Coordinated railway traffic rescheduling with the Resource Conflict Graph model

Working Paper

Author(s):
Toletti, Ambra; Laumanns, Marco; Weidmann, Ulrich

Coordinated railway traffic rescheduling with the Resource Conflict Graph model

Ambra Toletti, Marco Laumanns and Ulrich Weidmann

April 7, 2019

Coordinated railway traffic rescheduling with the Resource Conflict Graph model

Ambra Toletti [a,b,1], Marco Laumanns [c], Ulrich Weidmann [a]

[a] ETH Zurich, Institute for Transport Planning and Systems
Stefano Franscini Platz 5, CH-8093 Zurich, Switzerland [b,] SBB Informatics
Haslerstrasse 30, CH-3000 Bern 65, Switzerland
[1] E-mail: ambra.toletti@sbb.ch, Phone: +41 (0)79 389 95 33
[c] Bestmile SA
Av. de Rhodanie 58, CH-1007 Lausanne, Switzerland

Abstract

The train rescheduling problem is quite a popular topic in the railway research community. Many approaches are available to reschedule traffic in a network partition, but very few works address the coordination of these partitions. In railway systems with very dense traffic, e.g. the Swiss one, it is not always possible to partition the network such that local rescheduling algorithms can work completely independently one from another. This paper proposes a coordination approach for adjacent local rescheduling algorithms. These algorithms are based on the Resource Conflict Graph model, which enables the representation of the interlocking system at a very fine granularity. Simulations on data from the Swiss Federal Railways show the validity of this approach to improve the consistency of decisions at the common boundaries of adjacent local rescheduling algorithms.

1 Introduction

A detailed schedule and a prompt response to deviations are key elements for high quality railway services. Deviations are usually addressed by dispatchers' and traffic controllers' manual interventions. To deliver high quality services in areas with extremely dense traffic, the Swiss Federal Railways (SBB) are increasingly interested in algorithmic support for real-time rescheduling. SBB aims at increasing the number of train runs of about 30% by 2040. A small part of this additional capacity will be obtained through infrastructure expansions, the largest part through denser schedules. Consequently, dense traffic will no longer be confined to a few disjoint areas where automated systems can reschedule traffic independently one from another. At the same time, it is insensible to imagine that a unique centralized system could reschedule the entire Swiss railway network in real-time. Thus, problem decomposition and coordination frameworks become an impending necessity.

Although many researchers and practitioners have addressed the local rescheduling problem (see, e.g., the literature review by Cacchiani et al., 2014), few have ventured decomposition and coordination topics. This paper tries to fill this gap with:

1. a decomposition scheme for the Resource Conflict Graph model,

2. a coordination framework for improving the consistency of decisions between adjacent zones.

The innovativeness of the proposed methodology can be summarised by the following elements:

- The decomposition scheme considers the network topology, the amount of traffic, the compute server available and the operational rules usually applied by railway companies.

- The coordination framework enables high scalability but keeps intact the ability of the local algorithms to find optimal local schedules that consider complex interactions among trains such as track allocation conflicts and connections.

- The approach is developed specifically for the Resource Conflict Graph model, which showed very promising results for local rescheduling tasks but has often been rejected in favour of continuous time formulations.

Section 2 presents the more relevant literature about decomposition and coordination of railway traffic rescheduling. Section 3 presents the decomposition scheme, the coordination framework, and the model for local rescheduling. In Section 4, the methodology is evaluated via simulations with data from the Swiss railway network.

2   A very brief literature review

Many researchers and practitioners recognised the impracticability of solving large centralised railway timetabling and rescheduling problems and proposed to decompose them into smaller instances that are easier to solve. In many cases, the decomposition is geographical, but there are several exceptions. Herrigel et al. (2013), for instance, propose a hierarchical decomposition approach for timetabling in which the different classes of services are added to the schedule stepwise starting with the ones with higher priority.

Salido et al. (2007) model the train timetabling problem as a Constraint Satisfaction and Optimisation problem. They show that the distributed problem can be solved much faster than the centralised. They propose two strategies for partitioning: The first groups the variables depending on the service type; the second groups them geographically. To coordinate the different partitions, partial states are passed between the agents solving the sub-problems until a compatible assignment is reached. The two strategies result in similar computation times.

Corman et al. (2010) emphasise the need for considering not only the geography but also the amount of traffic to get subproblems of tractable size, but they do not outline any precise scheme to do this. Caimi (2009) formalises the network decomposition originally proposed by Laube et al. (2007). Again, the decomposition depends on geography and traffic density (cfr. Salido et al., 2007; Corman et al., 2010). *Condensation zones* are the bottlenecks of the network. *Compensation zones* coincide with regions with lower traffic density, where time reserves for delay absorption can be planned. It is assumed that each two condensation zones are separated by a condensation zone. This decomposition is adopted by Fuchsberger (2012) for a Model Predictive Control to reschedule condensation zones. Note that in this case the areas at the microscopic level are disjoint. The same holds for

Dollevoet et al. (2014), who iterate between delay management and local rescheduling in stations for network-wide control.

Fischer and Schlechte (2017) apply Lagrangian relaxation to the time-expanded formulation of the acyclic timetabling problem. They obtain subproblems that model avoidance of independent conflicts involving a small set of resources and few trains. Consistency of the global solution is achieved with a proximal bundle method.

Jia and Zhang (1994) propose a hierarchically distributed fuzzy decision-making system that mimics the decisions of human dispatchers via IF-THEN rules. An upper level takes the strategic decisions, while the operational decisions are taken by local controllers. Narayanaswami and Rangaraj (2015) propose to reschedule railway traffic on a single-track line using Multi-Agent Systems. A supervisor agent detects conflicts and informs an auctioneer. The auctioneer asks the affected train agents to place their bids and declares the winner. The look-ahead ranking bid ensures deadlock-avoidance. Then, supervisors determine the departure times of the conflicting trains and reschedule the rest of the services using a MILP. Finally, station agents allocate the required resources and check the feasibility of the solution.

3 Methodology

In this paper, the train rescheduling problem is represented via a Resource Conflict Graph model (RCG, see, e.g., Caimi, 2009). The model relies on discrete variables, each representing a blocking time stairway (i.e. a secured distance-time-path). Each blocking time stairway is linked with a route-time-speed alternative for a train movement between two operation points (e.g., stations, junctions, stops). Conflicting track allocations are prevented by constraints based on the cliques of the conflict graphs of the interlocking resources.

The fine granularity of the topology representation of this model enables the definition of new schedules that are conflict-free at track-circuit level but prevents its straightforward application to large networks and long horizons. To overcome the scalability problem the network is partitioned geographically into several local rescheduling areas. The RCG models of the local rescheduling areas are almost independent from each other, as they are only linked together by Lagrangian multipliers applied to the constraints requiring that adjacent areas make consistent speed and time decisions at the common boundaries.

The coordination of the areas is achieved when commonly agreed values for the Lagrangian multipliers are found. A standard approach to find these values is the subgradient method. In this paper, this standard approach is compared with an ad-hoc developed strategy. The strategy is based on the assumption that the decisions of adjacent areas are probably quite close to each others, given that all the local rescheduling problems aim at minimizing the deviation from a common global schedule (i.e. the yearly timetable). Thus, after each iteration, a trade-off schedule with the times and speeds at the boundaries is computed from the solutions of the adjacent areas. Then, each neighbour uses the trade-off schedule as part of its objective in the next iteration.

Section 3.1 presents the RCG model for each local rescheduling problem. Section 3.2 describes the decomposition and the coordination approaches.

