposition of large problems into smaller problems to be solved by local dispatching systems in order to ensure their coordination globally and to compute effective solutions for the whole rail network.

Acknowledgements

We thank the department of strategy and innovation of ProRail for providing the instances. This work is partially supported by the programs "Towards Reliable Mobility" of the Transport Research Centre Delft, the Dutch foundation "Next Generation Infrastructures" and ProRail.

References

[1] N. Berends, N. Ouburg. “Beschrijving ARI-functionaliteit”, *ProRail Internal Specification*, The Netherlands, (2005).

[2] W. Daamen, T. Houben, R.M.P. Goverde, I.A. Hansen, V.A. Weeda, “Monitoring system for reliability of rail transport chains”, *Proceedings of the 7th World Congress on Railway Research*, Montreal, Canada, (2006).

[3] A. D'Ariano, T. Albrecht. “Running time re-optimization during real-time timetable perturbations”, In: J. Allan, C.A. Brebbia, A.F. Rumsey, G. Sciutto, S. Sone, C.J. Goodman (Eds.), *Proceedings of the 10th International Conference on Computers in Railways*, Southampton, UK, pp. 531-540, (2006).

[4] A. D'Ariano, F. Corman, D. Pacciarelli, M. Pranzo. “Real-time train conflict detection and resolution: Global sequencing and local rerouting”, In: H.J. van Zuylen (Ed.), *Proceedings of the 9th TRAIL Congress 2006, Selected Papers,* Delft University Press, The Netherlands, pp. 77-92, (2006).

[5] A. D'Ariano, F. Corman, D. Pacciarelli, M. Pranzo. “A tabu search algorithm for rerouting trains during rail operations, *INFORMS Annual Meeting*, Seattle, Washington, U.S.A. First prize winner at RAS Student Paper Competition. Presentation available at the web site www.rasig.net, (2007).

[6] A. D'Ariano, D. Pacciarelli, M. Pranzo. “A branch and bound algorithm for scheduling trains on a railway network”, *European Journal of Operational Research*, 183 (2), pp. 643–657, (2007).

[7] A. D’Ariano, D. Pacciarelli, M. Pranzo. “Assessment of flexible timetables in real-time traffic management of a railway bottleneck”, *Transportation Research Part C*, in press, (2008).

[8] A. D'Ariano, M. Pranzo, I.A. Hansen. “Conflict resolution and train speed co-ordination for solving real-time timetable perturbations”, *IEEE Transactions on Intelligent Transportation Systems*, 8 (2), pp. 208-222, (2007).

[9] A. D’Ariano, M. Pranzo. “An advanced real-time train dispatching system for minimizing the propagation of delays in a dispatching area under severe disturbances”, *Networks and Spatial Economics*, to appear, (2008).

[10] W. Geitz. “Railway operations research in emerging economies - A practical point of view”, In: I.A. Hansen, A. Radtke, J. Pachl and E. Wendler (Eds.), *Proceedings of the 2nd International Seminar on Railway Operations Modelling and Analysis*, Hannover, Germany, www.iaror.org, (2007).

[11] R.M.P. Goverde. “Punctuality of Railway Operations and Timetable Stability Analysis”, T2005/10, *TRAIL Thesis Series*, The Netherlands, (2005).

[12] A. Kauppi, J. Wikström, B. Sandblad, A. Andersson. “Future train traffic control: control by re-planning“, *Cognition, Technology & Work*, 8 (1), pp. 50–56, (2006).

[13] A. Mascis, D. Pacciarelli. “Job shop scheduling with blocking and no-wait constraints”, *European Journal of Operational Research*, 143 (3), pp. 498-517, (2002).

[14] S. Oh, S. Hong, I. Choi. “Railway conflict detection and resolution in the Korean railway system”, In: J. Allan, C.A. Brebbia, R. Hill, G. Sciutto, S. Sone (Eds.), *Proceedings of the 9th International Conference on Computers in Railways*, Southampton, UK, pp. 675–684, (2004).

[15] N. Olsson, H. Hauglund. “Influencing factors on train punctuality results from some Norwegian studies”, *Transport Policy*, 11, pp. 387–397, (2004).

[16] J. Pachl, “Railway Operation and Control”, *VTD Rail Publishing*, Mountlake Terrace, USA, (2002).

[17] J. Törnquist, “Railway traffic disturbance management”, *Doctoral Dissertation Series Number 2006:03, School of Engineering, Blekinge Institute of Technology*, ISBN 91-7295-086-2, (2006).