An iterative heuristic for passenger-centric train timetabling with integrated adaption times

Gert-Jaap Polinder[1,2], Valentina Cacchiani[3], Marie Schmidt[1,2] and Dennis Huisman[2,4,5]

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

[2] *Erasmus Center for Optimization in Public Transport (ECOPT), Rotterdam, The Netherlands*

[3] *DEI, University of Bologna, Viale Risorgimento 2, I-40136 Bologna, Italy*

[4] *Econometric Institute, Erasmus School of Economics, Erasmus University Rotterdam, P.O. Box 1738, 3000DR Rotterdam, The Netherlands*

[5] *Process quality and Innovation, Netherlands Railways, P.O. Box 2025, 3500HA Utrecht, The Netherlands*

Abstract

We aim at constructing a timetable that minimizes average perceived passenger travel time, which, in addition to the in-train and transfer times, includes the adaption time (waiting time at the origin station). Adaption time minimization allows us to avoid strict frequency regularity constraints and, at the same time, to ensure regular connections between passengers' origins and destinations. Besides considering safety restrictions (i.e., headway times, overtaking and crossing constraints), passenger routing, based on origin-destination demand pairs, must be taken into account when building the timetable.

This problem can be modelled as an extension of a Periodic Event Scheduling Problem (PESP) formulation, but cannot be directly solved by a general-purpose solver for our real-size instances. In this paper, we propose a heuristic approach consisting of two phases that are executed iteratively. First, we solve a simplified timetabling model, and determine an ideal timetable that minimizes the average perceived passenger travel time but neglects safety restrictions. Then, a Lagrangian-based heuristic makes the timetable feasible by modifying train departure and arrival times as little as possible. The obtained feasible timetable is then evaluated to compute the resulting average perceived passenger travel time, and a feedback is sent to the Lagrangian-based heuristic so as to possibly improve the obtained timetable from the passenger perspective, while still respecting safety constraints. We have tested the proposed iterative heuristic approach on real-life instances of Netherlands Railways, showing that it converges to a feasible timetable very close to the ideal one.

Keywords: public transportation planning, heuristic, integration of timetabling and passenger routing, adaption time

1 Introduction

Recent years have seen an increased attention to environmental pollution in terms of carbon dioxide and nitrogen emissions. Road transportation is responsible for the majority of these emissions [CBS, PBL, RIVM, WUR, 2019]. Combining this with the high levels of congestion on the streets, a shift towards public transportation is desired. However, currently many trains are full, in particular in rush hours, so more capacity is needed. This leads to a higher number of trains on the tracks. The problem of designing a good railway timetable with limited infrastructure resources is a challenging problem. Increasing the frequencies of trains makes it even more complicated to find a good timetable. In a good timetable, travel options for passengers are spread regularly over time, so passengers can travel whenever they like and do not have to wait a long time. However, when infrastructure resources become limiting to accommodate all trains, this regularity of train services cannot always be realised. In such a situation, trade-offs have to be made between having a regular service or having increased waiting times. Furthermore, improving the level of service on one part of the network often implies a decreased level of service on other parts of the network.

In this paper, we aim at constructing a timetable that minimizes average perceived passenger travel time, and that can be safely operated on a given infrastructural network. The perceived travel time does not only consist of the time a passenger actually travels, but also includes *adaption time*. Adaption time is the time difference between the desired moment of departure and the actual departure time. That means, if a passengers would like to depart at :40, and can depart only at :50, the adaption time is 10 minutes. As is common in many European countries, we compute a periodic timetable, i.e., a timetable for a base period that is repeated throughout the day.

We consider infrastructure constraints on a macroscopic level. That is, we consider railway stations with a number of tracks connecting them. In macroscopic timetabling, headway constraints impose a minimum time difference between trains that share part of the tracks to avoid crossings and overtakings that are not possible due to the infrastructure and to enforce a minimum safety distance between trains running on the same track.

Many timetabling approaches impose an upper time limit on transfer times and regularity of services by additional constraints. Instead of using these strict constraints, we omit them, since our objective, the minimization of the perceived travel time of passengers, will penalize long transfer times and waiting times at origin stations. In particular, adaption time minimization allows to effectively synchronize trains at stations based on passenger demand. Furthermore we allow to cancel planned trains if that leads to a better objective value.

We define the *Passenger-Oriented Timetabling (POT)* problem as follows: Given an infrastructure network with stations and tracks connecting them, and a line plan, specifying line routes, stopping patterns and frequencies: find a timetable including all or a subset of the trains that satisfies the headway restrictions induced by the infrastructure network and minimizes average perceived travel time, where we assume that passengers will travel on shortest route according to perceived travel time.

POT can be formulated as a mixed-integer linear program combining a periodic event scheduling model with an approach for modelling average perceived travel time including adaption time as developed in Polinder et al. [2020] for strategic timetabling. However, the resulting model is very difficult to solve. Therefore, in this paper we propose an iterative approach that combines (extended versions of) two existing approaches. First, we compute an *ideal* timetable, that is: a timetable that does not need to respect infrastructure restrictions, using the method proposed in Polinder et al. [2020] for strategic timetabling. Secondly, we transform this ideal timetable into a feasible timetable, that is: make it satisfy the infrastructure restrictions, using an extension of the Lagrangian heuristic (LH) proposed in Cacchiani et al. [2010] with the goal to find a timetable that stays as close as possible to the ideal timetable, but satisfies the infrastructure restrictions. In a next step, we compare the resulting timetable to the ideal timetable and evaluate how the changes influence the quality of the timetable. Based on this, we provide feedback to the Lagrangian heuristic to improve the quality of the newly found timetable.

Our contribution in this paper is threefold: First, we define the POT problem, which calls for determining a timetable that minimizes the average perceived travel time (that includes the adaption time) and satisfies safety restrictions. Second, we propose an iterative approach to POT that extends and combines an integer programming approach to find an ideal (but possibly infeasible) timetable with a Lagrangian heuristic to repair this timetable to feasibility, and employs feedback from the former to the latter to improve the timetable from the passenger perspective. Third, we demonstrate our approach on three case studies on the Dutch railway network. We show that our algorithmic approach performs better than the alternative of directly incorporating headway restrictions in the integer program for timetabling, and that it converges to a feasible timetable very close to the ideal one.

The remainder of this paper is organized as follows. In Section 2, we give an overview on research that is related to and relevant for this study. In Section 3 we introduce and define the POT problem in detail. Afterwards, we describe our iterative approach to solve this problem in Section 4. We test our approach on three case studies on the Dutch railway network in Section 5. Finally, the paper is concluded in Section 6.

2 Related Work

Timetabling with PESP. The problem of finding a train timetable is extensively studied in the literature. Often, the timetabling problems are modelled as the problem of assigning times to nodes ('events') in a graph ('event-activity network') where arcs ('activities') represent the time constraints. This formulation makes periodic timetabling a special case of the Periodic Event Scheduling Problem (PESP, Serafini and Ukovich [1989]), which can be used for various applications with periodically recurring events. Details how PESP can be applied in railway timetabling problems can be found in Odijk [1996], Peeters [2003]. An overview on what can be included in a PESP framework regarding periodic timetabling can be found in Liebchen and Möhring [2007].

Essentially, PEPS is a feasibility problem, as its task is to find a feasible solution, satisfying a set of restrictions. Several approaches exist to solve PESP, for example using constraint programming [Schrijver and Steenbeek, 1993] or by transforming PESP into a SAT formulation [Großmann et al., 2012]. PESP is known to be NP-hard [Serafini and Ukovich, 1989], although for real-life instances feasible solutions can often be found in a short time by the aforementioned approaches.

Often PESP is extended by an objective function that is to be optimized (cf. Peeters [2003], Liebchen [2008], Caimi et al. [2017]). These objective functions can cover a number of subjects, like optimizing the customer satisfaction, minimizing the costs of the operator, finding a timetable that is as close as possible to an infeasible input timetable, or computing a delay-resistant timetable [Cacchiani and Toth, 2012, Lusby et al., 2018]. The optimisation version of PESP is normally much more computationally challenging than solving the feasibility problem. Approaches to solve such problems cover various techniques, like integer-programming [Liebchen, 2008, Nachtigall, 1994, Liebchen and Peeters, 2009], a modulo-simplex heuristic [Nachtigall and Opitz, 2008] or combining machine-learning with a SAT formulation [Matos et al., 2018]. In our paper, we employ a PESP-based model for the first phase of our approach, in which the ideal timetable is computed based on the average perceived travel time.