3.1 Resource conflict graph model for local rescheduling (RCG)

Fuchsberger (2012) and Caimi et al. (2012) successfully applied the RCG model for predic-

tive control in a condensation area around a large station with highly complex topology. The current work is based on the enhanced formulation used in Toletti et al. (2017), where the blocking time stairways are cut at operation points and are initiated from a database created using a simulation-based approach. This simulation-based blocking time stairway patterns generation makes sure that the train dynamics and the topology utilization of the runs are taken into account by the final schedule, as the model does not represent them explicitly.

The RCG model for railway traffic rescheduling is an integer linear program in which the (binary) decision variables correspond to decisions about deferring train runs, cancellations, passenger transfers and blocking time stairways between operation points. Each blocking time stairway is built around a train trajectory, that is the time-space-speed representation of a train run. Thus, a train trajectory defines the departure time and track from an operation point and the arrival time and track at the successive operation point, as well as the routes and speed profiles in-between.

Given a geographical area, let $\mathcal{S}$ be the set of its operation points and $\mathcal{T}$ the set of trains travelling in it. The set of tracks in operation point $s \in \mathcal{S}$ is denoted by P_s, and $P_{s,t}$ is the set of tracks that a specific train $t \in \mathcal{T}$ can use. It is assumed that each train runs through a predefined sequence $(s_0, s_1, \ldots, s_{n_t})$ of operation points, where s_0 is either the departure station, depot or portal from which the train enters the control area and s_{n_t} either the final station, depot or portal from which it leaves the control area. Let $\mathcal{B}_{t,i}$ be a set of alternative blocking time stairways for the run of train t from s_{i-1} to s_i.

In this framework, a *resource* is a subset of infrastructure elements that are locked and released simultaneously according to the *Blocking Time Theory* (see, e.g., Hansen and Pachl, 2014). $\mathcal{R}$ denotes the set of resources in the control area and $\mathcal{B}^r$ the set of all blocking time stairways using a resource $r \in \mathcal{R}$. $\underline{b(r)}$ and $\overline{b(r)}$ denote the lock and release times of resource r by the blocking time stairway $b \in \mathcal{B}^r$. For each resource $r \in \mathcal{R}$, the nodes of the *conflict graph* correspond to the blocking time stairways $\mathcal{B}^r$ and the edges link together the nodes of blocking time stairways overlapping on resource r.

Let mt_i^t be the minimum dwell time of train t in operation point s_i. Blocking time stairways that are not separated by this minimum time are said to be *incompatible* (nomenclature by Fuchsberger, 2012). The sets of maximal incompatible sets are denoted by Ω_i^t. Similarly, let mv_i^t be the maximum difference tolerable between the arrival speed of train t at operation point s_i and its departure. Blocking time stairways whose speed difference is larger than the tolerated one are said *speed-incompatible* and the set of maximal speed-incompatible sets is denoted by Ωv_i^t.

The binary decision variables of the problem indicate whether runs between two consecutive stations are scheduled in the current horizon, whether they are cancelled, whether a trajectory corresponding to a blocking time stairway is selected, and whether scheduled connections are kept. In this paper we neglect passenger connections, thus the following decision variables are considered:

$$r_{t,i} = \begin{cases} 1, & \text{if the run of train } t \text{ from } s_{i-1} \text{ to } s_i \text{ is scheduled} \\ & \text{in the current horizon} \\ 0, & \text{else} \end{cases} \tag{1}$$

$$h_{t,i} = \begin{cases} 1, & \text{if the run of train } t \text{ from } s_{i-1} \text{ to } s_i \text{ is cancelled} \\ 0, & \text{else} \end{cases} \tag{2}$$

$$x_b = \begin{cases} 1, & \text{if the schedule uses } b \\ 0, & \text{else} \end{cases} \tag{3}$$

Since freight and service trains cannot be cancelled, h. variables are defined for passenger trains only. If a train run is neither scheduled in the current horizon nor cancelled, then it is deferred to the next horizon.

Linear constraints model the functional requirements of railway operations, time-space-speed consistency, absence of conflicts, and feasibility of passenger transfers, as follows:

$$r_{t,i} \leq r_{t,i-1} \qquad \forall t \in \mathcal{T}, i = 2, \ldots, n_t \tag{4}$$

$$r_{t,i} + h_{t,i} \leq 1 \qquad \forall t \in \mathcal{T}_p, i = 1, \ldots, n_t \tag{5}$$

$$h_{t,i} \geq h_{t,i-1} \qquad \forall t \in \mathcal{T}_p, i = 2, \ldots, n_t \tag{6}$$

$$r_{t,i} = \sum_{b \in \mathcal{B}_{t,i}} x_b \qquad \forall t \in \mathcal{T}, i = 2, \ldots, n_t \tag{7}$$

$$\sum_{b \in C} x_b \leq 1 \qquad \forall C \in \mathcal{C}^r, r \in \mathcal{R} \setminus \bigcup_{s \in \mathcal{S}} P_s \tag{8}$$

$$0 \leq \sum_{\substack{b \in A_{.,p} \\ \alpha(b) \leq \alpha}} x_b - \sum_{\substack{b \in D_{.,p} \\ \delta(b) < \alpha}} x_b \leq 1 \qquad \begin{array}{l} \forall \alpha \in \{\alpha(b), b \in A_{.,p}\}, p \in P_s, \\ s \in \mathcal{S} \end{array} \tag{9}$$

$$\sum_{\substack{b \in A_{.,p} \\ \alpha(b) = \alpha}} x_b \leq 1 \qquad \begin{array}{l} \forall \alpha \in \{\alpha(b), b \in A_{t,p}\}, p \in P_s, \\ s \in \mathcal{S} \end{array} \tag{10}$$

$$\sum_{b \in A_{t,p}} x_b - \sum_{b \in D_{t,p}} x_b \leq 0 \qquad \forall p \in P_{.,t}, t \in \mathcal{T} \tag{11}$$

$$\sum_{b \in U} x_b + \sum_{b \in V} x_b \leq 1 \qquad \forall (U, V) \in \Omega_i^t, t \in \mathcal{T}, i = 2, \ldots, n_t - 1 \tag{12}$$

$$\sum_{b \in U} x_b + \sum_{b \in V} x_b \leq 1 \qquad \forall (U, V) \in \Omega v_i^t, t \in \mathcal{T}, i = 2, \ldots, n_t - 1 \tag{13}$$

Constraints (4) ensure that at most one blocking time stairway is allocated to each successive section and none is allocated to trains that have not been scheduled at a previous point. Constraints (5) link $r_{t,i}$ to $h_{t,i}$ variables, where $\mathcal{T}_p$ denotes the set of passenger trains. Cancellation consistency is ensured by constraints (6). Constraints (7) link $r_{t,i}$ to x_b variables. Conflicts are avoided thanks to constraints (8-10): (9-10) prevent conflicts at operation points, while (8) are used for the other resources, where $\mathcal{C}^r$ denotes the set of maximal cliques of the conflict graph for resource r. Constraints (11) force trains to depart from operation points from the same track they have arrived to. The *self-connection* constraints (12) ensure that trains only depart after they have arrived to the operation point and the minimum dwelling time has expired and that the arrival and departure times coincide (up to a given tolerance value) for trains passing an operation point without stopping. Similarly, Constraints (13) ensure speed consistency.

