


Stability of railway dispatching solutions under a stochastic and dynamic environment

Conference Paper**Author(s):**

Quaglietta, Egidio; [Corman, Francesco](#) ; Goverde, Rob M.P.

Publication date:

2013

Permanent link:

<https://doi.org/10.3929/ethz-b-000184812>

Rights / license:

[In Copyright - Non-Commercial Use Permitted](#)

Stability of railway dispatching solutions under a stochastic and dynamic environment

Egidio Quaglietta^{1,†}, Francesco Corman^{2,3}, Rob M.P. Goverde¹

¹Dept. of Transport and Planning, Delft University of Technology, The Netherlands

²Center for Industrial Management, KU Leuven, Belgium

³Dept. of Transport Engineering & Logistics, Delft University of Technology

[†]Corresponding author: e.quaglietta@tudelft.nl

Abstract

In the last decade simulation models and optimization environments have been developed that are able to address the complexity of real-time railway dispatching. Nevertheless, actual implementations of these systems in practice are scarce. Essential for implementation of an advanced dispatching system is the trust of traffic controllers into a stable working of the system. Nervous systems might change advice suddenly, and even switch back to a solution previously discarded, as time and knowledge of the perturbation progress. To this end, we propose several metrics and a framework to assess the stability of railway dispatching solutions under incomplete knowledge, and report on the evaluation of the state-of-the-art dispatching system ROMA, coupled with the simulation environment EGTRAIN, here considered as a surrogate of the real field. Rescheduling plans calculated at different control stages have been compared for different prediction horizons of the rescheduling tool. This setup has been applied to the Dutch Utrecht-Den Bosch corridor. Results obtained from this case study show that more stable control strategies are achieved when using shorter prediction horizons. Train retiming is scarcely sensitive to changes in the prediction horizon but more affected by the dynamic propagation of random disturbances over time.

Keywords

Train dispatching, stability analysis, real-time railway traffic management,

1 Introduction

Railway traffic is strongly influenced by random disturbances during operations which cause deviations from the original schedule and thereby reducing performances. In order to cope with small perturbations, the design of a robust timetable can be an effective solution. If larger disturbances or service disruptions are observed, it is necessary to adopt real-time dispatching measures to effectively reschedule (reorder, retime or reroute) train services into new updated conflict-free train path plans. In practice, dispatching decisions are taken by traffic controllers based on their own experience or rule-of-thumbs, to solve observed conflicts as soon as possible. The myopic and limited knowledge that dispatchers have of traffic evolution can anyway lead to implement plans that are ineffective or even counterproductive. For this reason, several approaches have been proposed in literature (see e.g. [11], [26], [9]) for the optimal real-time management of traffic perturbations.

Most of them refer to a closed-loop rolling horizon framework (e.g. [17], [6], [3]) where at regular time intervals (*rescheduling interval*) current traffic information (e.g.

train speeds and positions) is collected from the field; the behaviour over a pre-set time period ahead (called *prediction horizon*) is then predicted according to some mathematical model; if track conflicts are detected a new complete traffic plan is then computed. This procedure is then iterated over time in a fashion that can be time-driven or event-driven. In real systems, the effectiveness of these plans can be strongly compromised if a large deviation between the actual and the predicted behaviour is observed, due to stochastic and dynamic evolution of traffic. This might lead to a nervous behaviour of continuously changing solutions, that is not well accepted by human dispatchers and practitioners. For this reason, particular attention must be paid to the stability of rescheduling plans. A plan is defined as stable when its (initial) structure is invariant to random perturbations occurring on the network within a given time period Δt . In other words, the initial actions of a stable control strategy will remain the same even if computed at a Δt later, with respect to updated traffic information. Understanding the stability of dispatching solutions in a rolling horizon framework is therefore a preliminary task for an effective control plan of real railway operations. So far, only little research has tackled this issue (e.g. [15], [20]), given the lack of advanced decision support tools in the railway industry and the unavailability of optimal rescheduling models (proposed in the academic literature) interfaced with the real field or with accurate simulation models that could represent a valid surrogate of the reality.

To this purpose, this paper studies the stability of real-time schedules of railway traffic within a dynamic and stochastic environment. An innovative framework is developed which integrates the Alternative-Graph based tool ROMA [9] for computing optimal rescheduling plans with a stochastic microscopic model for simulating railway traffic, EGTRAIN [23]. The investigation has been performed by considering a rolling horizon approach and referring to a single perturbed traffic scenario as a random realization of the reality. At regular time intervals updated traffic information is gathered from the simulation model (considered here as the real field) and transferred to the rescheduling tool to compute optimal plans. Solutions obtained at successive stages are compared for three relevant locations in the network to understand how optimal control plans change over time due to random evolution of train dynamics. The comparison is based on different indicators: the sequence of trains at a certain location, the amount of trains that are reordered with respect to the previous solution, the average amount of time shift compared to the original timetable (retiming), and the total number of reordering instructions that the dispatcher would give to trains, if he implemented the entire optimal control strategy. This study has been then repeated for different prediction horizons in order to comprehend how relevant this parameter is with respect to the stability of optimal plans. The proposed methodology has been applied to a real case-study in the Netherlands: the railway corridor between Utrecht and Den Bosch. Results show the effectiveness of the developed framework and the usefulness of the proposed methodology to analyze the stability of optimal plans under a stochastic and dynamic environment.

In the following section a literature review on rescheduling methods and stability analysis of dispatching plans is provided. A description of the framework developed is given in section 3. Section 4 illustrates the methodology adopted to perform the stability analysis, while the application to a real case study and relative results are reported in section 5. Conclusions are supplied in section 6.

2 Literature review

In literature, works addressing the stability of rescheduling plans are mostly concerning the management of activities in a job-shop or manufacturing environment. Here the stability is measured by means of the number of rescheduling instructions that must be taken to implement a control strategy [5], or by means of the number of jobs processed on different machines in the initial and the new schedule [1], or also considering the deviation of job starting times (retiming) and job sequences between the original and the revised schedules ([29], [8]). Several authors ([8], [14], and [16]) proposed a method for the dynamic or stochastic scheduling problem addressed to minimize the makespan and the deviation from the initial schedule considering a bi-criteria objective function that simultaneously takes into account the efficiency and the stability of rescheduling plans. Vieira et al. [28] determined the existence of a conflict between avoiding setups (as a metric of stability) and reducing flow-time (metric of efficiency). The rescheduling interval significantly affects the above objectives, as also concluded in [5], [16] and [24]. In their study, Mehta and Uzsoy [19] and Cowling and Johansson [8] indicate that schedules that are robust to stochastic disturbances can be generated without a lot of degradation of system performance. Bidot et al. [2] conclude that while the length of rescheduling intervals decreases the selected performance metric (makespan) improves.

In the field of railway traffic management most approaches that have been proposed focus on efficiently generating optimal schedules to minimize train delays, through an open-loop optimization process which involves a variety of assumptions on objectives and certain and deterministic conditions. Macroscopic approaches have been proposed by Carey and Lockwood [4] who developed an iterative decomposition approach for solving the train timetable and path problem in a railway network with one-way and two-way tracks. Higgins et al. [13] introduced a complex nonlinear mixed-integer program that incorporates the lower and upper limits on speed for each train on each segment, with the aim of minimizing the total train tardiness and fuel consumption. Dorfman and Medanic [11] obtain time-efficient and energy-efficient suboptimal schedules based on the concept of a local greedy travel-advance strategy and using a discrete-event model, relying on train priority rules and a capacity check algorithm to prevent deadlocks.

Törnquist and Persson [26], who represented the train rescheduling problem as a mixed-integer linear program, considering the network as divided in different segments composed of n parallel tracks. Near-optimal solutions are identified by means of a heuristic approach which is proved to be efficient also for large scale scenarios. Microscopic models are instead proposed by Mazzarello and Ottaviani [18] who set up an advanced real-time traffic management tool which consists of a speed regulation module (SPG), and a train scheduling and routing module (CDR) based on an alternative graph representation of train movements and responsible for generating conflict-free schedules. D'Ariano et al. [9], [10] developed an alternative-graph based model, called ROMA (Railway traffic Optimization by Means of Alternative graphs) which formulates the train rescheduling problem as a job shop with no store constraints. The rescheduling solution which minimizes the maximum consecutive delay on the network is identified by means of three greedy heuristics and a truncated version of a branch and bound [10], while a tabu search algorithm is adopted for finding suitable alternative routes for the rerouting problem (Corman et al. [7]). Lüthi [17] analyzes how advanced decision support tools for rescheduling traffic could be integrated into real railway operations, and defines schematic frameworks to practically implement a closed-loop integration. Caimi et al. [3] propose a dispatching assistant in the form of a model predictive control for complex station areas.