Timetabling based on time-space graphs. Time-space graphs constitute an alternative graph-based modelling approach to event-activity networks. In these approaches, time is discretized, and a time-expanded network is used:nodes correspond to events at specific time instants, and a path in the graph corresponds to a timetable. In time-space graph models, variables represent the choice of arcs (or paths) of this graph. Approaches based on these kind of models have mainly been used for aperiodic timetabling, although recent works have shown their effectiveness for periodic timetabling as well [Martin-Iradi and Ropke, 2019, Zhang et al., 2019].

An advantage of time-space graph models is that constraints on running and dwelling times of a single train are directly embedded within the definition of the graph: only arcs corresponding to feasible running or dwelling times are added. In addition, computing a timetable for a single train corresponds to solving a shortest path problem, and can be efficiently done by dynamic programming algorithms. Time-space graph models easily allow the option of not scheduling some trains by assigning them a dummy path: this is particularly useful when a feasible solution containing all trains does not exist, for example in highly congested networks. The drawback of these models is the size of the graph that can be extremely large for practical instances. For this reason, most approaches in this category solve the timetabling problem heuristically by decomposing it through column generation or Lagrangian relaxation, i.e., trains are scheduled in sequence. As a consequence, it is difficult to handle constraints that involve multiple trains.

Several works propose models based on time-space graphs. Brännlund et al. [1998] apply Lagrangian relaxation of infrastructure constraints, and propose a heuristic algorithm based on this relaxation that uses subgradient optimization and bundle methods. A similar approach is developed in Caprara et al. [2002] for timetabling on a corridor: safety restrictions are relaxed in a Lagrangian way, and near-optimal multipliers are obtained through a subgradient optimization procedure. Cacchiani et al. [2010] extend the time-space graph based approach from Caprara et al. [2002] to insert freight trains into an existing aperiodic timetable, staying as close as possible to given *ideal* timetables for the freight trains. Cacchiani et al. [2008] and Martin-Iradi and Ropke [2019] propose column-generation based methods for models in which variables represent paths in the time-space graph. Zhang et al. [2019] propose a multi-commodity network flow model for periodic timetabling, and apply Lagrangian relaxation and Alternating Direction Method of Multipliers. Recently Ait-Ali et al. [2020] present a bundle method based on a disaggregate approach, where the optimisation is performed with separate dual information for each train.

In our paper, we extend the heuristic from Cacchiani et al. [2010] to deal with periodic timetabling, and use it in the second phase of our approach, to make a given *ideal* timetable feasible.

Passenger-centric objective functions. There are several approaches to measure the quality of a timetable from the viewpoint of the passengers, and, as shown in Hartleb et al. [2019], the choice of evaluation approach will have an impact on which timetables are considered to be 'good' and 'optimal'.

A common approach in OR approaches to timetabling is to minimize the total passenger travel time. The most simple models for measuring and optimizing travel time within a PESP approach minimize a function over the weighted durations of the activities in the timetabling instance, where weights represent the number of passengers using that activity (cf. Peeters [2003]). This relies on the assumption that it is a priori known on which activities passengers travel.

Schmidt and Schöbel [2015] propose a mixed-integer linear programming model that *integrates* (aperiodic) timetabling and passenger routing. Borndörfer et al. [2017] provide a similar, PESP-based model for the periodic case and study the impact of different routing assumptions.As PESP is already a challenging problem in itself, the integration of this problem with passenger routing makes it even more difficult to find good solutions. Schiewe and Schöbel [2019] propose an 'applicable' approach that relies on (heuristic) preprocessing and bound generation. Other approaches [Lübbe, 2009, Siebert and Goerigk, 2013] solve the problem iteratively: first passengers are routed through the network. Based on these fixed routes a timetable is computed. Then passengers are rerouted based on the timetable. This is repeated until a stopping criterium is met.

Martin-Iradi and Ropke [2019] propose a time-space-graph-based approach to find periodic timetables minimizing passenger travel time, and include frequency constraints to guarantee that trains of the same line are spread along the cycle time. In a column generation approach that is designed to minimize the travel times of the trains, each feasible solution found during the process is evaluated with respect to the passenger travel time, and the best solution is kept. In Farina [2019], the same problem as in Martin-Iradi and Ropke [2019] is considered and modelled on a time-space graph, and a Large Neighbourhood Search algorithm is proposed.

The above-mentioned approaches have in common that they evaluate timetables based on the assumption that every passenger will choose the shortest route (with respect to (perceived) travel time) towards his destination, just as we do. It is well understood in transport modelling, however, that not all passengers will choose the shortest route [de Dios Ortúzar and Willumsen, 2011]. Instead, in transport modelling discrete choice models are used to describe how passengers distribute over different route options. Hartleb and Schmidt [2019] investigate how to integrate passenger distribution models instead of routing along shortest routes into PESP.

Including adaption time into passenger-centric objectives. Besides the travel time between departure at the origin and arrival at the destination, also the number of travel options between origin and destination and their timing play a crucial rule in evaluating timetables from a passenger perspective [de Dios Ortúzar and Willumsen, 2011]. E.g., a timetable with four travel options between origin and destination, offered every fifteen minutes, would most likely be preferred to a timetable where there is just one such option (or four, all departing at the same time), even if in the latter case the travel time is slightly shorter.

Focusing solely on passenger travel time, measured from departure at the origin in the evaluation of a timetable, neglects the effect that the spread of travel options over time has on the quality of a timetable. This can be overcome by including *adaption time* in the objective function, while making an assumption on the distribution of 'desired departure times' of passengers over time.

There are several publications on timetabling on lines and corridors, where adaption time is explicitly included in the objective function. Often, 'adaption time' is called 'waiting time' in this context. We use the term 'adaption time' throughout this literature review, also when referring to literature where the authors use the term 'waiting time', to avoid confusion with the waiting time at transfers. For single rail rapid transit lines, Barrena et al. [2014b] and Barrena et al. [2014a] propose, respectively, an exact and an adaptive large neighborhood search minimizing adaption time. A single rail line is also considered in Zhu et al. [2017], where a bi-level model is proposed: the upper level model determines the train headway times to minimize the total passenger perceived costs (given by adaption time, in-vehicle time and penalty costs associated with arriving at the destination outside the desired interval), while the lower level determines passenger arrival times at their origin stations. A genetic algorithm is used to solve it. Yin et al. [2017] proposed an integrated approach to determine train schedules and speed profiles with the aim of minimizing energy consumption and passenger adaption time. A Lagrangian based algorithm is developed for solving it for a bidirectional urban metro line. A rail corridor is considered in Niu et al. [2015], where a quadratic model to determine train timetables based on given time-varying passenger demand data is proposed. It aims at minimizing the total adaption time at stations, and is solved by GAMS [GAMS] for a high-speed rail line.

Wang et al. [2015] propose a very detailed event-driven model for timetabling on urban networks with the objective to minimize a weighted sum of travel time (including adaption time) and energy consumption. Their solution approach is based on sequential quadratic programming and a genetic algorithm, and tested on a small network with two cyclic lines with the Matlab optimization toolbox [The Mathworks].

Instead of including adaption time into the objective function, Gattermann et al. [2016] group passengers into time slices, and add a penalty to the objective function if passengers do not depart in the respective time slice. They propose a (non-linear) PESP-based mathematical programming formulation, and transfer it to a SAT formulation to solve it. While the model allows to group passengers into (predefined) time slices and penalize deviation from the respective time slice, a heuristic to including adaption time, in the numerical experiments reported in Gattermann et al. [2016] however, only one time slice (that spans the whole period) is used.