The objective function can model train delays, customer inconvenience (delays, waiting times, connections missed, cfr. Caimi et al., 2012; Toletti and Weidmann, 2016) and/or energy targets (Toletti et al., 2015). Hereinafter, we aim at minimising the sum of arrival delays at scheduled stops or at the boundaries of the area, which can be computed in the

following way

$$f(x,r,h) = \sum_{t \in \mathcal{T}} \sum_{\substack{i=1,\ldots,n_t \mid \\ \exists \text{scheduled} \\ \text{event at } s_i}} (1 - r_{t,i} - h_{t,i}) \cdot \Delta$$

$$+ \sum_{t \in \mathcal{T}} \sum_{\substack{i=1,\ldots,n_t \mid \\ \exists \text{scheduled} \\ \text{event at } s_i}} \sum_{b \in \mathcal{B}_{t,i}} x_b \left(\alpha(b) - \hat{\alpha}_{t,i} \right) \qquad (14)$$

where Δ denotes the size of the rescheduling horizon, $\alpha(b)$ is the arrival time of blocking time stairway b and $\hat{\alpha}_{t,i}$ the scheduled arrival time of train t at s_i, while $\overline{\mathcal{H}}$ is the rescheduling horizon end. A fixed penalty equals the length of the rescheduling horizon is assigned to runs that are scheduled outside the rolling horizon (first double-sum of 14). The last row of (14) quantifies the delay of the chosen blocking time stairways.

Table 1: Notation of the resource conflict graph model

Input sets	
$\mathcal{R}$	(infrastructure) resources in control area
$\mathcal{S}$	operation points (stations, stops, junctions, portals) in control area
P_s	tracks at operation point $s \in \mathcal{S}$
$\mathcal{T}$	trains running in the control area during the considered horizon
$\mathcal{T}_p \subseteq \mathcal{T}$	passenger trains running in the control area during the considered horizon
$\mathcal{B}_{t,i}$	blocking time stairway candidates (combinations of routing, timing and speed control) for the run of train $t \in \mathcal{T}$ in section $i = 1, \ldots, n_t$
Input (timetable) parameters	
$(s_0, s_1, \ldots, s_{n_t}) \in \mathcal{S}^{n_t+1}$	ordered operation points on the itinerary of train $t \in \mathcal{T}$
$\hat{\alpha}_{t,i}$	scheduled arrival time of train $t \in \mathcal{T}$ at i^{th} operation point on its itinerary
$\hat{\delta}_{t,i}$	scheduled departure time of train $t \in \mathcal{T}$ from i^{th} operation point on its itinerary
Decision variables	
$r_{t,i}$	decision of including i^{th} run of train $t \in \mathcal{T}$ in rescheduling solution of considered rescheduling horizon
$h_{t,i}$	decision of cancelling i^{th} run of train $t \in \mathcal{T}$
x_b	decision about including blocking time stairway b in rescheduling solution
Model sets	
$\mathcal{B}^r \subseteq \bigcup_{t \in \mathcal{T}} \bigcup_{i=1}^{n_t} \mathcal{B}_{t,i}$	blocking time stairways using resource $r \in \mathcal{R}$
$\mathcal{C}^r$	maximal conflict cliques for resource $r \in \mathcal{R}$
$A_{t,p}$	blocking time stairway candidates for train $t \in \mathcal{T}$ arriving at operation point track $p \in \bigcup_{s \in \mathcal{S}} P_s$
	Continued on next page

Table 1 – continued from previous page

$D_{t,p}$	blocking time stairway candidates for train $t \in \mathcal{T}$ departing from operation point track $p \in \bigcup_{s \in \mathcal{S}} P_s$
$A_{\cdot,p} = \bigcup_{t \in \mathcal{T}} A_{t,p}$	blocking time stairways arriving at operation point track $p \in \bigcup_{s \in \mathcal{S}} P_s$
$D_{\cdot,p} = \bigcup_{t \in \mathcal{T}} A_{t,p}$	blocking time stairways departing from operation point track $p \in \bigcup_{s \in \mathcal{S}} P_s$
Ω_i^t	maximal incompatible sets for arrival and departure events of train $t \in \mathcal{T}$ at operation point $s_i \in \mathcal{S}$
Functions	
$\alpha(b)$	arrival time of blocking time stairway b
$\delta(b)$	departure time of blocking time stairway b
Parameters of the objective function	
Δ	size of rescheduling horizon
$\overline{\mathcal{H}}$	end of rescheduling horizon

3.2 Non-hierarchical coordination framework

This section proposes a non-hierarchical framework for distributed railway traffic rescheduling with the RCG model (1)-(13). Assuming that the size of the rescheduling horizon and the number of trains to be considered by a RCG model are inherited by the yearly timetable, which is fixed, it is sensible to partition the rescheduling problem geographically. The following definitions of *network partitioning* and *problem partitioning* are used.

Definition 3.1 A network partitioning *of a network* $\mathcal{N}$ *consists of a set* $\mathcal{N}_0, \mathcal{N}_1, \ldots, \mathcal{N}_m$ *of connected sub-networks or* zones *of* $\mathcal{N}$ *such that*

(R1) the zones cover the entire network, i.e., $\bigcup_{i=0}^{m} \mathcal{N}_i = \mathcal{N}$,

(R2) the boundaries $\partial\mathcal{N}_i$ *of the each zone* $\mathcal{N}_i$ *are operation points, i.e.,* $\partial\mathcal{N}_i \subset \mathcal{S}$, $\forall 0 \leq i \leq m$, *where* $\mathcal{S}$ *denotes the set of operation points in* $\mathcal{N}$.

The set $S_{i,j} := \{s \in \mathcal{S} | s \in \mathcal{N}_i \cap \mathcal{N}_j\}$ *denotes the* portals *between zones* $\mathcal{N}_i$ *and* $\mathcal{N}_j$, *i.e., the operation points at the common boundary.* $\mathcal{S}_{portals} := \bigcup_{i=0}^{m} \bigcup_{j>i} S_{ij}$ *indicates the set of all portals. Each operation point* $s \in \mathcal{S} \setminus \mathcal{S}_{portals}$ *is in exactly one sub-network* $\mathcal{N}_{\cdot}$. *Zones sharing a boundary, i.e.* $S_{ij} \neq \emptyset$, *are said* adjacent *or* neighbouring.

Definition 3.2 *Let a network partitioning* $\mathcal{N}_0, \mathcal{N}_1, \ldots, \mathcal{N}_m$ *of* $\mathcal{N}$ *be given. And let* $\Phi = \min_{x \in X} f(x)$ *be the mathematical representation of the rescheduling problem in network* $\mathcal{N}$, *where* x *is a vector of decision variables,* X *is the set of decisions satisfying the constraints of railway traffic management, and* f *represents the objectives of railway traffic rescheduling (e.g., minimise train delays). A* problem partitioning *of* Φ *is a set of local rescheduling problems* $\phi^0, \phi^1, \ldots, \phi^m$ *in sub-networks* $\mathcal{N}_0, \mathcal{N}_1, \ldots, \mathcal{N}_m$ *that can be solved in a distributed way and such that combining their solutions one obtains a feasible global schedule.*

Using the notation from Table 1, the problem of rescheduling traffic in a network $\mathcal{N}$ with a set of operation points $\mathcal{S}$ is denoted as

$$\Phi = \min_{(x,r,h)\in X} f(x,r,h) \tag{15}$$

where X denotes the set of values of (x,r,h) satisfying constraints (1)-(13).

The central topics of this section are the partitioning of (15) into a set of local rescheduling problems $\{\phi^i\}_i$ satisfying Definition 3.2 and the coordination of the local rescheduling problems to get a feasible global schedule. In fact, when local rescheduling of adjacent areas is executed separately, consistent track, time, and speed assignments at portals are not guaranteed. Even if a common global timetable is used as target by all zones, explicit coordination is often required (see, e.g., the results in Section 4). Coordination can be achieved in different ways: in non-hierarchical approaches adjacent areas exchange values at boundaries directly with each other to ensure coordination; while in multilevel approaches an upper level enforces coordination at portals by fixing common time, speed and track values. Here a non-hierarchical approach is proposed.