The closed-loop discrete-time system advises rescheduling trains according to solutions of a binary linear optimization model addressed to maximize customer satisfaction. The rescheduling module interfaces with an approximated model for calculating train trajectories; that is why this framework neglects some practical aspects that can only be considered by using very precise simulation models.

Only little work has been addressed so far to analyze the stability of dispatching plans. Meng and Zhou [20] study the robustness of a meet-pass plan for a disrupted single-track rail line, against random variations both in the running times and in the duration of the disruption. A macroscopic stochastic programming model is used in a rolling horizon framework to identify robust schedules for every roll period. The quality of the solution is then evaluated to understand how it is affected by the accuracy of the information on the disruption duration. Lee and Ghosh [15] study the stability of a decentralized algorithm (RYNSORD) with soft reservation for efficient scheduling and congestion mitigation. The investigation adopts a large-scale simulation model to generate perturbed traffic conditions. A steady-state operating point is initially identified, and the stability analysis is conducted by examining whether and when the system returns to the previous steady-state condition under a disturbed scenario. Törnquist[27] investigates how the quality of optimal rescheduling solutions is affected by the objective function and the length of the prediction horizon used in the rescheduling tool. The heuristic approach HOAT is used to reschedule traffic on a Swedish network under several perturbed scenarios.

The major drawbacks of these works are that they: *i*) only analyze the quality of optimal rescheduling solutions but do not study their variation over time when computed within a realistic closed-loop scheme; *ii*) refer to macroscopic, static, deterministic, and/or approximated models. This means that on one hand the models proposed in literature are not able to accurately reproduce system dynamics (resulting from the interaction of train speed profiles, delays and signalling system) and local stochastic phenomena that are inherently present in the real system and responsible for local solution instability. This is indeed a relevant issue to understand how optimal plans can perform during real-time operations. On the other hand, there is the necessity to interface models for the optimal rescheduling in a closed-loop with simulation environments that can represent a valid substitute of the railway field. In this way it is possible to direct the research towards the realization of on-line Decision Support Systems based on advanced algorithms, that can be actually implemented in the real field for a more effective management of real-time traffic perturbations. This issue is one of the objectives of the European FP7 project ONTIME [21] that involves a large group of infrastructure managers and universities of several European countries.

3 A framework for the stability analysis of rescheduling solutions

3.1 The framework

To analyse the stability of optimal rescheduling solutions an innovative framework has been set up which connects the rescheduling tool ROMA to a detailed microscopic model for the stochastic simulation of railway traffic, EGTRAIN, that is here considered as if it is the real field. Figure 1 illustrates the SysML representation of this framework, showing the input-output communication between the rescheduling and the simulation modules. Both these modules need to be initialized by specifying all the input data relative to the infrastructure, the rolling stock characteristics, the signalling and the ATP features, the original timetable, the entrance train delay (intended as the delay suffered from trains when they enter the network area under examination), as well as the stochastic

disturbances considered during the microscopic traffic simulation.

At regular time intervals, current traffic information (positions and speeds of trains) are transferred from EGTRAIN as input to the ROMA tool. Such information is used by the Track Conflict Detection module as the basis for forecasting traffic conditions over the prediction horizon, in order to identify potential track conflicts. The blocking time theory [12] is adopted for this purpose. This means that a track conflict is detected when an overlap between the blocking times of two trains is observed for a certain block section. The traffic prediction performed by this module is based on deterministic process times (i.e. running times and dwell times based on the train dynamics infrastructure characteristics, and published plans) without taking into account random disturbances to train operations. If no track conflicts are identified, the current schedule (that can be the original timetable if no dispatching actions have been selected) can be still operated without any modification. Otherwise, the detected conflicts (described by their space-time coordinate of the conflicts and IDs of conflicting trains) are sent as input to the Track Conflict Resolution module. This module computes a new conflict-free plan (rescheduling solution) obtained by retiming/reordering and/or rerouting trains in order to minimize the maximum consecutive delay on the network.

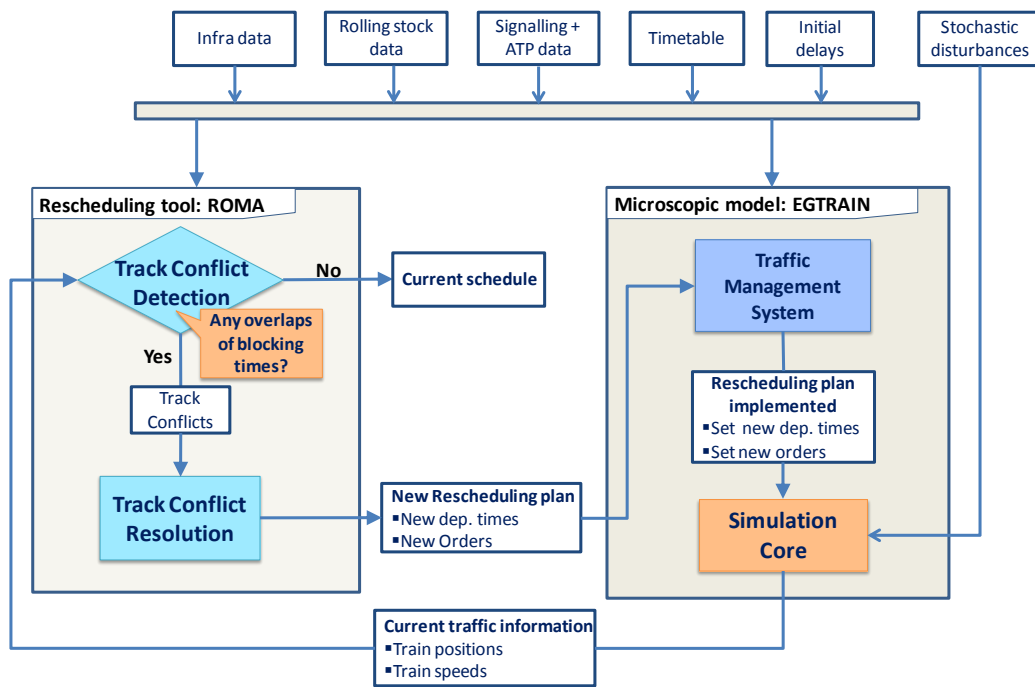


Figure 1: SysML representation of the framework for the stability analysis of rescheduling solutions.

The updated departure times and train orders provided by the rescheduling plan are sent as input to the Traffic Management System module of the EGTRAIN model. Departure times and orders given by the rescheduling tool are considered as hard constraints when implemented in the simulation environment.

Railway traffic is microscopically simulated by the EGTRAIN simulation core according to the computed real-time plan, and considering additional stochastic disturbances to train operations. In particular, entrance train delays and perturbations to dwell times at stations are considered. A more detailed description of both the rescheduling and the simulation modules is respectively provided in the following subsections.

3.2 The detection and resolution model ROMA

Apart from a data loading and a post processing phase, the dispatching problem conceptually defines two problems, namely a conflict detection problem and a conflict resolution problem. For the solution of both problems alternative graphs are used, which are composed by nodes representing train operations (passage of trains over a block section) and directed arcs connecting two nodes (e.g. arc (i,j) connects node i with j). Fixed arcs (*Fix*) constrain successive operations of the same train to be separated by a given process time w_{ij} ; while Alternative arcs (*Alt*) separate potentially conflicting operations of different trains over shared block sections, by given headways w_{ij} . In a compact manner, an alternative graph can be represented as follows:

$$t_j - t_i \geq w_{ij} \text{ for all } (i, j) \in \text{Fix},$$

$$(t_j - t_{s(i)} \geq w_{ij}) \vee (t_i - t_{s(j)} \geq w_{ji}) \text{ for all } ((s(i), j), (s(j), i)) \in \text{Alt}.$$

where the variable t_i , for $i = 1, \dots, n - 1$, represents the starting time of operation i and corresponds to the entrance time of a train in the associated block section. The first group of constraints models fixed order constraints such as the running of trains over successive sections, dwell processes, entrance in the network. The second kind of constraints models the choice between the two orders of potentially conflicting trains. Each order can be selected by choosing a particular side of the disjunction. The notation $s(i)$ refers to the successor of operation i on the path of a train. Further details on alternative graph models can be found in [7].