Polinder et al. [2020] consider timetabling in the strategic railway planning phase. Like in the POT problem, they aim at finding a periodic timetable that minimizes perceived travel time (a weighted sum of in-train, transfer, and adaption time and transfer penalties) under the assumption that passenger demand is uniformly distributed over the period. However, in contrast to the POT problem, they do not consider infrastructure constraints, arguing that these are not relevant in the strategic planning phase. In this paper, we use the approach developed in Polinder et al. [2020] for strategic timetabling to compute an *ideal* timetable in the first phase of our solution approach. It is therefore described in more detail in Section 4.2.

Compared to the existing literature on passenger timetabling, we include the adaption time minimization in the objective, instead of having strict regularity constraints in order to gain flexibility. In addition, we consider a railway large network while most works tackle the problem on a single line or corridor.

3 Problem Description

3.1 Input

The timetable that is to be designed is based on three items: First, the infrastructure network on which the trains operate. Second, an origin-destination matrix representing passenger demand. Third, a line plan that specifies line routes and frequencies.

We consider an infrastructure network on which the trains have to be operated. There are generally three levels of detail on which such a network can be considered: the macroscopic, mesoscopic and microscopic level [Radtke, 2014, Goverde et al., 2016]. As usual in tactical planning, we consider the infrastructure on the macroscopic level [Radtke, 2014, Chapter 3.3]. That means, the network contains the stations, a number of tracks between the stations, estimated driving and dwell times, and headway times between consecutive trains. Further details like block sections and signalling systems are not important at the tactical planning stage and can be included in a later planning stage [Radtke, 2014, Chapter 3.4].

Passenger demand is given in the form of an origin-destination matrix $\mathcal{OD}$. For each OD-pair $k \in \mathcal{OD}$, the corresponding matrix entry d_k represents the number of passengers who want to travel from the origin to the destination in one period.

A line plan specifies a set of train lines that are to be operated on the given infrastructure network. Each train line consists of a route through this network, a list of stations where the train stops (a stopping pattern) and a frequency that specifies how often the line is operated per hour. We assume that all lines are operated in both directions. Note that in the line planning phase, no timetable is known yet. Therefore, while line planning can take into account constraints on the infrastructure utilization, and on eligible frequencies, it is not ensured that there exists a feasible timetable where all trains specified in the line plan can be operated. Therefore, we allow our method to cancel trains if necessary.

3.2 Passenger-oriented timetabling

Based on the infrastructure network, the demand encoded in the OD-matrix $\mathcal{OD}$, and the line plan, the *passenger-oriented timetabling (POT)* problem can be summarized as

$$\text{Minimise} \sum_{k \in \mathcal{OD}} d_k \cdot R_k(\pi) \tag{3.1}$$

$$\text{Such that TimetablingRestrictions}(\pi) \tag{3.2}$$

$$\text{RoutingRestrictions}_k(\pi) \qquad \forall\, k \in \mathcal{OD}. \tag{3.3}$$

Here, π is the timetable, d_k is the demand for OD-pair k and $R_k(\pi)$ is the average perceived travel time for the passengers of OD-pair k.

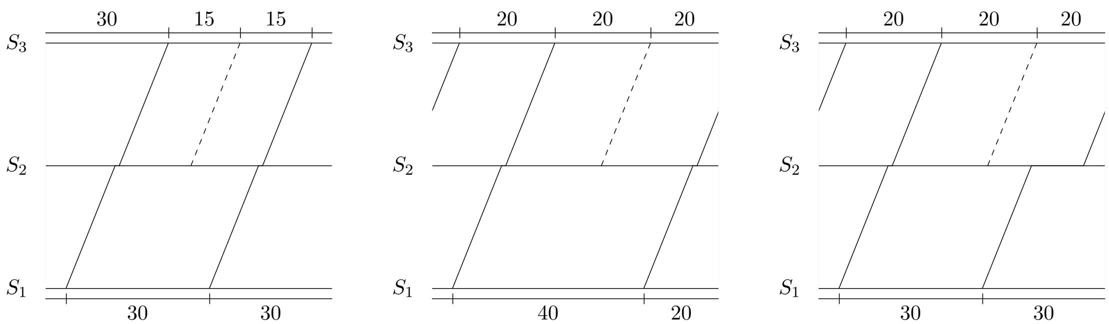
We aim at finding a timetable π that is operationally feasible (as ensured in constraints 3.2) and integrate the routing of passengers through the network (constraint (3.3)) such that the average perceived travel time (objective (3.1)) is minimized. The timetabling restrictions 3.2 can be formulated as standard PESP constraints, with the particularity that synchronization constraints and upper bounds on transfer times are not included. See Section 3.2.2 for details.

Constraints (3.3) are auxiliary constraints that compute passenger routes according to shortest perceived travel times in the timetable, to be able to evaluate the timetable with respect to total perceived travel time in the objective (3.1). The modelling of these constraints and the objective is described in Section 3.2.3.

By optimizing with respect to total perceived travel time, which includes the adaption time, we are able to exclude synchronisation constraints and upper bounds on transfer times from our modelling, and thus trade-off the synchronization on different part of the network in our model. This is illustrated in the following section, using an example from Polinder et al. [2020].

3.2.1 Example

The inclusion of adaption time in the objective function allows us to trade-off regularity of the timetable on different lines and dwelling times of trains at intermediate stations. This is best illustrated through an example taken from Polinder et al. [2020]. There are three stations S_1, S_2, S_3 and two train lines; L_1 and L_2. L_1 visits all three stations (with a frequency of two trains per hour), and L_2 only visits S_2 and S_3 (with a frequency of one train per hour). If we consider only passengers traveling from S_1 to S_2 and S_3, an optimal timetable spreads train departures evenly over time, so that every 30 minutes a train departs from S_1, and the average waiting time at S_1 is 15 minutes. Unnecessary dwelling time at S_2 should be avoided to keep in-train times for passengers to S_3 low, leading to a departure of L_1-trains every 30 minutes from S_2. However, in order to optimize the timetable for passengers from S_2 to S_3, trains from S_2 should depart every 20 minutes. Depending on passenger numbers for the different OD-pairs and the weighing factor for the adaption time, we would thus trade-off the three factors: regularity on the first part of the network, regularity on the second part of the network, and dwell time at S_2. The three options are visualized in Figure 1.



(a) $S_1 \rightarrow S_2$ synchronised

(b) $S_2 \rightarrow S_3$ synchronised

(c) All synchronised with additional dwell time at S_2

Figure 1: Time space diagrams for different synchronisation options (taken from Polinder et al. [2020])

3.2.2 Timetabling restrictions

In this section we describe the modeling of the constraints (3.2), for which we use the periodic-event scheduling approach, first described by Serafini and Ukovich [1989] which is very common in periodic railway timetabling.

The timetabling constraints are formulated based on the line plan and the infrastructure network. More precisely, the line plan defines a set of events V, the departures and arrivals of the trains at stations, which need to be scheduled. As we consider a periodic timetable, these events are periodic, i.e., they re-occur every time period. The event times have to satisfy several restrictions to form a feasible timetable. There are minimum driving times between stations, constraining the arrival time of a train to the preceding departure time. Furthermore, in stations we have minimum dwell times to let passengers board and alight. We also have minimum transfer times to make connections between trains. Additional restrictions can be added to ensure a safe operation of trains, given the infrastructure network. In macroscopic timetabling, a safe operation is ensured through headway restrictions upon departure and arrival at stations. Headway restrictions ensure that a minimum time between two consecutive trains is respected upon departure and arrival at a station, in order to avoid collisions. We also consider overtaking constraints, ensuring a minimum headway time is respected on tracks between stations, when the headway constraints only are not enough to prevent overtakings. Finally, crossing constraints ensure that if two trains use a track in opposite directions, they can do this safely, without meeting each other halfway.

In periodic timetabling, the restrictions are often referred to as *activities*. All activities are stated between pairs of events. The set of activities is denoted by A. The events V and activities A together can be visualized as an *Event-Activity Network (EAN)*.