Note that each train run from an operation point to the next one can be mapped to exactly one sub-network $\mathcal{N}_i$ of a network partitioning defined according to Definition 3.1. Then, the railway traffic rescheduling problem (15) can be reformulated as

$$\Phi = \min_{(x,r,h)\in X} \sum_{i=0}^{m} f^i(x^i,r^i,h^i) \tag{16}$$

where x^i, r^i, h^i denote the decision variables linked to each sub-network $\mathcal{N}_i$, and f^i denotes the part of the objective function that depends exclusively on them.

The RCG model includes several constraints that ensure consistency and conflict freedom at every operation point, including portals. Hereafter, these consistency constraints at portals are referred to as *complicating constraints*. The idea of the problem partitioning scheme proposed in this paper is to apply a Lagrangian relaxation to the complicating constraints and get

$$\min_{(x,r,h)\in X'} \sum_{i=0}^{m} f^i(x^i,r^i,h^i)$$
$$+ \sum_{s\in\mathcal{S}_{portals}} (\lambda_r^s A_r^s r + \lambda_h^s A_h^s h + \lambda_x^s [A_x^s x - 1]) \tag{17}$$
$$\lambda \geq 0$$

where X' denotes the set defined by (1)-(13) without the complicating constraints, A_r^s denotes the coefficients of variables r in constraints of type (4) related to portal $s \in \mathcal{S}_{portal}$, A_h^s denotes the coefficients of variables h in constraints of type (6) related to portal s, A_x^s denotes the coefficients of variables x in conflict constraints of types (9)-(10) and in consistency constraints of types (11)-(13) related to portal s, and λ is a vector of Lagrangian multipliers.

Given the linearity of the complicating constraints, the terms of (17) can be rearranged. In addition, X' is the product of $(m+1)$ disjoint sets $X_0, X_1, \ldots, X_m$ that represent the local RCG models (1)-(13) in the sub-networks $\mathcal{N}_0, \mathcal{N}_1, \ldots, \mathcal{N}_m$. Then, the rescheduling problem (16) is equivalent to finding a solution to

$$\Phi = \max_{\lambda} \sum_{i=0}^{m} \phi^i(\lambda) \tag{18}$$

Figure 1: Iterative approach for non-hierarchical coordinated rescheduling of adjacent zones.
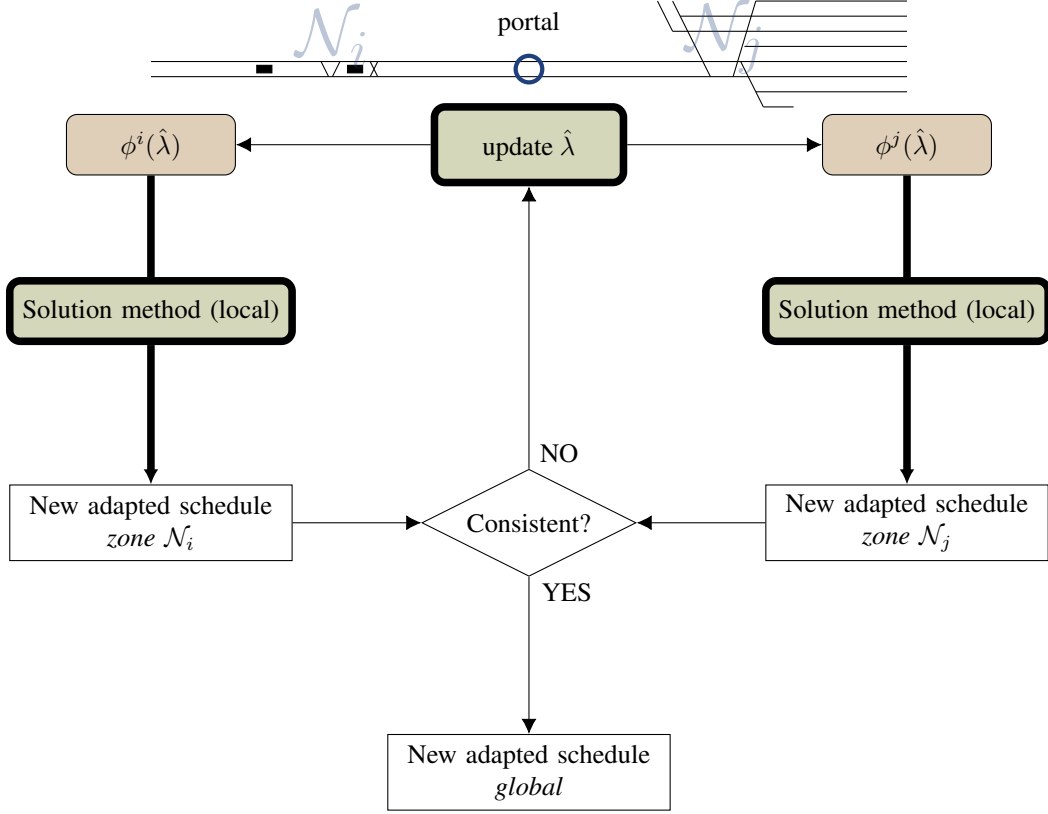
where $\phi^i(\lambda)$ denotes the local rescheduling problem in zone $\mathcal{N}_i$ defined as

$$\phi^i(\lambda) = \min_{(x^i,r^i,h^i)\in X_i} \left\{ f^i \left(x^i, r^i, h^i \right) \right.$$
$$+ \sum_{s\in\partial\mathcal{N}_i} \left(\lambda_r^s A_r^{si} r^i + \lambda_h^s + A_h^{si} h^i \right. \tag{19}$$
$$\left. \left. + \lambda_x^s \left[A_x^{si} x^i - 1 \right] \right) \right\}$$

where A_r^{si}, A_h^{si}, A_x^{si} denote the columns of the coefficient matrices A_r^s, A_h^s, A_x^s corresponding to the decisions in zone $\mathcal{N}_i$.

Lagrangian dual problems such as (18) are usually solved using an iterative approach that starts with some initial estimates of the Lagrangian multipliers $\lambda(0)$ and updates them at each iteration with the current solutions to local problems $\{\phi^i\}_i$. Figure 1 shows the coordination loop for distributed non-hierarchical railway traffic rescheduling. Each rescheduling zone is modelled via (19) and solved separately from the others. All new adapted local schedules are put together and consistency at the portals is checked. If the obtained global schedule is consistent, it is returned. Otherwise, the estimates of the Lagrangian multipliers
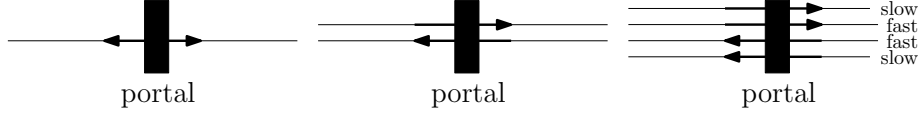
Figure 2: Tracks assignments at portals

are updated and injected into the local models to be solved again.

Note that, if large densely utilised stations are used as portals, the number of complicating constraints, i.e., involving variables from different zones, is huge. Consequently, a huge number of dual variables (λ) is needed, which results in a large number of operations to update the coefficients of the local rescheduling problems $\{\phi^i\}_i$ and to compute the new dual estimates after each iteration. In addition, looking more closely at the reference SBB network one discovers that a number of operation points represent junctions and switch regions where it is not sensible to stop. This observation motivates the following definitions.