Given a prediction horizon, the entrance times of trains, and suitable weights for the time durations w_{ij} (considered as deterministic), the conflict detection task is performed by visiting the graph. This means that starting times are associated to each operation, while not enforcing yet the orders of trains. A conflict is detected if the computed starting times lead to overlaps between blocking times of different trains.

The conflict resolution task is tackled instead by solving the mathematical problem of choosing a particular side of the disjunction, i.e., a particular train order. The scheduling problem defined is solved within ROMA by a Branch and Bound (BB) scheduling algorithm. This is an exhaustive algorithm that explores all the reordering alternatives and chooses the one minimizing the maximum consecutive delay. Here a truncated branch and bound [10] is considered that returns near-optimal schedules for practical size problems within a short computation time. A typical objective function of the conflict detection and resolution (CDR) problem is the minimization of the maximum consecutive delay, which can be computed easily on an alternative graph.

Running and headway times are a function of the speed profiles of trains, which again depend on the ordering decisions taken. Thus a speed adjustment phase is performed on the scheduling solution to deliver a feasible train plan.

3.3 The stochastic microscopic simulation model EGTRAIN

A “self-built” microscopic model [23] has been used in order to overcome applicability

limits of commercial models that do not permit an easy manipulation of all network characteristics and train parameters, or to be interfaced with mathematical tool-boxes for “black-box” optimization. In particular EGTRAIN (Environment for the desiGn and simulaTion of RAILway Networks) is an object-oriented model developed in C++, representing in detail attributes of each element of the railway network. It consents to accurately describe the dynamic evolution of train state variables (e.g. speed and position) considering both planned and perturbed traffic.

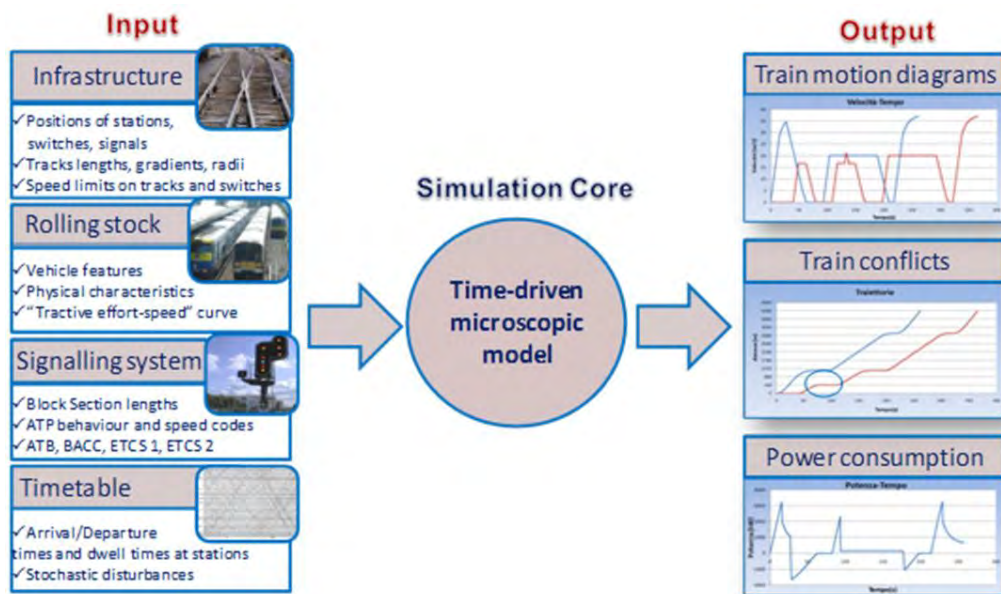


Figure 2: Input-output of the microscopic railway simulation model EGTRAIN

It is a time-driven, synchronous, microscopic and stochastic multi-train simulation model which administers input data in the following four interacting modules:

- *Infrastructure module.* The railway network is here modelled as a directed graph with arcs containing all information about track attributes (e.g. gradients, radii, speed limits on tracks and switches), while details about spatial coordinates of signals, switches and stations are assigned to nodes.
- *Rolling stock module.* Tractive effort-speed curve of the traction unit, maximum deceleration rate, jerk value, as well as train composition (number of wagons, masses of the coaches, etc.) are all input data of this module. A sub-module for the calculation of train energy consumption is also included.
- *Signalling system module.* In this module, interactions between rail vehicles and signalling equipments are modelled, for different signalling systems and ATP characteristics. The positions of signals and beacons, the length of block sections, as well as the braking behaviour and the speed codes of the ATP system are all input to this module. The Dutch NS'54 speed signalling system with ATB train protection is implemented for this study.

- *Timetable module.* Inputs required by this module are departure/arrival times at stations, but also minimum dwell times can be specified. Stochastic train delays can also be considered by specifying the type and the parameters of the respective distribution function.

The train movements along the track are obtained by integrating the Newton’s motion formula over time: for each time step, the maximum surplus force between the traction unit’s wheels and the tracks is calculated to determine the acceleration function which is then integrated a first time to provide speed function and a second time to provide train position. Output data returned by this model are:

- *Train diagrams* (e.g. distance-time trajectories, speed-distance diagrams, etc.),
- *Train conflicts* (e.g. conflicts due to blocking time overlaps)
- *Energy consumption diagrams* (e.g. mechanical power-time diagrams, mechanical energy-distance diagrams).

A successful validation process of the model has already been performed [23], verifying the congruence of its outputs with those returned by the consolidated microscopic model *OpenTrack* for the same input data set.

4 Methodology

4.1 The rolling horizon approach used for the stability analysis

To solve the traffic scheduling problem within a dynamic and stochastic environment a rolling time horizon approach has been adopted. This approach decomposes the dynamic (i.e. with information and status known only as time goes by) scheduling problem into a number of static (i.e. deterministic and with complete information) sub-problems that can be solved by using static scheduling methods. A schedule is produced for each sub-problem and is then regularly updated by taking into account fresh information from the field.

Figure 3 represents the case of a small network with 3 trains (V1, V2, V3) that in case of perturbation may need to be reordered at their passage from signal CP. The total observation period H is decomposed in multiple time periods named “Stages”, which are partially overlapping and spaced at regular time intervals, namely the rescheduling intervals (RI). At the beginning of each stage (i.e. at times t_1, t_2, t_3, \dots) current traffic information (e.g. train positions represented by red circles) is collected from the field and set as input to the rescheduling tool. This latter, on the basis of the information received, predicts how traffic will evolve over the prediction horizon PH (respectively $PH1, PH2, \dots$), detecting the presence of potential conflicts by means of the blocking time theory. If conflicts are detected, a new rescheduling plan is generated. The rescheduling plan is provided with a certain delay (d_1, d_2, d_3, \dots) from the beginning of the stage due to the computation time of the process. For an effective real-time management it is necessary that this computing time is always acceptably shorter than the RI . After each rescheduling interval a new stage is activated and a new plan is produced on the basis of updated traffic information. In a closed-loop control framework, a new rescheduling solution should be implemented each time that this latter differs from the one computed at the previous stage.

In this paper, we assume that only the plan computed at the first stage is implemented while the other successive solutions are computed but not put into operation. In this way we can separate the different factors of the dynamic control process and observe how the solutions change over time as effect of the stochastic perturbations only. This assumption

represents the case of a dispatcher that implements only the optimal plan computed at the first stage, but observes how successive solutions, elaborated on the basis of current traffic information, vary over time.