A periodic timetable is an assignment of times in to events, i.e., it can be denoted as $\pi : V \mapsto \{0, 1, \ldots, T-1\}$. A commonly used model for periodic railway timetabling on a macroscopic level is the *Periodic Event Scheduling Problem (PESP)* [Serafini and Ukovich, 1989]. The task in PESP is to assign arrival and departure times to the events V, i.e., find a timetable $\pi : V \mapsto \{0, 1, \ldots, T-1\}$, fulfilling periodic time constraints of the form

$$(\pi_j - \pi_i - \ell_{ij} \mod T) + \ell_{ij} \in [\ell_{ij}, u_{ij}] \qquad \forall\, (i,j) \in A, \tag{3.4}$$

where ℓ_{ij}, u_{ij} model lower and upper bounds on the time difference between events i and j, and T is the cycle period of the timetable. This can be reformulated for use in an integer linear program as

$$\pi_j - \pi_i + Tp_{ij} \in [\ell_{ij}, u_{ij}] \qquad \forall\, (i,j) \in A, \tag{3.5}$$

where p_{ij} is a binary variable denoting the modulo operator, i.e., a shift from one cycle to the next. All aforementioned activities can be formulated as a PESP-constraint. An overview of other activities that can be formulated in a PESP context can be found in Liebchen and Möhring [2007], Kroon et al. [2014].

As mentioned in Section 1, we do not impose upper bounds on transfer times. However, each PESP-constraint requires an upper bound. We deal with this by setting for such $(i,j) \in A$ the upper bound to $u_{ij} = \ell_{ij} + T - 1$. Due to this, all time differences between events i and j are possible, i.e., there is no operational restriction. When for example $\pi_j = \pi_i + 2$ and $\ell_{ij} = 3$, the time difference between events i and j is two minutes, but also any multiple of T minutes can be added (due to the Tp_{ij} term in (3.5)). In this case, the transfer time is not 2 minutes, but 62 (when $T = 60$), since $\ell_{ij} = 3$ en $u_{ij} = 62$. By this way of modelling, the proper transfer times for passengers can be determined. The same principle holds when upper bounds of, for example, trip and dwell times are omitted. In this way, no operational restrictions are added. In section 4, when our model is explained, these transfer activities are used to determine the correct passenger paths and their perceived duration.

3.2.3 Passenger route choice and evaluation of the timetable

In constraints (3.3) we model that each passenger will choose a route that minimizes his perceived travel time. We model this following the approach outlined in Polinder et al. [2020] for strategic timetabling.

The perceived travel time of a passenger consists of the following components:

Adaption time This is the time difference between the desired departure time of the passenger and the moment the train departs that brings him to his destination. The adaption time is weighted by a factor γ_w.

In-train time This is the time the passenger actually spends in the train, both when the train is driving and when it dwells at a station.

Transfer time This is the time a passenger has to spend on some station to transfer from one train to another. The transfer time is weighted by a factor γ_s.

Transfer penalty If the passenger needs to transfer from one train to another, a penalty of γ_t is added for each transfer. This is done to model the fact that passengers in general do not like to have a transfer [de Keizer et al., 2015].

Note that while the adaption time depends both on the chosen route and on the desired departure time of the passenger, in-train time, transfer time, and transfer penalty are characteristics of the route. We therefore refer to the weighted sum of these as *perceived route length* in the remainder of this paper.

To evaluate the timetable, we follow the approach described in Polinder et al. [2020] and assume that passenger demand per OD-pair is distributed uniformly over the period. I.e., every time unit (in our model: every minute) $\frac{d_k}{T}$ passengers would like to depart from the origin station of OD-pair k to travel to the destination station of OD-pair k. The rationale behind this assumption is that the timetable is usually constructed a number of years to six months before the actual day of operation, and we cannot expect that time-dependent demand is known accurately.

To compute the average perceived travel time for an OD-pair k, $R_k(\pi)$, we group passengers according to their desired average adaption time and the routes they would take correspondingly. This allows us to compute in-train time, transfer time, transfer penalty and average adaption time per route and weigh it with the corresponding passenger number, to obtain $R_k(\pi)$ as a weighted average. If, due to train cancellations, there is no route from origin to destination of OD-pair k, we set $R_k(\pi) := M$, where M is a (high) penalty value. In the remainder of this paper, we refer to $d_k \cdot R_k(\pi)$ as the *evaluation contribution* of OD-pair $k \in \mathcal{OD}$.

To evaluate the timetable we sum up the evaluation contributions of the OD-pairs and obtain

$$\sum_{k \in \mathcal{OD}} d_k \cdot R_k(\pi, Y_k) \tag{3.6}$$

which we use as objective function (3.1) in our problem.

We now explain the grouping of passengers in more detail: We precompute a set of routes $\mathcal{R}^k$ for each OD-pair $k \in \mathcal{OD}$ as paths in the event-activity network. The set of routes is computed based on the line plan and contains routes that could be a reasonable option for OD-pair k. We denote the set of all first departure events on routes for OD-pair k as the set of *relevant* departure events for OD-pair k, V^k.

Note that which route from $\mathcal{R}^k$ a passenger of OD-pair k takes, depends on the desired departure time of the passenger, the time at which the first departure events of the routes are scheduled, and the perceived route lengths. In particular, if, based on a timetable, for each OD-pair we divide the period into time slices according to the scheduled times of the relevant departure events, we know that passengers arriving in the same time slice will choose the same route. We can therefore group the passengers of OD-pair k into $|V^k|$ groups.

Based on the assumption that passenger demand is uniformly distributed over time, the number of passengers in the group associated with relevant departure event v is proportional to the time that has passed since the previous departure event v'. It can be computed as $\frac{d_k}{T} \cdot (\pi_v - \pi_{v'})$. Based on the assumption of uniformly distributed arrivals, the average time between desired departure and π_v is $\frac{(\pi_v - \pi_{v'})}{2}$. (Note that the average adaption time of the group could be longer: in case that the route starting with event v has a high perceived length, the passengers from this group may decide to take a later-departing route.)

The above-made considerations allow us to compute the total adaption time for a given timetable. To integrate these calculations into a mathematical program that determines and evaluates the timetable simultaneously we compute the values A_v^k which denote the number of minutes before event v in which no other departure of a route towards the destination of OD-pair k takes place, as

$$A_v^k := \min_{v' \in V^k} \left\{ \pi_v - \pi_{v'} + T\alpha_{v,v'} \right\}, \qquad \forall\, v \in V^k, \tag{3.7}$$

where binary variables $\alpha_{v,v'}$ account for a shift from one cycle to the next, similar to the modulo operator p_{ij} in (3.5). Then the number of passengers in the group associated with relevant departure event v is given as $\frac{d_k}{T} \cdot A_v^k$ and the average time between desired departure and π_v for members of this group is $\frac{A_v^k}{2}$.

The perceived duration of the shortest route r for passengers in the group associated with relevant departure event v can be modelled as

$$\frac{A_v^k}{2} + \min_{v' \in V^k} \min_{r \in \mathcal{R}_{v'}^k} \left\{ \gamma_w \cdot \left(\pi_{v'} - \pi_v + T\alpha_{v,v'} \right) + Y_r \right\}. \tag{3.8}$$

Here, the first expression corresponds to the adaption time of passengers in the group. As mentioned before, passengers can choose to have a longer adaption time by waiting for a later departure v'. This is modelled by the first term in the minimum. The second term, Y_r, is the perceived route length of route r, starting with departure event v'. Note that the perceived duration Y_r of such a route is now dependent on the timetable but has to be calculated as the (weighted) sum of the duration of the activities used in the route.

3.3 Lower bound and excess evaluation contribution

Based on minimum drive, dwell, and transfer times and penalties, we can compute lower bounds on perceived route lengths. Furthermore, by predetermining routes for each OD-pair, we can compute a lower bound on the total adaption time of one OD-pair, by assuming that departure times for this OD-pair are perfectly synchronized. The sum of the lower bound for perceived route length for an OD-pair, multiplied with the number of passenger for this OD-pair, and the lower bound on adaption time, gives us a lower bound on the OD-pair's perceived travel time. We refer to the difference between the evaluation contribution of an OD-pair and the so-computed lower bound as *excess evaluation contribution*. The sum of lower bounds over all OD-pairs provides us with a lower bound on the evaluation value of the timetable.

4 Solution Approach

This section describes the algorithmic approach that we use to solve the problem. We first outline the overall approach, before we describe the involved steps in more detail.