Definition 3.3 *A* no-wait portal *is an operation point between zones where*

- *no train is planned to stop and*

- *the tracks are so short that if a conflict between two train movements occur, the blocking time intervals of at least one commonly used line resource adjacent to the track overlap.*

Definition 3.4 *A* valid no-wait network partitioning *of a network $\mathcal{N}$ consists of a set $\mathcal{N}_0$, $\mathcal{N}_1$, ..., $\mathcal{N}_m$ of zones of $\mathcal{N}$ that satisfies the requirements (R1-R2) of Definition 3.1 and*

(R3) the boundaries of the zones are no wait portals, i.e., $\partial \mathcal{N}_i \subset \{s \in \mathcal{S} | s \text{ no-wait portal}\}$, $\forall 0 \leq i \leq m$.

Consider problem (18) and let $\mathcal{N}_0, \mathcal{N}_1, \ldots, \mathcal{N}_m$ be a valid no-wait network partitioning. In addition, consider the fact that there are often operational rules that fix the direction of travel of the tracks at portals (e.g. see the examples in Figure 2): On single lines there is only one possible track assignment for all movements; In multiple-track sections the maximum capacity is attained when all tracks are used for one direction (and train type). With these assumptions the conflict-freedom constraints (9)-(10) and track consistency constraints (11) linked to the portals of the valid no-wait network partitioning can be removed from the definition of the domain X, while the other complicating constraints for all portals $s \in \mathcal{S}_{portals}$ become

$$(4) \quad \rightarrow \quad r_{t,\rightarrow s} = r_{t,s\rightarrow} \quad \forall t \in \mathcal{T} \tag{20}$$

$$(6) \quad \rightarrow \quad h_{t,\rightarrow s} = h_{t,s\rightarrow} \quad \forall t \in \mathcal{T} \tag{21}$$

$$(12) \quad \rightarrow \quad \sum_{b \in A_{t,p}} \alpha(b) x_b = \sum_{b \in D_{t,p}} \delta(b) x_b \quad \forall t \in \mathcal{T} \tag{22}$$

$$(13) \quad \rightarrow \quad \sum_{b \in A_{t,p}} v\alpha(b) x_b = \sum_{b \in D_{t,p}} v\delta(b) x_b \quad \forall t \in \mathcal{T} \tag{23}$$

where $p \in P_s$ denotes the track prescribed by the mentioned operational rules for each train, $\rightarrow s$ indicates the section of a train itinerary with arrival at portal s, and $s \rightarrow$ the section departing from it.

Let $\mathcal{T}^i_{\to s}$ denote the set of trains in zone $\mathcal{N}_i$ leaving from portal $s \in \partial\mathcal{N}_i$ and $\mathcal{T}^i_{s\to}$ the ones entering it. In addition, given that each train is supposed to pass only once through each portal, one can uniquely define the variables of one zone linked to the runs from and to a portal s using the index s instead of the run index. The application of a Lagrangian relaxation to the complicating constraints (20)-(23) gives

$$
\begin{aligned}
\Phi = \min_{(x,r,h)\in X'} \sum_{i=0}^m \Bigg\{ & f^i(x^i, r^i, h^i) \\
& + \sum_{j\neq i} \sum_{s\in\mathcal{N}_i\cap\mathcal{N}_j} \Bigg[\sum_{t\in\mathcal{T}^i_{\to s}\cup\mathcal{T}^i_{s\to}} \left\{ \lambda^{s,t}_{r,ij} \left[r^i_{t,s} - r^j_{t,s} \right] + \lambda^{s,t}_{h,ij} \left[h^i_{t,s} - h^j_{t,s} \right] \right\} \\
& \quad + \sum_{t\in\mathcal{T}^i_{\to s}} \left(\lambda^{s,t}_{x1,ij} \left[\sum_{b\in A_{t,p}} \alpha(b)x^i_b - \sum_{b\in D_{t,p}} \delta(b)x^j_b \right] \right. \\
& \qquad\qquad \left. + \lambda^{s,t}_{x2,ij} \left[\sum_{b\in A_{t,p}} v\alpha(b)x^i_b - \sum_{b\in D_{t,p}} v\delta(b)x^j_b \right] \right) \\
& \quad + \sum_{t\in\mathcal{T}^i_{s\to}} \left(\lambda^{s,t}_{x1,ij} \left[\sum_{b\in D_{t,p}} \delta(b)x^i_b - \sum_{b\in A_{t,p}} \alpha(b)x^j_b \right] \right. \\
& \qquad\qquad \left. + \lambda^{s,t}_{x2,ij} \left[\sum_{b\in D_{t,p}} v\delta(b)x^i_b - \sum_{b\in A_{t,p}} v\alpha(b)x^j_b \right] \right) \Bigg] \Bigg\} \\
& \lambda \geq 0
\end{aligned}
\tag{24}
$$

where $\lambda^{s,t}_{r,ij}$, $\lambda^{s,t}_{h,ij}$, $\lambda^{s,t}_{x1,ij}$, $\lambda^{s,t}_{x2,ij}$ denote the multipliers for complicating constraints (20), (21), (22), and (23). The next two subsections describe how to update the Lagrangian multipliers in each iteration.

3.3 Subgradient method no-wait formulation

Lagrangian dual problems such as (24) are usually solved using the subgradient method (see, e.g., Terelius et al., 2011). This method iteratively updates the λ estimates according to

$$
\lambda(k+1) = \lambda(k) + w \cdot q(k)
\tag{25}
$$

where $\lambda(k)$ denotes the value of λ used at the kth iteration to solve the local problems $\{\phi^i\}_i$, $w(k)$ indicates the step size (see, e.g., Nedic and Ozdaglar, 2009), and $g(k)$ indicates a subgradient of the function $Q(\lambda) = \sum_{i=0}^m \phi^i(\lambda)$ at $\lambda(k)$. The subgradient method with constant step size is guaranteed to converge to some value very close to the optimum (see e.g, Boyd et al., 2003). With a constant step size w and an initial estimates of the duals

$\lambda(0) = 0$ each local rescheduling problem of (24) can be formulated as:

$$\phi^i(\lambda(K+1)) = \min_{(x^i, r^i, h^i) \in X_i} \left\{ f^i(x^i, r^i, h^i) \right. \tag{26}$$

$$+ \sum_{j \neq i} \sum_{s \in \mathcal{N}_i \cap \mathcal{N}_j} \left[\sum_{t \in \mathcal{T}^i_{\to s} \cup \mathcal{T}^i_{s \to}} \left\{ \tilde{\lambda}^{i,s}_{t,r}(K) r^i_{t,s} + \tilde{\lambda}^{i,s}_{t,h}(K) h^i_{t,s} \right\} \right.$$

$$+ \sum_{t \in \mathcal{T}^i_{\to s}} \sum_{b \in A_{t,p}} \tilde{\lambda}^{ij}_{s,t,p}(b,K) x^i_b$$

$$\left. \left. + \sum_{t \in \mathcal{T}^i_{s \to}} \sum_{b \in D_{t,p}} \tilde{\lambda}^{ij}_{s,t,p}(b,K) x^i_b \right] \right\}$$

$$\tilde{\lambda}^{i,s}_{t,r}(K) = 2w \sum_{k=0}^{K} \left[r^i_{s,t}(k)_- r^j_{s,t}(k) \right] \tag{27}$$

$$\tilde{\lambda}^{i,s}_{t,h}(K) = 2w \sum_{k=0}^{K} \left[h^i_{s,t}(k) - h^j_{s,t}(k) \right] \tag{28}$$

$$\tilde{\lambda}^{i,\to s}_{t,p}(b,K) = 2w \sum_{k=0}^{K} \left[\alpha(b) \left(\alpha^i_{s,t,p}(k) - \delta^j_{s,t,p}(k) \right) + v\alpha(b) \left(v\alpha^i_{s,t,p}(k) - v\delta^j_{s,t,p}(k) \right) \right] \tag{29}$$

$$\tilde{\lambda}^{i,s\to}_{t,p}(b,K) = 2w \sum_{k=1}^{K} \left[\delta(b) \left(\delta^i_{s,t,p}(k) - \alpha^i_{s,t,p}(k) \right) + v\delta(b) \left(v\delta^i_{s,t,p}(k) - v\alpha^j_{s,t,p}(k) \right) \right] \tag{30}$$

This can be interpreted as follows: in each iteration, the local rescheduling problem ϕ^i receives suggestions based on the results of the previous iteration on how to change the values of the variables at the portals to get a consistent global schedule. The signs of the differences indicate whether the value should grow or fall and their absolute values scaled according to $2w$ suggest the entity of the change needed.