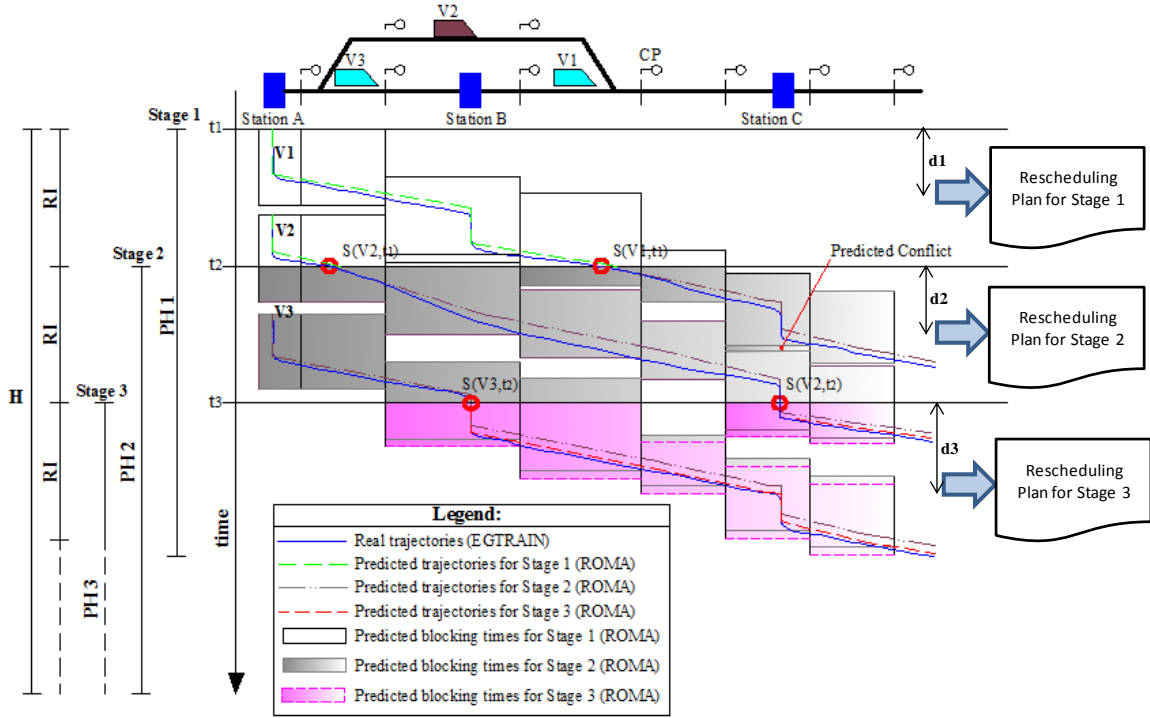


Figure 3: Rolling time horizon approach for the stability analysis of rescheduling solutions

The methodology applied to analyse the stability of traffic rescheduling plans is described as follows:

- Implementation of the solution computed at the first stage.* At stage 1, train entrance delays are set as input to the ROMA tool which predicts the traffic evolution over the prediction horizon (considering deterministic train running and dwell times), detects possible conflicts due to such delays, and provides a first rescheduling solution obtained by retiming/reordering trains. This plan is then implemented in the EGTRAIN model by means of the Traffic Management System module. During the entire observation period, the traffic will therefore follow this rescheduling plan. For the example in Figure 3 this means that at time t_1 the rescheduling tool is fed with train entrance delays (current train information is not available since trains have not entered the network yet) and computes a solution. This solution is implemented in the field and imposes the order (V1, V2, V3) at signal CP. In fact, real trajectories (solid blue lines) follow this plan.
- Generation of the base-case perturbed scenario.* At this point, the microscopic model EGTRAIN is activated to simulate railway traffic according to the obtained rescheduling solution, taking into account both train entrance delays and additional stochastic disturbances to dwell times at stations. These latter are considered as independently and identically distributed Gaussian variables. In the current

preliminary study, dwell times are obtained by considering only a single random sample. Rescheduling plans are computed for each stage taking this realization as base-case. Further research would investigate the impact of different random samples and distributions. In Figure 3, perturbations determine the difference between the predictions (dotted lines) and real trajectories (solid blue lines).

- c) *Calculation of the solutions at successive stages.* Iteratively, at the beginning of each successive stage, train speeds and positions are gathered from EGTRAIN and set as input to the ROMA tool. Based on this information, ROMA forecasts traffic behaviour assuming deterministic running and dwell times, and checks the presence of potential conflicts. In this case a conflict-free schedule is generated by retiming and reordering trains. To decouple the dynamics of prediction and control, these plans are not implemented in EGTRAIN but only compared to analyze how they vary as consequence of unplanned dynamics. In the example, at time t_2 current speeds and positions (red circles) of trains V1 and V2 are collected from the field and sent to ROMA. A deviation between real and predicted (dotted grey line) trajectories is observed, that increases with the flow of time due to propagation of random disturbances. At time t_3 ROMA is fed with updated traffic information which allows the prediction to be adjusted and obtain a more accurate detection of track conflicts. In this case the deviation between real and forecast trajectories (dotted red line) is reduced with respect to the previous stage.
- d) *Solution comparison.* The solutions returned by ROMA for each consecutive stage are analyzed to understand how they differ from each other. In particular, they are compared in terms of train orders at given locations of the network (called checkpoints). A retiming metric measures for each checkpoint the average extent of train retiming with respect to the original schedule. The number of reorderings with respect to the previous solution are also calculated for each plan. This metric highlights the amount of different instructions that would be given from the dispatcher to trains within each rescheduling interval, giving also insights in the practical feasibility of the computed plans. All these metrics are better described in the following section.

A further investigation is then conducted to comprehend how the stability of rescheduling solutions is affected by the length of the prediction horizon used to foresee track conflicts. This means that the steps described above are repeated for different lengths of the prediction horizon.

4.2 Metrics used to compare rescheduling solutions

The comparison among the rescheduling plans computed for each stage is performed by evaluating the differences in both the train orders at given checkpoints and the retiming with respect to the original timetable for the same checkpoints. In particular the following metrics are considered:

- *ID number of the solution (Solution ID).* Rescheduling solutions provide for each checkpoint a permutation of trains that sets the order in which they should pass from that checkpoint. A positive integer number $ID \in [0, N]$ is assigned to each different permutation of trains, in order to uniquely identify them. Solutions giving the same order are identified with the same ID. As can be seen in Table 1 three different IDs are assigned to three different permutations respectively: 1 to (V1, V2, V3), 2 to (V2, V1, V3) and 3 to (V3, V1, V2). For certain checkpoints, it can happen that the rescheduling plans computed at a given stage provide ordered lists in which some

trains are missing. This can occur if *i*) trains are not scheduled to pass from that location, *ii*) they have already passed that checkpoint, *iii*) they enter the network after the end of the selected prediction horizon, *iv*) they have already finished their run. In this case, if these lists do not present a new permutation of elements, they receive the same ID of the list that is equal when deleting from it the missing trains. For example the ID = 3 is assigned to the list (V3, V1) in which train V2 is missing. This is because such sequence is obtained from the solution with ID = 3 by deleting train V2. Of course this assumption is valid as far as cancelling trains is not used as a control action.

Table 1: Evaluation of rescheduling plans for the case of Figure 3.

<i>Solution ID</i>	<i>Train Order at CP</i>	<i>Sequence number stage i (a_i)</i>	<i>Sequence number stage i-1 (b_i)</i>	<i>N° relative reordering (NRR)</i>
1	V1	1	1	0
	V2	2	2	
	V3	3	3	
2	V2	1	2	1
	V1	2	1	
	V3	3	3	
3	V3	1	3	2
	V1	2	2	
	V2	3	1	

- *Number of Relative Reordering (NRR)*. For a rescheduling solution produced at stage *i*, this metric gives the total number of trains that have been reordered with respect to the previous solution returned at stage *i-1*. This indicator is frequently used in the field of telecommunication engineering when analyzing the order of packets in digital streams over computer networks [22]. In general an element is defined as reordered if it has a sequence number smaller than its predecessors. The sequence number a_j assigned to each train j of a list of K trains is a consecutive integer from 1 to K which follows the same order provided by the solution computed at stage *i-1*. The sequence of trains (b_1, \dots, b_K) , observed for the plan produced at the following stage *i*, is said in the same order of the previous solution if

$$b_k < b_q \quad \forall k \in [1, K], \quad 0 < q < K \quad (1)$$

otherwise it is said out-of-order and train k is defined as reordered. In this case the total number of reordered trains corresponds to the Number of Relative Reordering and is indicated by *NRR*. An example is shown in Table 1. If the sequence number a_j attributed to trains (V1, V2, V3) follows the order of solution 1 (i.e. with ID = 1), the number of reordering relative to solution 2 is 1, since train V1 does not satisfy condition (1) and is therefore reordered with respect to solution 1. In the same way if the sequence number a_j follows the order given by solution 2, the amount of reordering relative to plan 3 is 2 because both trains V1 and V2 are reordered.

- *Number of reordering instructions (NRI)*. The sum of the *NRR* over all the rescheduling stages gives the total number of reordering instructions that the dispatcher would provide to trains if he implemented the solution of each consecutive

stage. If NRR_s is the number of relative reorderings corresponding to the plan computed at stage s and S is the total number of stages, the number of reordering instructions NRI is calculated as:

$$NRI = \sum_{s=1}^S NRR_s \quad (2)$$

In the following of this paper this metric is referred to a period of 1 hour since the rescheduling is performed over such time span.