4.1 High-level description of the solution approach

It is possible to extend the mathematical programming formulation for strategic passenger-oriented timetabling stated in Polinder et al. [2020] to include safety constraints (headway, overtaking and crossing contraints) that model infrastructure requirements, by adding more PESP-constraints. In that way, POT can be modelled as a mixed-integer (quadratic) program. However, the incorporation of the safety restrictions leads to a strongly interconnected event-activity network, which in itself leads to a challenging PESP problem to solve on networks of realistic size (cf. Goerigk and Liebchen [2017], Liebchen et al. [2008]). Combined with the variables and constraints introduced to the model and the objective function, the problem formulation becomes unsuitable to solve large real-world instances on general purpose IP solvers.

Therefore, in this section we propose an iterative approach. A graphical overview of our approach is shown in Figure 2. The numbers mentioned in the sequel refer to the numbers in this figure.

In a first step, we construct an 'ideal' timetable (i.e., one which does not need to take safety restrictions into account) using the solution approach for the strategic timetabling

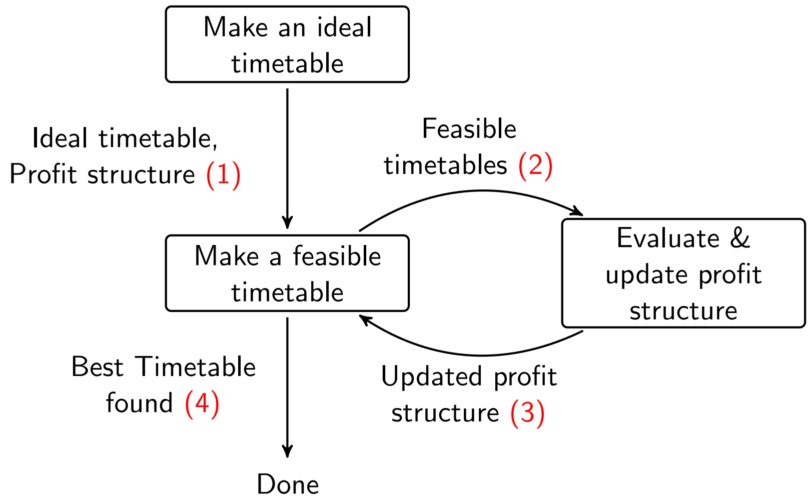


Figure 2: Flow diagram of our approach

problem from Polinder et al. [2020]. This approach is summarized in Section 4.2. An extension of the Lagrangian heuristic (LH) from Cacchiani et al. [2010] is used to modify the timetable to make it feasible with respect to infrastructure restrictions, while staying as close as possible to the ideal timetable. As LH requires a particular structure for the objective (as outlined in Section 4.3), a transition has to be made from one module to the next. This is done by specifying a profit structure for each train in the line plan, that is based on the relative importance of the trains (1). In fact, different profit structures are used, to generate a pool of feasible timetables with LH. We detail in Section 4.3.2 how profit structures are chosen. We evaluate all feasible timetables from the pool of timetables found with LH (2) with the evaluation function (3.6). Based on the evaluation values, we select one or several feasible timetables for comparison with the ideal timetable. We check for which OD-pairs the evaluation contribution is improved became better and for which it gets worse. Based on this, we update the profit structure (3) and rerun LH to hopefully find a better timetable.

We repeat this procedure, until no improvements are found any more. We then end with the best found timetable (4). Each phase of the algorithm is explained in more detail below.

4.2 Make an ideal timetable

We define an *ideal* timetable as a timetable that minimizes average perceived travel time, without necessarily being feasible with respect to infrastructure and safety requirements. When removing all PESP constraints modelling infrastructure and safety requirements from the formulation of POT (3.1)-(3.3), the corresponding event-activity network becomes much less dense. However, for large real-world instances, general-purpose solvers may still struggle to find an optimal (or provably good) solution in a reasonable approach of time. We therefore adopt the solution approach proposed in Polinder et al. [2020], which studies POT problems without infrastructure and safety constraints in a strategic timetabling context.

We first linearise the quadratic objective and the constraints that take a minimum over a set. Secondly, we use the proposed stepwise heuristic to construct a starting solution. This heuristic is related to a 'Relax-and-Fix' heuristic [Belvaux et al., 1998, Wolsey, 1998]. In this heuristic, only the largest OD-pairs are taken into account. Then, all variables are relaxed to continuous variables, except for the variables related to the timetabling restrictions. This leads to a relatively easy model to solve and it leads to a feasible timetable. After that, step by step other variables are changed into integers, to improve the quality of the timetable. In each step, the solution of the previous step is used as a warm start for the new step. After the heuristic finishes, the resulting timetable is used as a warm start to solve the full mathematical model. We use IBM Cplex to solve the various models [IBM, 2019].

4.3 Make a feasible timetable

In this section, we focus on the second phase of our solution approach. After an ideal timetable is computed, we compute a timetable that is feasible, i.e., satisfies all the safety restrictions, and is as similar as possible to the ideal one. A Lagrangian heuristic algorithm (LH) is used in this second phase. It extends the method proposed in Cacchiani et al. [2010], where it has been applied for solving a non-periodic train timetabling problem. For the sake of clarity, we briefly describe, in Section 4.3.1, the main steps of LH, and refer to Cacchiani et al. [2010] for further details. Then, in Section 4.3.2, we present the new features added to LH in order to cope with additional real-world constraints.

4.3.1 Main steps of LH

LH takes as input the description of the infrastructure network, an ideal timetable, and a profit structure. The ideal timetable contains, for every train, the desired departure and arrival times at every visited station. In order to derive a feasible timetable, LH can change the ideal timetable (i) by moving (earlier or later) the departure time of some trains from their origin stations (*shift*) and consequently moving the arrival and departure times, at all the stations visited by the train, by the same amount, (ii) by increasing the dwell time at some of the visited stations (*stretch*) and (iii) by cancelling trains. Each of these changes is undesirable, and thus it is penalised, but not all changes have the same importance. Clearly, train cancellation has a deeper impact on passengers, as the line frequency is reduced or even some OD-pairs might not have a travel option to reach their destinations. However, shift and stretch also affect the passenger travel as they influence the adaption, in-train and transfer times. In addition, the same change applied to different trains (e.g., intercity versus local trains, high- versus low-frequency train lines) has different consequences. In order to obtain a feasible timetable that is as similar as possible to the ideal one, and also to give different importance to the different changes, we define a *profit structure*. In particular, each train is associated a *train profit*, a *shift penalty*, a *stretch penalty*, a *maximum shift value*, and a *maximum stretch value*. The train profit corresponds to the importance of scheduling the train, and is decreased by the shift penalty for every minute of shift, and by the stretch penalty for every minute of stretch. The maximum shift and stretch values represent the bounds on the changes of the respective type that can be applied to obtain a feasible timetable. Note that, as in Cacchiani et al. [2010], we do not consider the option of decreasing the dwell time at a station, nor that of increasing the train travel time between consecutive stations.

To represent the train timetabling problem, LH uses a time-space multi-graph, in which every node corresponds to a train event, i.e., to a departure or an arrival time of a train from/at a station along a track. Arcs represent the travel of a train between two consecutive stations or the stop of a train at a station, and are partitioned into arc sets, one for each train. Different trains can have different travel and dwell times, but it is also possible that two (or more) trains have the same departure and arrival nodes: therefore, there can be multiple arcs (of different trains) between the same nodes, i.e., we deal with a multi-graph. A *path* in this time-space multi-graph corresponds to a train timetable that respects the train travel and dwell times. An Integer Linear Programming (ILP) model based on this time-space graph contains one binary variable for each arc, that assumes value one if the arc is selected in the solution. In this ILP, each arc is assigned a *profit* that is used to obtain a timetable as close as possible to the ideal one: the profit associated with the travel arc of a train from its origin station to the consecutive one is given by the train profit decreased by the shift penalty counted for every minute of shift incurred by that departure time; the profit associated with each arc corresponding to a stop at a station is zero, if that arc corresponds to the minimum dwell time, or is decreased by the stretch penalty counted for every minute of stretch incurred at that station. Clearly, if a train is cancelled, no profit is obtained.