3.4 Negotiation based no-wait formulation

This section presents an alternative to the subgradient method for updating the λ's. This method uses only the local schedules from the last iteration. For each portal an "intermediate schedule" is computed by combining the current local schedules with some weights that reflect the relative importance of exchanged information. Then, the values $\tilde{\lambda}$ are defined to quantify the distances of the decision alternatives from the intermediate schedule. The relative importance of exchanged information can depend on:

(a) direction: The most natural way to coordinate the movement of trains along different areas is by considering their travel directions. For road traffic, Xie et al. (2014) suggest that communicating outflows to neighbouring intersections within a rolling horizon provides an implicit coordination. Direction-based precedences are the base of the coordination for railway traffic rescheduling by Corman et al. (2012). The strength of this strategy is that earlier due decisions get higher priority than the following ones. A drawback of this strategy is that trains might be forced into an already congested area even if they could stay a little longer in a previous uncongested area without creating any inconvenience.

(b) hierarchy: Analogous to the microscopic timetabling work-flow with condensation and compensation areas proposed by Caimi (2009). The local schedules of condensation zones are imposed to the adjacent compensation zones. Of course this strategy can be applied to different hierarchies than condensation zones before compensation zones (e.g., model based Kersbergen et al., 2016). A drawback of this approach is that the rescheduling problem in a low-rank area can result infeasible due to the boundary conditions imposed by higher-ranked neighbours.

(c) negotiation: A combination of the previous two strategies. Adjacent zones negotiate the passing track, time and speed of each train at the common portals aiming at reaching a common result. The weight of each local schedule for computing the intermediate schedule can depend on the travel direction of the train or/and the type of area (condensation or compensation). Note that this strategy also allows the definition of overlapping areas.

Strategies a) and b) are special cases of strategy c), in which the weight of the schedule information of either one direction or area is one and the other zero. Table 2 exemplifies the three strategies for the situation depicted in Figure 1 and assuming zone $\mathcal{N}_i$ is a condensation zone and zone $\mathcal{N}_j$ a compensation zone. Strategy c) also enables the assignment of different weights to each of the components track, time and speed.

For given negotiation weights and assuming a fixed track allocation at portals, the approximate duals $\tilde{\lambda}$ can be updated as follows:

$$\tilde{\lambda}_{t,r}^{i,s}(k+1) = \varphi(t) \left[1_{\{w_{t,s}^{i,r} \geq w_{t,s}^{j,r} \wedge r_{s,t}^i(k)=0\}} + 1_{\{w_{t,s}^{i,r} < w_{t,s}^{j,r} \wedge r_{s,t}^j(k)=0\}} \right] \tag{31}$$

$$\tilde{\lambda}_{t,h}^{i,s}(k+1) = \varphi(t) \left[1_{\{w_{t,s}^{i,h} \geq w_{t,s}^{j,h} \wedge h_{s,t}^i(k)=0\}} + 1_{\{w_{t,s}^{i,h} < w_{t,s}^{j,h} \wedge h_{s,t}^j(k)=0\}} \right] \tag{32}$$

$$\tilde{\lambda}_{t,p}^{i,s}(b,k+1) = \varphi(t) \left[\left| \alpha(b) - \left(w_{t,s}^{i,time} \alpha_{s,t}^i(k) + w_{t,s}^{j,time} \delta_{s,t}^j(k) \right) \right| \right. \tag{33}$$
$$\left. + \left| v\alpha(b) - \left(w_{t,s}^{i,speed} v\alpha_{s,t}^i(k) + w_{t,s}^{j,speed} v\delta_{s,t}^j(k) \right) \right| \right]$$

$$\tilde{\lambda}_{t,p}^{i,s}(b,k+1) = \varphi(t) \left[\left| \delta(b) - \left(w_{t,s}^{i,time} \delta_{s,t}^i(k) + w_{t,s}^{j,time} \alpha_{s,t}^i(k) \right) \right| \right. \tag{34}$$
$$\left. + \left| v\delta(b) - \left(w_{t,s}^{i,speed} v\delta_{s,t}^i(k) + w_{t,s}^{j,speed} v\alpha_{s,t}^j(k) \right) \right| \right]$$

where $\varphi(t)$ is a function of the available time for decisions about train t; $w_{t,s}$ are the weights of the decisions about train t at portal s in the zones $\mathcal{N}_i$ and $\mathcal{N}_j$ concerning the components

Table 2: Weights of exchanged information for rescheduling in adjacent zones: Examples for Figure 1.

Travel direction / Approach	$\mathcal{N}_i$		$\mathcal{N}_j$	
	inbound	outwards	inbound	outwards
(a) direction	0	1	0	1
(b) hierarchical	1	1	0	0
(c) negotiation	[0, 1]	[0, 1]	[0, 1]	[0, 1]

time and speed, and of decisions on deferring it to the next horizon or cancelling it; $r_{s,t}(k)$ is the decision taken in the kth iteration about deferring the run of train t on the section adjacent to portal s; $h_{s,t}(k)$ is the decision selected in the kth iteration about cancelling the run of train t on the section adjacent to portal s; $\alpha_{s,t}(k)$ and $\delta_{s,t}(k)$ are the arrival and departure times of train t at the portal s found within the kth iteration; and $v\alpha_{s,t}(k)$ and $v\delta_{s,t}(k)$ the arrival and departure speeds.

As a train approaches a portal, agreeing on a time and a speed becomes more important, i.e., the coordination costs increase with approaching application time of the decision. Thus, φ is a monotonically decreasing function such as

$$\varphi(t) = \frac{M}{\min(\alpha_{t,i}, \alpha_{t,j}) - h_{begin}} \tag{35}$$

where h_{begin} is the beginning of the current rescheduling horizon and M is a large constant.

Differently from the subgradient method, there is no guarantee that this method converges. Thus, the following rule is applied. If no convergence has been reached after a given number of iterations, the values at the portals are fixed to the ones found in the last iteration according to the weights of the components. The number of iteration should be determined according to the available time for computation and the available compute server.

4 Experiments

The non-hierarchical coordination approach for RCG has been tested using simulations of the partition of the Swiss railway network managed by SBB shown in Figure 3.