- **Average Retiming (AR).** This metric measures, for a given location, the deviation between the passage times given by the rescheduling plan and the ones established by the original schedule. The retiming metric can be expressed as:

$$AR = \frac{1}{K} \sum_{j=1}^K (T_{pass_{resch,CP_j}} - T_{pass_{TT,CP_j}}) \quad (3)$$

where $T_{pass_{resch,CP_j}}$ represents the time instant at which train j is scheduled to pass at location CP_j according to the rescheduling plan, $T_{pass_{TT,CP_j}}$ is the passage time established from the original timetable (TT) and K is the total number of trains passing at CP_j within the selected prediction horizon.

5 Case study: the Utrecht - Den Bosch corridor

5.1 Case-study description

The proposed framework has been applied to a real-case study: the double-track corridor between the cities of Utrecht and Den Bosch in the Netherlands. This corridor has a length of more than 48 km with a total of 8 stations. The schematic layout is illustrated in Figure 4. The network is equipped with a fixed-block signalling system and the traditional Dutch automatic train protection ATB system (see [25] for more information) whose behaviour and characteristics have been implemented in both ROMA and EGTRAIN.

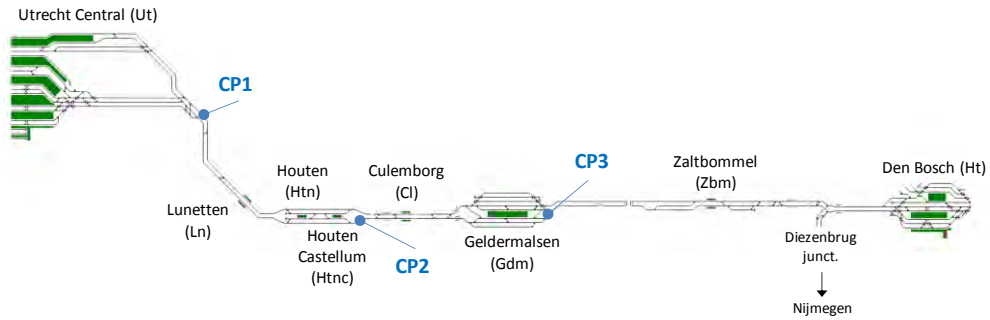


Figure 4: Schematic layout of the Utrecht - Den Bosch corridor (as given by InfraAtlas 2011), with the locations (CP1, CP2 and CP3) in which train reordering is considered.

This was necessary to accurately model train movements on the track accordingly with the rules imposed by these systems.

The hourly traffic pattern on this corridor is composed per each direction of: 4 Intercity trains (IC) between Utrecht (Ut) and Den Bosch (Ht) that stop only in Ut and Ht; 2 regional trains (Reg) between Ut and Ht stopping at all stations; 2 regional trains operating between Utrecht and Geldermalsen (Gdm) and also stopping at Lunetten (Ln), Houten (Htn), Houten Castellum (Htnc), and Culemborg (Cl). Two intercity and regional trains operating between Nijmegen and Den Bosch are also scheduled to run between the Diezenbrug junction (where the branch from Nijmegen merges) and Ht, but these trains are not taken into account in the analysis. Also freight trains are neglected. For the sake of simplicity, only trains running along the Ut - Ht direction are considered. This assumption does not alter the results of the investigation since in this double-track corridor there is no interaction between trains running in opposite directions. The train services are operated according to a periodic timetable with one hour period. The scheduled minimum dwell time for the ICs is 120 s at both Ut and Ht. For the regional trains it is 120 s for Ut and Ht, 240 s for Gdm (where an overtaking is scheduled) and 24 s for all the other stops. Traffic is observed for a total period of 2 hours while the rolling horizon approach described in section 3 is analysed only for the first hour. A total number of 16 trains is considered during the entire observation period. Train names are shown in Table 2. The ID of each train is obtained by adding to its name a prefix that is equal to 1 if it is scheduled to depart during the first hour and 2 if departs within the second hour.

Table 2: Train names used for trains running along the Ut- Ht direction

Train Name	Category	Stops
D8001, B8001	IC	Ut, Ht
D35001, B35001	IC	Ut, Ht
D160001, B160001	Reg	Ut, Ln, Htn, Htnc, Cl, Gdm, Zbm, Ht
D60001, B60001	Reg	Ut, Ln, Htn, Htnc, Cl, Gdm,

Trains can overtake each other at three different locations where ICs and regional trains run on different routes: at the exit of the Utrecht station area (CP1), at the exit of the Houten interlocking area (CP2) and at the exit of the Geldermalsen station (CP3). These locations constitute therefore the checkpoints where train reordering can be applied by real-time plans.

A single disturbed scenario is examined, which is obtained by drawing a single random sample of entrance delays and dwell times at stations respectively from a Weibull and a Gaussian distribution (μ , σ). In particular for the dwell time distribution the mean μ is the scheduled dwell time, while the standard deviation σ is assumed to be the 30% of the mean ($\sigma = 0.3 \cdot \mu$). At the first stage a rescheduling solution is elaborated by ROMA on the basis of train entrance delays and is implemented in EGTRAIN.

Figure 5 shows the base-case (as simulated by EGTRAIN) used for the analysis, where traffic follows the plan given at the first stage under the selected disturbed conditions. Some trains are forced to stop and wait at some locations to give way to late trains and respect the order given by the plan computed at the first stage. For example this is the case of the IC 1D35001 that is forced to stop at the location Htnc in order to give way to the delayed Reg 1D160001 and satisfy the rescheduled order for the checkpoint CP2. Other trains instead are forced to stop or slow down because of track conflicts (e.g. IC 1D35001 slows down before Cl).

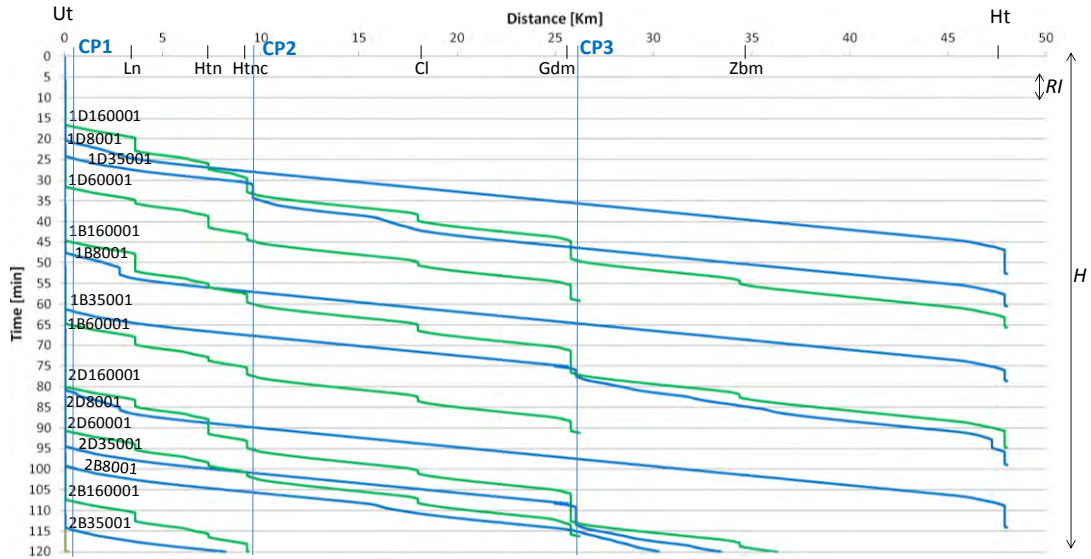


Figure 5: Traffic behaviour simulated by EGTRAIN following the first rescheduling solution given by ROMA. ICs are reported in blue, regional trains in green.

The study has also investigated how the stability of optimal rescheduling plans is influenced by the length of the prediction horizon used to compute solutions. Four different lengths have been considered: 15, 60, 30 and 90 minutes. A total of four experiments (one for each value of the prediction horizon) have been therefore carried out. The rescheduling interval is always set to 5 minutes. This means that for each experiment, 12 successive stages having the same length of the prediction horizon are considered

5.2 Results

A total of 13 different permutations have been observed over the whole analysis. The ID assigned to each one of these permutations is reported in Table 3.