The objective of LH is to maximise the total profit of all trains. The constraints require to select, for each train, arcs that form a path in the time-space graph, and do not conflict with arcs selected for any other train, i.e., satisfy all safety restrictions. Obviously, the difficulty of solving the problem comes from the latter constraints: therefore, LH applies a Lagrangian relaxation of all these constraints. This allows us to easily compute the solution of the relaxed problem (*Lagrangian solution*) by dynamic programming, since it consists of finding, for each train, the most profitable path in the time-space graph. In order to improve the Lagrangian multipliers associated with the relaxed constraints, LH iteratively executes a subgradient optimization procedure, in which Lagrangian multipliers are updated and added to the arc profits, so as to take into account the constraint violations or looseness. Meanwhile, at each iteration of the subgradient optimization procedure, to determine a feasible timetable, LH applies the following steps: (i) it orders trains based on their profits in the Lagrangian solution (*Lagrangian profit*), (ii) schedules one train at a time in the most profitable way while avoiding all conflicts with the previously scheduled trains, and (iii) applies a local search procedure that tries to find a better path for one train at a time (if it had shift, stretch or was cancelled) by keeping all other paths as fixed, and this time considering the original arc profits.

4.3.2 New features of LH

Adapting LH to periodic timetabling. The version of LH developed in Cacchiani et al. [2010] is developed for a non-periodic train timetabling problem. However, it is capable of handling a 'periodicity' of one day: namely, in the non-periodic timetabling problem, the timetable was repeated in the same way every day. In this work, we apply LH to a periodic train timetabling problem, where the cycle time is one hour: in this context, beside changing the length of the period, we need to ensure that the duration of the total shift time window (earlier and later shift) plus the total stretch is smaller than the cycle time. If this is not guaranteed, then we cannot uniquely define the shift or stretch penalty of an arc: for example, a shift of one minute would not be different from a shift of sixty-one minutes. We note that this limitation is reasonable, since usually at least two trains with the same origin and destination stations should be scheduled in each cycle time, and thus it is not useful to globally shift or stretch a train more than the cycle time.

Rolling stock restrictions. Another change is applied to the original LH as described in Cacchiani et al. [2010] in order to deal with basic rolling stock constraints: in practice, usually trains of the same line (i.e., trains with the same origin and destination stations, and stopping at the same intermediate stations) are scheduled in "pairs", so that when a train is scheduled in one direction, another train is also scheduled in the opposite direction. The reason is that, in this way, the same rolling stock (physical train) is assigned to both services, and we also obtain a more regular timetable that has the same number of trains running in both directions. In general, according to the line frequency, there can be more than two trains of the same line in a period: in this case, we need to guarantee that the same number of trains is scheduled in both directions. Since LH allows train cancellation as one of the changes that can be applied to obtain a feasible timetable, we need to guarantee that, if a number of trains of a line is cancelled in one direction, then the same number of trains is also cancelled in the opposite direction. Clearly, this can be easily obtained by simply cancelling additional trains: however, cancelling trains is highly undesirable. Therefore, we modify LH by including a new procedure as follows. At each iteration of the subgradient optimization procedure, when a feasible timetable has been determined and the local search procedure has been executed to improve it, we check, for every train line, the number of trains cancelled in each direction.

If this number is not the same in both directions, then we cancel additional trains, so that the same number of trains are scheduled in both directions. Once all train lines have been processed, for each train line, we try to reschedule trains in pairs by computing, for each train, the most profitable path compatible with the previously scheduled ones: this computation is performed by dynamic programming considering the original arc profits. Note that it is possible to schedule previously cancelled trains thanks to the additional train cancellation applied at the beginning of this procedure. If the same number of feasible paths is found for both directions of a line, then the corresponding train paths are fixed in the execution of this procedure, otherwise trains are cancelled again. Since this procedure can change the set of scheduled trains, after executing it, we apply the local search procedure to possibly further improve the timetable. We observe that, for the considered instances, the number of trains per line is two or four, and thus this procedure can be executed efficiently.

Intermediate shift penalties. A final extension we developed is used for the feedback process, that is applied after the timetables have been evaluated. In this process, LH takes as input the infrastructure description, the ideal timetable, and a new profit structure. Beside the possibility of updating shift and/or stretch penalties, we also include, during the feedback, the option of penalising the shift at some intermediate stations visited by a train (so not only at the origin station of the train). Indeed, we observed that the timetables produced by LH sometimes show irregular departure headway times from some intermediate stations, and it is thus useful to penalise this irregularity. To this aim, we add, for every minute of shift, an *intermediate shift penalty* to the profit of each arc that corresponds to the stop of a train at a station: based on the dwell time at the station, the departure time from that station determines the corresponding intermediate shift. LH is then executed by considering these additional penalties in the computation of the Lagrangian solution and of the feasible timetable, and in the local search procedure.

Note that adding an intermediate shift penalty at a station where a train line starts, has no effect on this specific train line, since there is no arc associated with a dwell time of this train line at this station. However, for departing trains of this line, we do have the regular shift penalty, but this is not penalized on an arc associated with a stop of a train.

4.4 Evaluate & update profit structure

In the remainder of this section, we describe how we provide feedback and update the profit structures, in order to find better timetables.

We evaluate each timetable generated by LH using the evaluation function (3.6). The best timetable according to this evaluation function after running LH is referred to as the *best pure Lagrangian* (BPL) timetable. By comparing the evaluation contributions of all OD-pairs in the ideal timetable π and in the BPL-timetable π', we can identify the OD-pairs for which the evaluation contribution increased the most. We inspect the routes chosen for these OD-pairs and the corresponding trains to find the reason of the increase. In a 'feedback' step, we generate a new set Ψ of updated profit structures, based on the initially chosen profit structure S to penalise the undesired changes more. Since we are not able to predict by how much we should penalize deviation from the ideal timetable, we use a set of penalty values P and create several profit structures based on S and P. We proceed as follows:

1. Identify the OD-pairs for which the evaluation contribution increased the most. This increase can be caused by high passenger numbers or by a high increase in the perceived travel time. We refer to these OD-pairs as *relevant* OD-pairs in the remainder of this section. Let $\mathcal{O} = \{o_1, o_2, \ldots, o_\kappa\}$ be the set of origin-stations for the relevant OD-pairs.

2. We create $|\mathcal{O}| \cdot |P|$ new profit structures, one for each combination of penalty values from P at each station. We create the corresponding profit structure as follows: for each station $o_i \in \mathcal{O}$ and for each $p_j \in P$, we create a new profit structure that is based on S, with an additional intermediate shift penalty of value p_j that is assigned to all trains passing station o_i. Furthermore, all trains for which o_i is an origin or terminal station and which are *relevant* for the corresponding OD-pair identified in step 1, receive shift penalty $\max\{s(t), p_j\}$, where $s(t)$ is the regular shift penalty for train t in profit structure S.

   If $|\mathcal{O}| > 1$, we generate additional profit structures. In these new profit structures we apply the same principle as above, but now we apply the penalties to all *pairs* of stations. That means, for each $o_i, o_j \in \mathcal{O}$ and for each $p_m, p_n \in P$, apply (intermediate) shift penalty p_m to station o_i and penalty p_n to station o_j.

Given the set of updated profit structures Ψ, we again run LH, as outlined in Section 4.1. Each of these profit structures leads to a new timetable which we evaluate. If any of these timetables gives a better evaluation value, we stop the feedback process and finish with the best timetable generated using Ψ. Else, execute steps 1 and 2 again with the original profit structure as input, but now also identify OD-pairs as relevant for which the evaluation contribution increased the most in the best timetable generated using the profit structures in Ψ. The timetable that is the best after providing feedback is referred to as the *best after feedback* (BF).

We underline that the intermediate shift penalty is not adopted at every intermediate station where there are irregular departure headway times, but only at those that cause a significant increase in the evaluation value. Indeed, it would not be effective to penalise shifts at every station, since some changes are neededin order to get a feasible timetable. Therefore, we aim at penalizing the changes that have most impact on the evaluation value of the timetable. For the same reason, we do not use the intermediate shift penalty when LH is applied to the ideal timetable in the first round before the feedback process.