Figure 3: Test network.
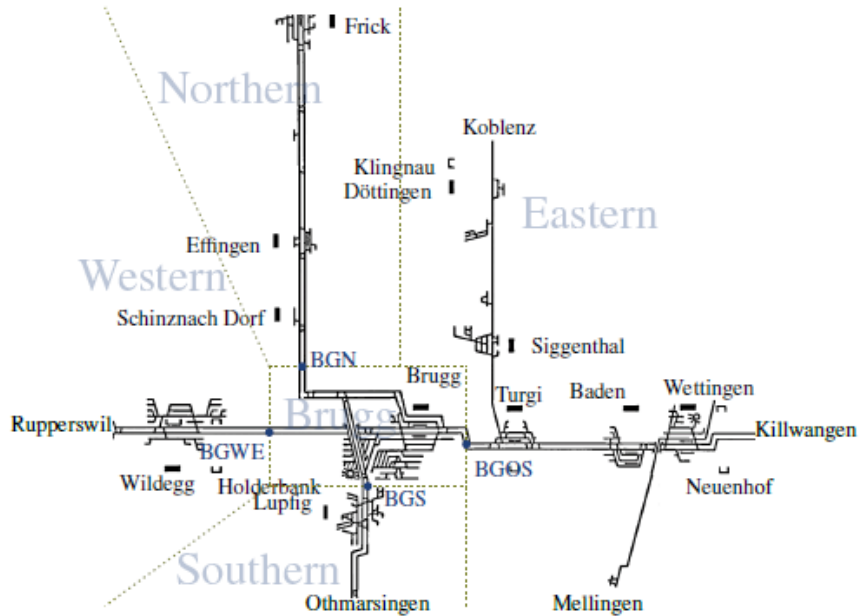
SBB provided the infrastructure data at track circuit level. SBB also made available one week of operational data to calibrate the delay distributions and to extrapolate the rolling stock characteristics of freight trains. The timetable and the rolling stock characteristics of passenger trains in the test areas were openly accessible online. The areas have been defined according to the track topology and the amount of traffic. The four boundaries are no-wait portals according to Definition 3.3, which makes the decomposition a valid no-wait partitioning according to Definition 3.4. The four portals are double track sections where operational rules prescribe that all trains use the left track (i.e. no routing alternative).

The rescheduling algorithms were developed in C++ executables that exploit CPLEX C++ libraries and an ad-hoc developed column generation approach to solve the local rescheduling problems (please refer to Toletti, 2018, for more detailed information on this approach and on the test settings). Five randomly generated scenarios have been run in the simulation tool OpenTrack, which interacted with the algorithms. The centralised RCG approach over the five areas has been run against the distributed, non-coordinated, resolution of the five local RCG models, and the non-hierarchical coordination approach with different updates of the Lagrangian multipliers:

- The subgradient method described in Section 3.3;

- the negotiation-based approach described in Section 3.4 with uniform weights;

- the negotiation-based approach with hierarchy weights, where the two condensation areas (Brugg and Eastern) have the highest hierarchy rank and the three compensation areas the lowest;

- the negotiation-based approach with direction-based weights.

Table 3 summarises the results. The experiments showed empirically that the coordination approach with hierarchy weights reduces the inconsistency of passing times at the boundaries with respect to the independent resolution of the local problems.

The solutions of the coordinated approach with hierarchy weights were comparable with the ones of the centralized rescheduling problem over the five areas. Given the reduced sizes of the local problems, they could be solved several times to improve consistency without exceeding the 180 seconds threshold usually applied for real-time rescheduling problems. The overall computation times of the distributed approach were higher than the centralised one. This is probably due to the small size of the analysed network which could still be solved in a centralised way by the compute server.

Remark: At this point the reader is most probably wondering how can the distributed approach give better objective values (train delays) than the centralised. This is due to the fact that the blocking time stairway patterns used to generate the variables of the RCG models where built according to the green wave principle. This requires large headways for trains to only encounter green signals. In over-saturated scenarios, as the test ones, this usually increases train delays. The softening of the consistency constraints at the portals by the Lagrangian relaxation allowed shorter headways which reduced the overall delay.

Figure 4 displays the time differences between the schedules produced by the neighbouring areas at the four portals for one of the five instances. The horizontal axis measures the

---

[1]The executable runs on the same 64-bit linux machine with four CPUs Intel(R) Core(TM) i5 CPU 760 at 2.80GHz and 7.7 GB memory (RAM).

Table 3: Results of coordinated rescheduling of several delayed scenarios of SBB test case.

delay set	strategy	train delay [s]	#stops at signals	#braking for route	#stops at OPs	#retimings	#reroutings	computation time[1] [s]
1	cgApP centr.	4104	12	17	124	109	40	34.54
	cgApP distr.							
	-none	5097	12	24	107	94	17	15.46
	-subgradient	3987	14	20	90	93	14	21957.52
	-hierarchy	3936	11	19	103	93	15	91.63
	-uniform	5128	12	26	104	95	12	90.51
	-direction	5128	12	26	107	95	17	90.79
2	cgApP centr.	3791	4	12	129	99	9	29.42
	cgApP distr.							
	-none	3592	10	22	101	93	21	12.36
	-subgradient	1375	9	16	94	86	31	24819.4
	-hierarchy	3006	12	27	92	87	22	97.92
	-uniform	3726	11	21	98	92	12	94.55
	-direction	3592	11	22	100	93	15	94.94
3	cgApP centr.	4100	2	12	128	99	24	34.52
	cgApP distr.							
	-none	4915	5	18	104	92	9	12.83
	-subgradient	1846	16	22	83	88	14	19194.46
	-hierarchy	2115	14	20	86	85	15	86.65
	-uniform	5107	6	17	104	93	16	90.55
	-direction	4915	5	18	104	92	9	93.74
4	cgApP centr.	2265	8	17	117	99	13	25.10
	cgApP distr.							
	-none	2544	11	21	95	83	8	9.61
	-subgradient	1975	14	24	91	80	17	940.10
	-hierarchy	2062	12	24	99	80	15	100.34
	-uniform	2109	14	24	89	73	25	89.34
	-direction	2544	11	21	95	83	8	98.55
5	cgApP centr.	2949	9	15	125	106	17	33.69
	cgApP distr.							
	-none	3481	12	24	105	92	9	12.49
	-subgradient	2348	11	16	90	81	12	23504.07
	-hierarchy	2277	12	18	102	88	16	138.52
	-uniform	3204	23	29	88	80	20	135.86
	-direction	3481	12	24	105	92	9	140.17

day time in seconds from midnight, the vertical labels the trains passing through each portal. A segment links the passing times at the portal assigned by the two adjacent rescheduling algorithms. The longer the segment, the greater the time inconsistency of the global schedule. The grey segments represent the case with no coordination, while different colors are assigned to the different multipliers' updates proposed in this work.

Figure 4 shows that our implementation of the subgradient method resulted insufficient to bring the neighbours to a commonly agreed solution. Instead, it forced several trains outside the considered horizon. The negotiation-based approach with all three weight settings resulted in the same problem at the lowest utilized portal BGWE, while it showed better results elsewhere. In particular, the hierarchy weights provided satisfactory results, in which the segments of the trains travelling in the same direction (even numbers/odd numbers) do not overlap vertically.