Table 3: ID assigned to the permutations returned by rescheduling solutions

0		1		2		3		4		5		6		7		8		9		10		11		12	
s_{TT}	Train name	s_{TT}	Train name	s_{TT}	Train name	s_{TT}	Train name	s_{TT}	Train name	s_{TT}	Train name	s_{TT}	Train name	s_{TT}	Train name	s_{TT}	Train name	s_{TT}	Train name	s_{TT}	Train name	s_{TT}	Train name	s_{TT}	Train name
1	1D8001	2	1D160001	2	1D160001	2	1D160001	3	1D35001	1	1D8001	1	1D8001	1	1D8001	1	1D8001	1	1D8001	3	1D35001	1	1D8001	1	1D8001
2	1D160001	1	1D8001	1	1D8001	1	1D8001	2	1D160001	3	1D35001	3	1D35001	3	1D35001	3	1D35001	3	1D35001	1	1D8001	2	1D160001	2	1D160001
3	1D35001	3	1D35001	3	1D35001	3	1D35001	1	1D8001	2	1D160001	2	1D160001	2	1D160001	2	1D160001	2	1D160001	2	1D160001	3	1D35001	3	1D35001
4	1D60001	4	1D60001	4	1D60001	4	1D60001	4	1D60001	4	1D60001	4	1D60001	4	1D60001	4	1D60001	4	1D60001	4	1D60001	4	1D60001	4	1D60001
5	1B8001	6	1B160001	6	1B160001	6	1B160001	5	1B8001	5	1B8001	5	1B8001	5	1B8001	5	1B8001	5	1B8001	5	1B8001	5	1B8001	5	1B8001
6	1B160001	5	1B8001	5	1B8001	5	1B8001	6	1B160001	6	1B160001	6	1B160001	6	1B160001	6	1B160001	6	1B160001	6	1B160001	6	1B160001	7	1B35001
7	1B35001	7	1B35001	7	1B35001	7	1B35001	7	1B35001	7	1B35001	7	1B35001	7	1B35001	7	1B35001	7	1B35001	7	1B35001	7	1B35001	6	1B160001
8	1B60001	8	1B60001	8	1B60001	8	1B60001	8	1B60001	8	1B60001	8	1B60001	8	1B60001	8	1B60001	8	1B60001	8	1B60001	8	1B60001	8	1B60001
9	2D8001	10	2D160001	10	2D160001	9	2D8001	10	2D160001	9	2D8001	9	2D8001	9	2D8001	9	2D8001	9	2D8001	11	2D35001	9	2D8001	9	2D8001
10	2D160001	9	2D8001	9	2D8001	10	2D160001	11	2D35001	13	2B8001	13	2B8001	10	2D160001	10	2D160001	11	2D35001	9	2D8001	13	2B8001	13	2B8001
11	2D35001	12	2D60001	13	2B8001	13	2B8001	9	2D8001	10	2D160001	10	2D160001	13	2B8001	11	2D35001	10	2D160001	10	2D160001	10	2D160001	10	2D160001
12	2D60001	11	2D35001	12	2D60001	12	2D60001	12	2D60001	12	2D60001	11	2D35001	11	2D35001	12	2D60001	12	2D60001	12	2D60001	11	2D35001	11	2D35001
13	2B8001	13	2B8001	11	2D35001	11	2D35001	13	2B8001	11	2D35001	13	2B8001	12	2D60001	13	2B8001	13	2B8001	13	2B8001	12	2D60001	12	2D60001
14	2B160001	14	2B160001	14	2B160001	14	2B160001	14	2B160001	14	2B160001	14	2B160001	14	2B160001	14	2B160001	15	2B35001	15	2B35001	15	2B35001	15	2B35001
15	2B35001	15	2B35001	15	2B35001	15	2B35001	15	2B35001	15	2B35001	15	2B35001	15	2B35001	15	2B35001	14	2B160001	14	2B160001	14	2B160001	14	2B160001
16	2B60001	16	2B60001	16	2B60001	16	2B60001	16	2B60001	16	2B60001	16	2B60001	16	2B60001	16	2B60001	16	2B60001	16	2B60001	16	2B60001	16	2B60001

The sequence number s_{TT} used to categorize permutations follows the order given by the original timetable. This order is identified with ID = 0. These are permutations of the complete list of elements, since they include all the 16 trains monitored during the entire observation period.

For each different value of the prediction horizon, the plan computed at the first stage and implemented in EGTRAIN has provided the permutations 1, 0, and 8 respectively for the checkpoints CP1, CP2, and CP3. Trains respect these orders in the base-case scenario considered for the analysis.

Results obtained are reported in Figure 6, Figure 7 and Figure 8, respectively for checkpoint CP1, CP2 and CP3. Part *a)* of these figures shows how the solution ID varies over time for different prediction horizons. It is worth noting that this represents only the ID of the permutation returned for a certain checkpoint, and it does not have any strictly numerical meaning. A larger difference in the ID of two solutions therefore does not mean a larger difference between their train orders. As can be seen, the rescheduling solution remains more or less the same for the first 20 minutes of operation, independently from the length of the prediction horizon used. Then, it starts to become unstable with changes that become more and more frequent with the passing of time. This is due to stochastic disturbances in the real field (EGTRAIN) that progressively propagate on the network, altering train running times. This induces a gap between real (produced by EGTRAIN) and predicted (given by ROMA) trajectories that leads to incorrect forecasts and uncertainty in the detection of track conflicts.

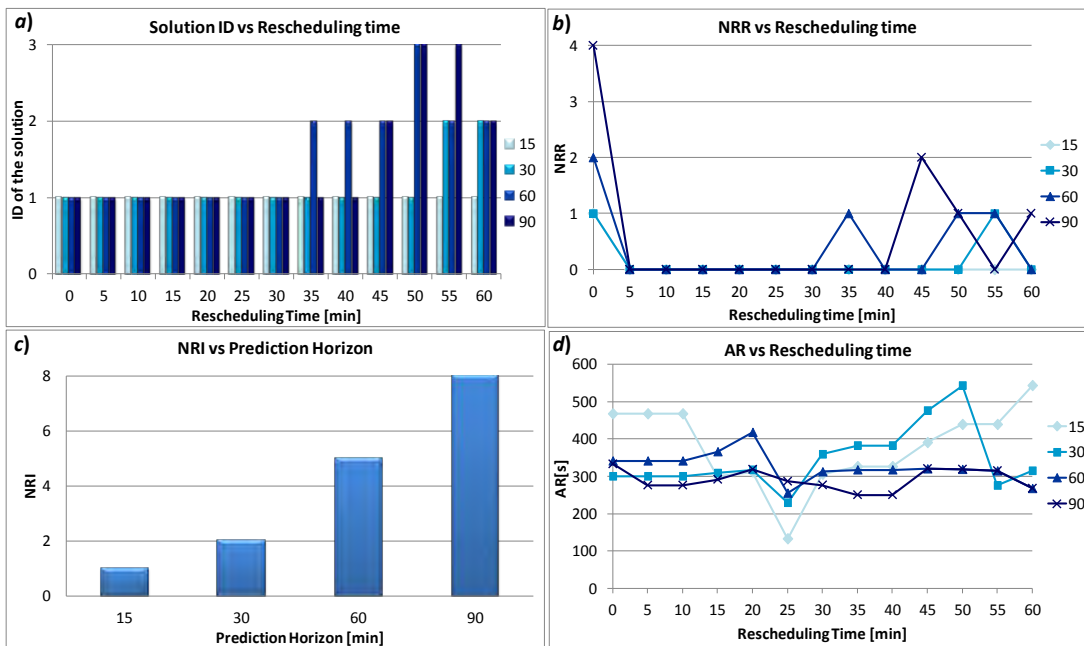


Figure 6: Results at checkpoint CP1 for different lengths of the prediction horizon.

In our test case, different conflicts are detected at consecutive stages and different plans are therefore computed.