The rationale behind our feedback approach, is that adaption times have a strong influence on the evaluation value of a timetable. If LH causes a higher irregularity in the new timetable compared to the ideal timetable, the evaluation value is likely to increase. For this reason, we focus on the shift penalties in the feedback process. However, many alternative strategies to provide feedback in order to reduce the evaluation contributions of OD-pairs can be thought of, as we can update initial train profits, shift and stretch penalties, maximum shift and maximum stretch, as well as adding intermediate shift penalties. We experimented with different strategies on how to update profit structures before we identified this one which was successful on our three test instances and that is presented here.

5 Case Study

In this section we perform three case studies. First, we describe the three instances in Section 5.1. Next, in Section 5.2 we describe the parameters that are used in our approach. In Section 5.3, we describe and discuss the obtained results. Finally, in Section 5.4 we benchmark our iterative approach with solving the POT model as an integer program.

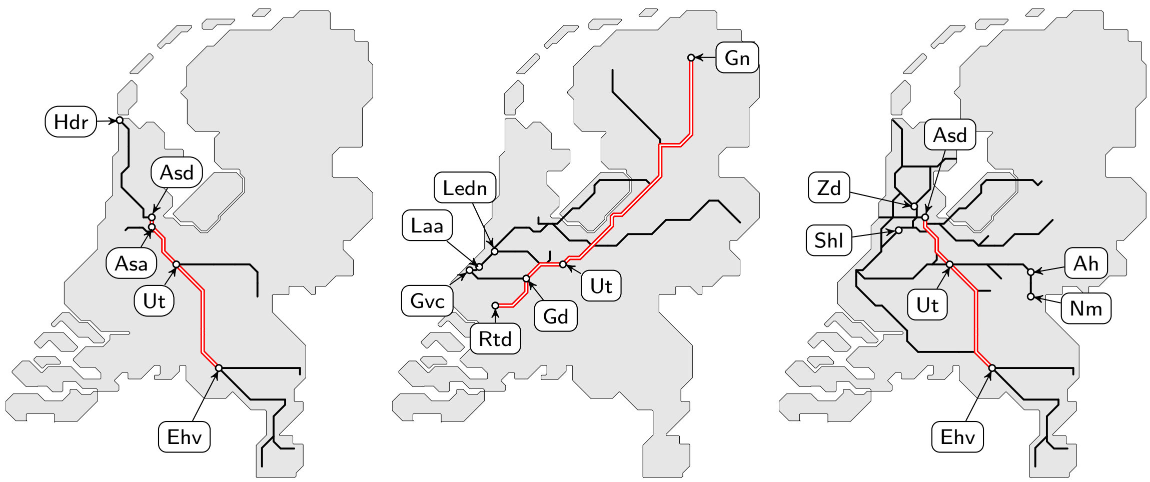
5.1 Instances

Here we describe the instances that we consider in more detail. Each instance is based on a central corridor. Figure 3 displays for each instance an overview of the network, the central corridor on which the instance is based is shown in red.

5.1.1 A2-corridor instance

The first instance we consider is the 'A2-corridor', which is a corridor between the stations Eindhoven (Ehv) and Amsterdam Central (Asd). The network contains 34 stations. Furthermore, five train lines are operated on this network with a frequency of two trains per hour in both directions, so there are 20 trains in total. All of them are Intercity-lines. The map of the corresponding network is shown in Figure 3a. We label five out of the 34 stations because they are used in the feedback of our approach. The stations Ehv and Asd are the ends of the central corridor, as is indicated by the red line.

In this instance, we consider 891 OD-pairs in total. The underlying event-activity network (as introduced in Section 3.2.2) contains 1344 events and 1460 drive and dwell activities. Note that these events do not only cover arrivals and departures at stations where the train stops, it also covers departure and arrival times of stations that the train passes, as also in these locations, consecutive trains need to satisfy the headway time of 3 minutes. To



(a) A2-corridor (b) Rotterdam-Groningen (c) Extended A2-corridor

Figure 3: Networks of the three instances considered

build the model for constructing the ideal timetable, 376 transfer activities are included to ensure the transfer possibilities. Note that they do not impose operational restrictions on the timetable as explained in Section 3.2.2. For a full mathematical programming formulation of POT, infrastructure constraints should be added. This leads to 2964 additional activities modelling safety distances.

5.1.2 Rotterdam-Groningen instance

The second instance covers part of the 2019 line plan of Netherlands Railways [NS]. It is centered on the line between Rotterdam (Rtd, in the South-West) and Groningen (Gn, in the North-East). This line is marked in red in Figure 3b. All lines that share a part of their route with the indicated line are added. Note that a consequence of this way of instance construction is that some, but not all lines operating on the network arcs indicated in black are included in our instance. The network contains 77 stations, of which 6 are labelled because they occur in our description of the results. In total we have 60 trains in the network.

The underlying event-activity network of this instance contains 1664 events and 1716 drive and dwell activities. The model for constructing the ideal timetable has 1402 additional transfer activities to enable all passenger routes. In order to deal with the restrictions modelling safety distances, 4004 additional activities are needed. There are 3810 OD-pairs. Note that, although there are many more trains compared to the A2-corridor instance, the number of events, as well as of drive and dwell activities, increased only slightly compared to the A2-corridor. The reason for this is that we now have a number of trains which only have a short route (and thus less events are needed per train line). The main increase is seen in the number of transfer activities, to generate all possible transfer routes, as well as in the number of activities modelling safety distances.

5.1.3 Extended A2-corridor instance

The third instance is an extension of the A2-corridor instance. The line plan is based on all train lines in the 2019 network of Netherlands Railways that share a part of their route with the corridor between Amsterdam Central (Asd) and Eindhoven (Ehv). This corridor is marked in red in Figure 3c, which also shows the remainder of the network, including the locations of several stations that we use in the discussion of the results later on. The total number of trains in this network is 88 and the network contains 140 stations.

This instance considers 11121 OD-pairs. The event-activity network contains 3160 events and 3308 drive and dwell activities. Furthermore, 3592 transfer activities are added to enable all passenger routes in the model to construct the ideal timetable. Adding the restrictions modelling safety distances leads to 8360 additional activities.

5.2 Parameters

This section describes the parameters that we use in our experiments. First, we describe instance parameters, as well as the parameters for the objective function. Next, we detail the profit structure that we use for LH.

To compute an ideal timetable, an integer programming problem has to be solved. The same has to be done for solving POT directly, as we do in Section 5.4. For this, we use a machine with an Intel Xeon Silver 4110 2.10Ghz processor with 96 GB of RAM installed. These mathematical programs are solved by Cplex 12.9.0 under default settings, using up to 15 parallel threads [IBM, 2019].

5.2.1 Instance & objective function parameters

In all our experiments we discretise time to minutes and use a period length of one hour, i.e., $T = 60$. In each of the instances, we take a headway time of three minutes into account between two trains leaving or entering a station. Furthermore, trains cannot overtake each other between two stations. When trains in opposite directions have to be separated in time when entering or leaving a station, we require this time to be at least one minute. If there are two tracks between stations, trains in the same direction will share one track, the other track is for the other direction. Finally, if there are four tracks available, two tracks are used per direction. The Intercity trains use one track and the local trains use the other, the same holds for the other direction.

The perceived travel time for the passengers consists of in-train time, transfer time, a transfer penalty, and the adaption time. In line with Polinder et al. [2020] and de Keizer et al. [2015], we set $\gamma_t = 20$, i.e., the transfer penalty equals 20 minutes. Secondly, we use $\gamma_s = 1$, i.e., transfer time weights as much as in-train time. This is done since we already have a transfer penalty. Finally, we take $\gamma_w = 3$, i.e., adaption time weights three times as much as in-train time.

If passengers no longer have a travel option when trains are cancelled, we add a penalty of value M to the average perceived travel time of these passengers, as explained in Section 3.2.3. In our experiments, we set this value M to be $24 \cdot T$, i.e., it equals a full day of travel time.