Figure 4: Time differences at the boundaries of adjacent rescheduling areas. The times assigned by the adjacent RCGs are linked by a segment to highlight the inconsistencies.
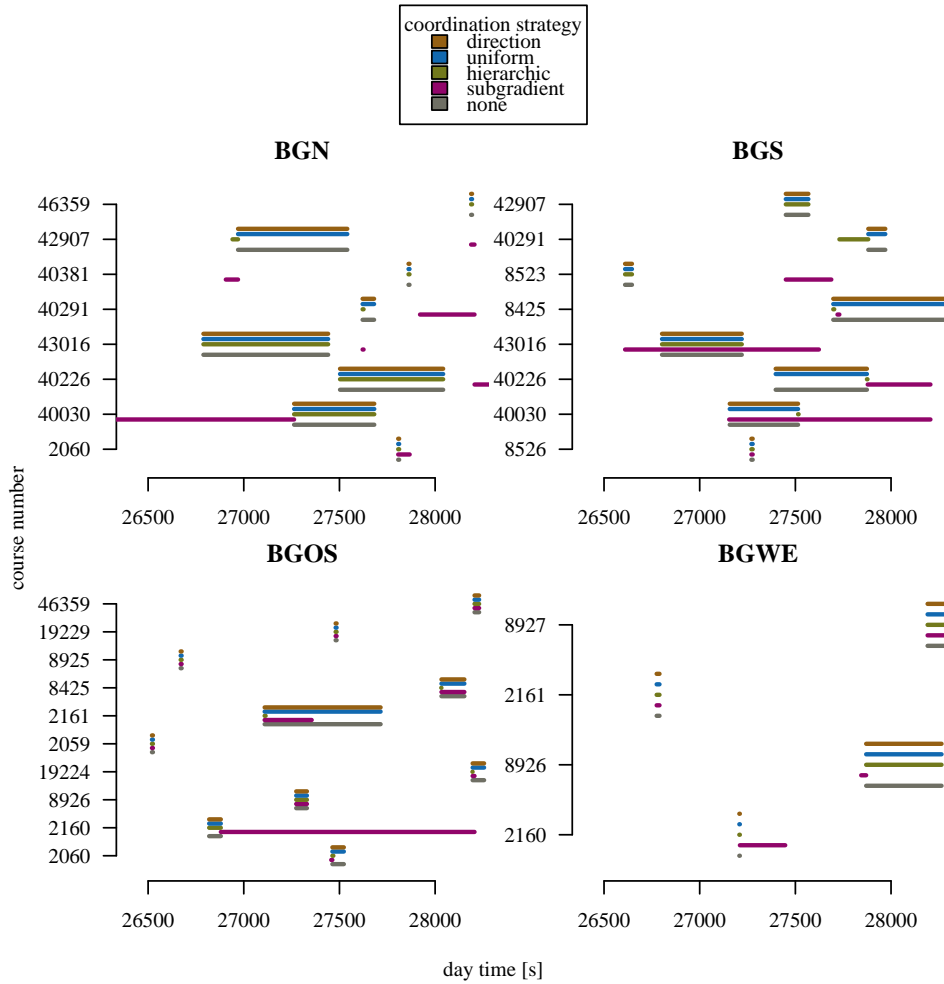
5 Conclusions

This paper focussed on problem decomposition and coordination for real-time rescheduling, a still open research question up-to-now. A methodology based on the Resource Conflict Graph model has been developed. Tests on a large subset of the SBB network showed empirically that this methodology provides schedules that are comparable with the centralised resolution of a unique, huge, problem. Assigning hierarchy weights that consider the traffic density in each partition to the areas performed much better than other settings in improving the consistency of decisions at the boundaries. The improvement with respect to the independent resolution of the local rescheduling problems is considerable. The approach enabled the coordination of two adjacent congested areas, which overcomes the need for

separation of such regions by some compensation areas that has characterized the previous approaches (e.g. Caimi, 2009; Fuchsberger, 2012; Laube et al., 2007). This enables large railway traffic rescheduling problems to be decomposed according to the performance of the available compute server such that the overall computation time of a new schedule meets the requirements of real-time operations.

Acknowledgements

References

Boyd, S., L. Xiao, and A. Mutapcic (2003). Subgradient methods.

Cacchiani, V., D. Huisman, M. Kidd, L. Kroon, P. Toth, L. Veelenturf, and J. Wagenaar (2014). An overview of recovery models and algorithms for real-time railway rescheduling. *Transportation Research Part B: Methodological 63*(0), 15–37.

Caimi, G., M. Fuchsberger, M. Laumanns, and M. Lüthi (2012). A model predictive control approach for discrete-time rescheduling in complex central railway station areas. *Computers and Operations Research 39*(11), 2578–2593.

Caimi, G. C. (2009). *Algorithmic decision support for train scheduling in a large and highly utilised railway network*. Dissertation no. 18581, ETH Zurich, Institut für Operations Research.

Corman, F., A. D'Ariano, D. Pacciarelli, and M. Pranzo (2010). Centralized versus distributed systems to reschedule trains in two dispatching areas. *Public Transport 2*, 219–247.

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

Dollevoet, T., F. Corman, A. D'Ariano, and D. Huisman (2014). An iterative optimization framework for delay management and train scheduling. *Flexible Services and Manufacturing Journal 26*(4), 490–515.

Fischer, F. and T. Schlechte (2017). Strong relaxations for the train timetabling problem using connected configurations. In G. D'Angelo and T. Dollevoet (Eds.), *17th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2017)*, Volume 59 of *OpenAccess Series in Informatics (OASIcs)*, Dagstuhl, Germany, pp. 11:1–11:16. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

Fuchsberger, M. (2012). *Algorithms for railway traffic management in complex central station areas*. Dissertation no. 20398, ETH Zurich.

Hansen, I. A. and J. Pachl (Eds.) (2014). *Railway Timetabling and Operations* (2 ed.). Hamburg, Germany: DVV Media Group GmbH, Eurailpress.

Herrigel, S., M. Laumanns, J. Szabo, and U. Weidmann (2013). Periodic railway timetabling with hierarchical decomposition in the PESP model. In *5th International Seminar on Railway Operations Modelling (RailCompenhagen2013)*.

Jia, L.-M. and X.-D. Zhang (1994). Distributed intelligent railway traffic control: A fuzzy-decisionmaking-based approach. *Engineering Applications of Artificial Intelligence 7*(3), 311–319.

Kersbergen, B., T. van den Boom, and B. D. Schutter (2016). Distributed model predictive control for railway traffic management. *Transportation Research Part C: Emerging Technologies 68*, 462–489.

Laube, F., S. Roos, R. Wüst, M. Lüthi, and U. Weidmann (2007). Puls 90 - ein systemumfassender Ansatz zur Leistungssteigerung von Eisenbahnnetzen. *ETR - Eisenbahntechnische Rundschau 03*, 104–107.

Narayanaswami, S. and N. Rangaraj (2015). A MAS architecture for dynamic, realtime rescheduling and learning applied to railway transportation. *Expert Systems with Applications 42*(5), 2638–2656.

Nedic, A. and A. Ozdaglar (2009). Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control 54*(1), 48–61.

Salido, M., M. Abril, F. Barber, L. Ingolotti, P. Tormos, and A. Lova (2007). Domain-dependent distributed models for railway scheduling. *Knowledge-Based Systems 20*(2), 186–194.

Terelius, H., U. Topcu, and R. M. Murray (2011). *Decentralized Multi-Agent Optimization via Dual Decomposition*, Volume 44, pp. 11245–11251.

Toletti, A. (2018). *Automated railway traffic rescheduling and customer information*. Ph. D. thesis, ETH Zurich.

Toletti, A., V. De Martinis, and U. Weidmann (2015). What about train length and energy efficiency of freight trains in rescheduling models? *Transportation Research Procedia 10*, 584–594.

Toletti, A., M. Laumanns, V. De Martinis, and U. Weidmann (2017). Multicriteria train rescheduling by means of an adaptive epsilon-constraint method. In *6th International Conference on Railway Operations Modelling and Analysis (RailLille2017), Lille, France, April 4-7, 2017*.

Toletti, A. and U. Weidmann (2016). Modelling customer inconvenience in train rescheduling. In *Swiss Transport Research Conference (STRC2016)*.

Xie, X.-F., S. Smith, T.-W. Chen, and G. Barlow (2014). *Real-time traffic control for sustainable urban living*, pp. 1863–1868.