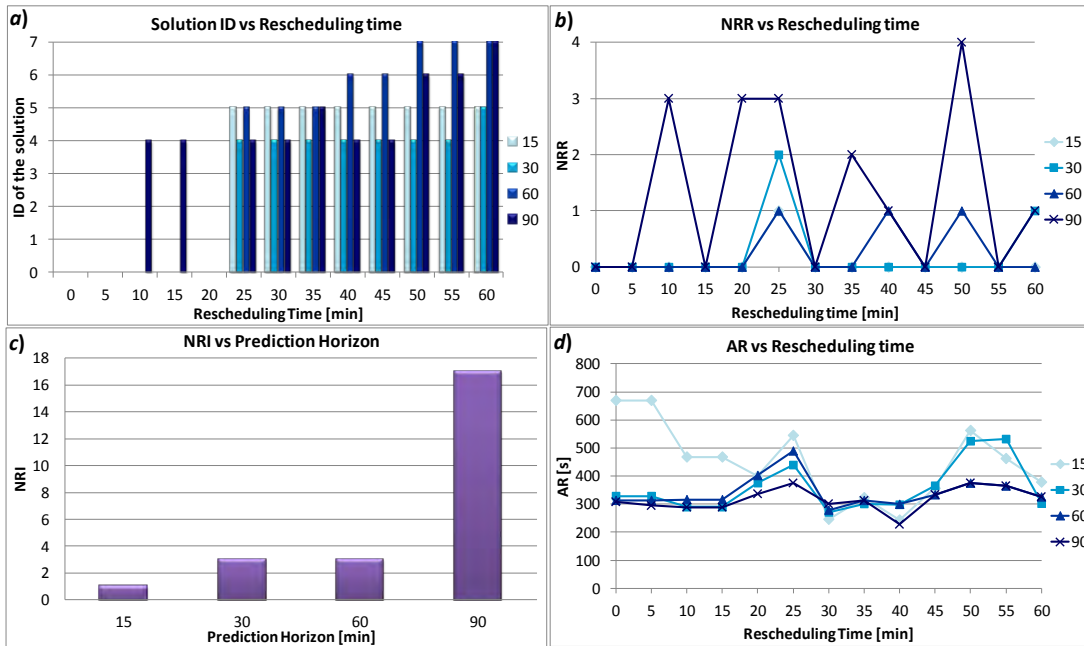


Figure 7: Results at checkpoint CP2 for different lengths of the prediction horizon.

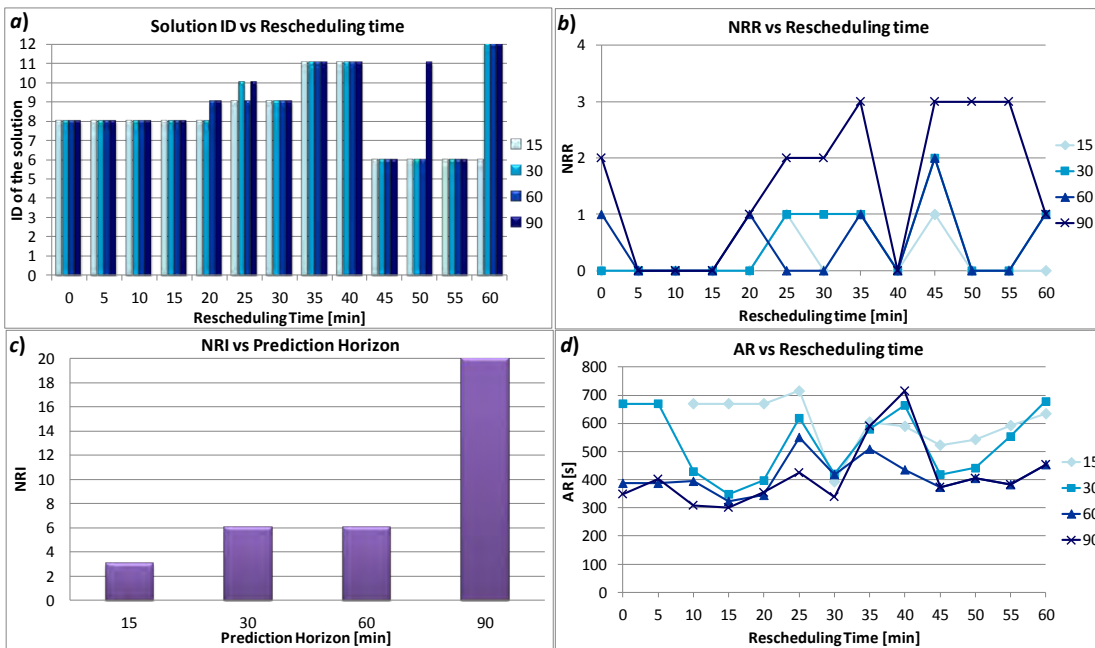


Figure 8: Results at checkpoint CP3 for different lengths of the prediction horizon.

This is the reason why oscillations of the rescheduling solution are observed in Figure 6a), Figure 7a) and Figure 8a). The effect is particularly relevant for longer prediction horizons. The motivation is that uncertainties in the prediction grow when traffic is foreseen over a wider time window ahead (as also stated in [27]), increasing the probability that the forecast is distant from reality. At a certain stage, trains that are already in the network must be predicted also for time instants that are quite far from the current time, with a higher probability that estimated trajectories will be different from the real ones. A large number of trains that have not entered the network yet are also forecast, with the disadvantage that their predictions cannot be adjusted accordingly to their current states, simply because this information cannot be obtained for these trains. These factors result in larger uncertainties when using longer prediction horizons.

The changes in the solution over time decrease instead progressively while shortening the prediction horizon since: *i*) trajectories of trains already in the network are predicted for a shorter period and can be adjusted more effectively according to their current states; *ii*) fewer trains that have not entered yet in the network are forecast. For these reasons it seems that shorter prediction horizons would result in an increased control stability.

This is also the conclusion when looking at the number of relative reordering NRR for different lengths of the prediction horizon (Figure 6b), Figure 7b) and Figure 8b)). For all the three checkpoints it is immediately seen that while shortening the prediction horizon, the value assumed by NRR at a fixed time instant decreases, and so does its variation over time. Figure 6c), Figure 7c) and Figure 8c) report instead for each checkpoint the number of reordering instructions that the dispatcher would provide over a period of 1 hour, if he implemented all the plans obtained at each stage. These figures show how the effort of the dispatcher in instructing trains with new orders varies if a different length of the prediction horizon is adopted for the rescheduling tool. For CP1 this effort presents an increasing trend with the length of the prediction horizon. An increasing trend is presented also for CP2 and CP3, with the particularity that the effort of the dispatcher is the same if using a prediction horizon of 30 or 60 minutes. This is due to the fact that for PH=60 min there is a smaller difference between consecutive solutions, although they change more frequently than in the case of PH=30 min (as shown by the variation of the solution ID in part a)). For this case study, the number of reordering actions tend to grow when longer PHs are set into the rescheduling tool. This can be explained by the fact that when using larger PHs, the rescheduling tool has a wider knowledge of both future traffic states (although only as expected value) and the time margins actually exploitable to mitigate the disturbance. Both these factors enlarge the domain of the possible rescheduling solutions and might lead to identify reordering as a more effective measure to manage perturbations. This outcome differs from what observed in [27] in which is stated that no significant difference in the solution is obtained when enlarging the PH over a certain threshold. Most likely such discordance is due to the consistent disturbances considered in our test field that differently from the case in [27], induce: key conflicts distributed over the entire observation period; perturbations with long term effects; and/or disturbances extended to more trains. The trend of NRI shown in part c) of the figures are strictly valid only for the perturbed scenario considered in this paper and cannot be generalized. A more reliable conclusion can be drawn only after having extended the analysis to a significant set of different disturbed scenarios. It is expected indeed that this latter kind of analysis returns outcomes that are in accordance with [27].

Such findings highlight anyway the importance that the analysis performed in this paper can assume also to design the parameters (e.g. rescheduling interval and prediction horizon) of an optimal control closed-loop setup, if the objective is a more stable

dispatching plan, rather than a control strategy that minimizes the effort of the dispatcher.

Figure 6d), Figure 7d) and Figure 8d) represent how train paths are averagely shifted in time (AR) with respect to the original timetable, for different prediction horizons. The trend shown by the indicator AR is opposite to the one observed for the metric NRR , given that when shortening the prediction horizon, the value assumed for a fixed time instant increases and also its instability over time grows. This is due to the fact that the rescheduling solution computed at a given time prescribes a train retiming that is more or less the same independently from the prediction horizon employed. This means that for a given rescheduling stage, the numerator of the AR metric remains more or less the same across different lengths of prediction horizon. The denominator instead decreases while shortening the horizon since a lower number of trains are considered in the forecast and therefore in the solution. That is why the value of AR increases for shorter time windows of the prediction. When the rescheduling time is around 30 min a similar value of AR is obtained for all the prediction horizons. This is only due to a local effect since at this stage some heavily retimed trains are not considered when shorter time windows are used for the forecast. For the case study examined in this paper, train retiming is therefore scarcely affected by the length of the prediction horizon, but it is more sensitive to the evolution of traffic behaviour over time, given the oscillations observed over all stages for a fixed value of PH . Retiming is therefore not as sensitive to the length of the prediction horizon as instead reordering is.

6 Conclusions

The implementation of automatic and advanced dispatching systems did not reach real operations yet. Optimal dispatching solutions can be in principle very sensitive to input data and be fragile when confronted with unplanned dynamics. When optimal solutions are embedded within an on-line closed-loop Decision Support System, this might lead to a stressed behaviour of continuously changing advices that would be hardly acceptable by human dispatchers and practitioners.

In this paper a stability analysis of optimal rescheduling plans is performed by adopting a rolling horizon approach within a stochastic and dynamic environment. An innovative framework has been used, which integrates the laboratory rescheduling tool ROMA with the stochastic microscopic railway simulation model EGTRAIN, which is here considered as a valid surrogate of the real field. The current analysis assumes that the dispatcher implements only the plan obtained at the first rescheduling stage, while the solutions of successive stages are computed on the basis of updated train information (speeds and positions) supposing a single randomly perturbed scenario. Consecutive optimal plans are then compared by a set of metrics, to understand how they change over time when additional stochastic disturbances to operations are considered. A further investigation is addressed to estimate how sensitive the stability of a control strategy is when varying the length of the prediction horizon in the rescheduling tool.

Results obtained for the case study Utrecht-Den Bosch show that longer prediction horizons result in more unstable control strategies. This is due to the uncertainty in the track conflict detection which increases when random disturbances are considered (since the probability that the traffic behaviour deviates from the prediction is higher). This aspect is observed from the trends of both the solution ID and the number of relative reordering NRR . When analyzing the number of reordering instructions, no general trend is identified that explains how the effort of the dispatcher varies when changing the

prediction horizon. For instance, even unstable (i.e. frequently changing) dispatching plans might result in low dispatcher effort, if the consecutive solutions computed are only slightly different. A study on a larger statistical sample and/or a specific detailed study on a case might help in understanding which value of PH minimizes the effort of the dispatcher. Train retiming is instead scarcely sensitive to variations of this parameter but mainly influenced by the propagation of stochastic disturbances over time.

The outcomes confirm the effectiveness of the developed framework and the usefulness of the proposed methodology to study the stability of optimal control plans during perturbed traffic operations. A similar investigation can also support the design of the parameters of the control framework (e.g. lengths of the rescheduling interval and the prediction horizon) in order to implement a more stable dispatching plan, rather than a control strategy that minimizes the effort of the dispatcher.

Future research will be addressed to extend this analysis to a significant number of randomly generated disturbed scenarios to draw conclusions that are independent from the case study considered. Then a complete closed-loop framework will be examined in which the entire control strategy (i.e. each consecutive solution computed at each stage) is implemented in the simulated field. This would allow not only the evaluation of the solution stability but also the effects on solution quality (delays, punctuality) when different perturbed conditions are observed.

7 References

- [1] Alagöz O., Azizoglu M., “Rescheduling of identical parallel machines under machine eligibility constraints”, *European Journal of Operational Research*, Vol. 149, 523–532, 2003.
- [2] Bidot J., Laborie P., Beck J.C., Vidal T., “Using Simulation For Execution Monitoring And On-Line Rescheduling With Uncertain Durations”, in: *Proceedings of the ICAPS’03 Workshop on Plan Execution*, Trento, Italy, 2003.
- [3] Caimi G., Fuchsberger M., Laumanns M., Lüthi M., “A Model Predictive Control Approach For Discrete-Time Rescheduling In Complex Central Railway Station Areas”, *Computers & Operations Research*, Vol. 39, 2578-2593, 2012.
- [4] Carey, M., Lockwood, D., “A model, algorithms and strategy for train pathing”, *Journal of the Operational Research Society*, Vol. 46 (8), 988–1005, 1995.
- [5] Church L.K., Uzsoy R., “Analysis Of Periodic And Event-Driven Rescheduling Policies In Dynamic Shops”, *International Journal of Computer Integrated Manufacturing*, Vol. 5, 153–163, 1992.
- [6] Corman F., D’Ariano A., Pranzo M., Hansen, I.A.. Effectiveness of dynamic reordering and rerouting of trains in a complicated and densely occupied station area. *Transportation Planning and Technology* 34(4), 341-362, 2011
- [7] Corman F., D’Ariano A., Pacciarelli D., Pranzo M., “A tabu search algorithm for rerouting trains during rail operations”, *Transportation Research Part B*, Vol. 44 (1),. 175-192, 2010.
- [8] Cowling P., Johansson M., “Using Real Time Information For Effective Dynamic Scheduling”, *European Journal of Operational Research*, 139, 230–244, 2002.
- [9] D’Ariano A., *Improving Real-Time Train Dispatching: Models, Algorithms And Applications*, PhD Thesis, Delft University of Technology, the Netherlands, 2008.

- [10] D'Ariano A., Pacciarelli D., Pranzo M., "A Branch And Bound Algorithm For Scheduling Trains In A Railway Network", *European Journal of Operational Research*, Vol. 183(2), 643–657, 2007.
- [11] Dorfman M.J., Medanic J., "Scheduling Trains On A Railway Network Using A Discrete Event Model Of A Railway Traffic", *Transportation Research Part B*, Vol. 38, 81-98, 2004.
- [12] Hansen I.A., Pachl J., *Railway Timetable and Traffic*, Eurailpress, Hamburg, , 2008.
- [13] Higgins, A., Kozan, E., Ferreira, L., "Optimal scheduling of trains on a single line track", *Transportation Research Part B*, Vol. 30 (2), 147–161, 1996.
- [14] Kimms A., "Stability Measures For Rolling Schedules With Applications To Capacity Expansion Planning Master Production Scheduling And Lot Sizing", *International Journal of Management Sciences*, Vol. 26 (3), 355–366, 1998.
- [15] Lee T.S., Ghosh S., "Stability Of RYNSORD - A Decentralized Algorithm For Railway Networks Under Perturbations", *IEEE Transactions on Vehicular Technology*, Vol. 50 (1), 287-301, 2001.
- [16] Leon V.J., Wu S.D., Storer R.H., "Robustness Measures And Robust Scheduling For Job Shops", *IIE Transactions*, Vol. 26 (5), 32–43, 1994.
- [17] Lüthi M. , *Improving the efficiency of heavily used railway networks through integrated real-time rescheduling*, PhD thesis, ETH Zurich, 2009.
- [18] Mazzarello M., Ottaviani E., "A Traffic Management System For Real-Time Traffic Optimisation In Railways", *Transportation Research Part B*, Vol. 41(2), 246-274, 2007.
- [19] Mehta S.V., Uzsoy R.M., "Predictable Scheduling Of A Job Shop Subject To Breakdowns", *IEEE Transactions on Robotics and Automation*, Vol. 14, 365–378, 1998.
- [20] Meng L., Zhou X., "Robust Single-Track Train Dispatching Model Under A Dynamic And Stochastic Environment: A Scenario-Based Rolling Horizon Solution Approach", *Transportation Research Part B*, Vol. 45, 1080-1102, 2011.
- [21] ON-TIME project, <http://www.ontime-project.eu>, last accessed 03.01.2013.
- [22] Perello J., Gunreben S., Spadaro S., "A Quantitative Evaluation Of Reordering In OBS Networks And Its Impact On TCP Performance", *Proceeding of the IEEE International Conference on Optical Network Design and Modelling*, 12-14 March 2008, Spain.
- [23] Quaglietta E., *A Microscopic Simulation Model For Supporting The Design Of Railway Systems: Development And Applications*, PhD thesis, University of Naples Federico II, Naples, 2011.
- [24] Sabuncuoglu I., Kizilisik O.M., "Reactive Scheduling In A Dynamic And Stochastic FMS Environment", *International Journal of Production Research*, Vol. 41 (17), 4211–4231, 2003.
- [25] Theeg G., Vlasenko S., *Railway Signalling and Interlocking: International Compendium*, Eurailpress, Hamburg, 2009.
- [26] Törnquist J., Persson J., "N-Tracked Railway Traffic Re-Scheduling During Disturbances", *Transportation Research Part B*, Vol. 41, 342-362, 2007.
- [27] Törnquist J., "Railway Traffic Disturbance Management-An Experimental Analysis Of Disturbance Complexity, Management Objectives And Limitations In Planning Horizon", *Transportation Research Part A*, Vol. 41, 249-266, 2007.
- [28] Vieira G.E., Herrmann J.W., Lin E., "Predicting The Performance Of Rescheduling Strategies For Parallel Machines Systems", *Journal of Manufacturing Systems*, Vol. 19, 256–266, 2000.
- [29] Wu S.D., Storer R.H., Chang P.C., "One-Machine Rescheduling Heuristics With Efficiency And Stability As Criteria", *Computers and Operations Research*, Vol. 20, 1–14, 1993.