Finally, since the numbers in the OD-matrix are confidential, we scale all evaluation values. That is, we divide all evaluation numbers by the evaluation value of the ideal timetable, and multiply this by 100, so the evaluation value of the ideal timetable is indexed to 100. That means that an increase in the evaluation value by one unit means the evaluation value is 1% higher than that of the ideal timetable.

5.2.2 Chosen profit structure

As explained in Section 4.3.1, the Lagrangian heuristic requires the specification of a profit structure. Different profit structures can produce different timetables. As an initial profit structure, we define the train profits based on the train type (Intercity, local, etc.) and also on line frequencies. For the train type 'Intercity', we consider a base profit of 4000. For the train type 'local train', the base profit is reduced by 10% to 3600. For trains that partly operate as an Intercity and partly as a local train, the base profit is reduced by 5% (to 3800). Then, for each pair of consecutive stations that the train visits, we identify the number of trains that travel between the same pair of consecutive stations, and take the minimum m of these numbers along the train route. The train profit is computed as its base profit divided by m. As an example, if we consider a local train line, whose line frequency is two, and that is the only train line offering a service on some part of the network, then on this part of the network, there are only two trains and hence $m = 2$. Then the profit for the trains in this line is $(4000 \cdot 0.9)/2 = 1800$. If another line with frequency two is present as well on the considered part of the network, we have $m = 4$ and the profit is 900.

For the shift and stretch penalties, we consider equal values for all trains, and globally three alternative options: (1) shift penalty set to 20 and stretch penalty to 10; (2) both shift and stretch penalty set to 15; (3) shift penalty set to 10 and stretch penalty to 20. Namely, we assign more importance to the shift penalty in the first case, same importance to both changes in the second case, and more importance to the stretch penalty in the third case. Indeed, it is not known a priori whether a shift or a stretch is worse: it depends on the location where this happens and what the influence is on the regularity of trains in general. In addition, we want to explore a rather broad spectrum of profit structures, because it is not a priori known which changes in the timetable have the least negative effect on the evaluation of the timetable according to evaluation function (3.6).

As maximum shift and maximum stretch, we also consider the same values for all trains, and start by assigning value 5 to both of them: this means that each train can have its departure time from its origin station up to 5 minutes earlier or 5 minutes later, and a total stretch along its route of up to 5 minutes. Then, we also consider two other options, that increase the possibilities of scheduling trains: maximum shift set to 10 and maximum stretch to 5, and maximum shift set to 5 and maximum stretch to 10. Indeed, when the maximum shift and stretch are set to small values, it might not always be possible to schedule all trains, leading to low quality of the solutions. Overall, by combining the different shift/stretch penalties and the maximum shift/stretch, we thus have 9 different profit structures that lead to 9 timetables. The total number of iterations for each run of LH is set to 250 in our experiments.

In the feedback process, we set values for the intermediate shift penalties at some stations, as well as for the initial shift penalty of a train starting in such a station. We use values of 10, 20 and 30 for these penalties.

5.3 Results of the algorithm

We execute our timetabling approach presented in Section 4 on the three instances.

5.3.1 A2-corridor

Make an ideal timetable. Since obtaining an optimal solution is out of reach for this instance, we use a time limit of two hours. The resulting timetable has, as mentioned earlier, a normalized objective value of 100. All other values are reported in relation to this value. The lower bound that is proven by CPLEX is 97% of the objective value. Hence the remaining gap is 3%.

Figure 5a displays a time-space diagram, showing the ideal timetable between Hdr and Ut. Hdr is in the most northern part of the network and Ut is halfway the corridor (see Figure 3a. Time is shown on the horizontal axis, between 0 and 60, i.e., one cycle period is displayed. Space is shown on the vertical axis, where several stations are mentioned. The lines on the right of the figures display the number of tracks that are present. In the diagram itself, each colored line corresponds to a train and displays at what time a train visits a given location. The colors of the trains correspond to the colors in Figure 4, where the routes of the five train lines in this instance are displayed.

Even though no regularity restrictions are added to the model, we see in Figure 5a that the trains are spread over time rather regularly in this network. This is due to the inclusion of the adaption time in the objective function. Adaption time is low, if departure times of

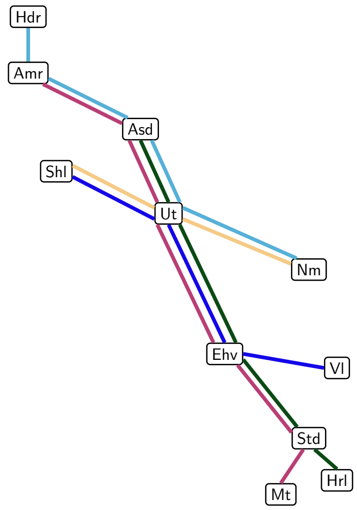


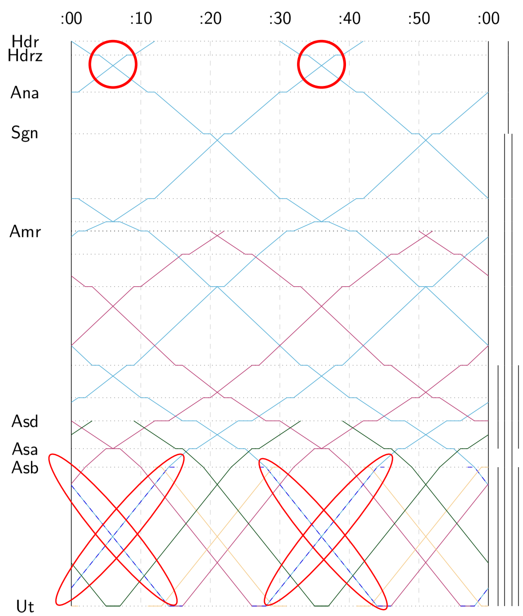
Figure 4: Overview of lines in A2-corridor (taken from Polinder et al. [2020])

routes for an OD-pair are equally spaced in time. See discussion in Polinder et al. [2020]. However, the ideal timetable on the A2-corridor does not satisfy the headway restrictions. There are two conflicts, which are indicated by red circles in Figure 5a. At one location, two trains are scheduled at the exact same time: the light blue and dark blue trains are scheduled at the exact same time between Ut and Asb. At another location two trains are scheduled to cross each other in a single track area between Hdr and Sgn, where trains can only pass each other at stations.

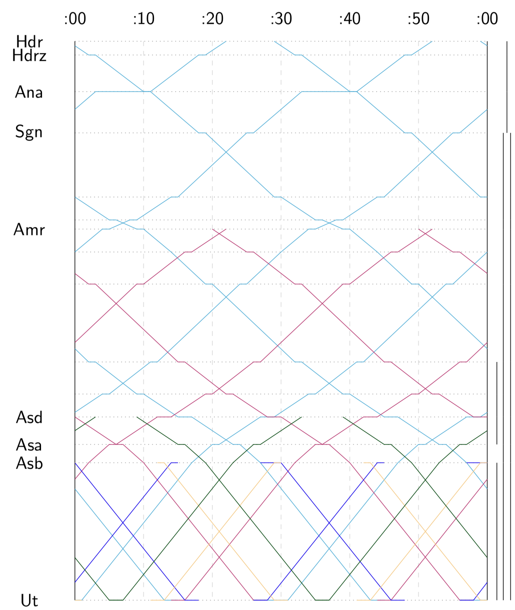
Make a feasible timetable. In order to make a feasible timetable, we run LH with the nine different parameter sets specified in Section 5.2.2 for the profit, shift and stretch penalties and bounds. This leads to nine feasible timetables, all satisfying the headway restrictions. Note that although LH allows to cancel trains, for all nine chosen parameter sets, all trains are scheduled. The best found timetable in this step, BPL, has an evaluation value of 100.18, i.e., the evaluation value increased by 0.18% with respect to the ideal timetable. For this BPL timetable, the time space diagram is shown in Figure 5b. There it can clearly be seen that the conflicts are resolved. Trains that crossed on a single track area are now stretched such that they pass each other at a station. Secondly, the trains that were scheduled at the same time are now moved away from each other.

Evaluate & update profit structure. In order to provide feedback, we follow the approach stated in Section 4.4.

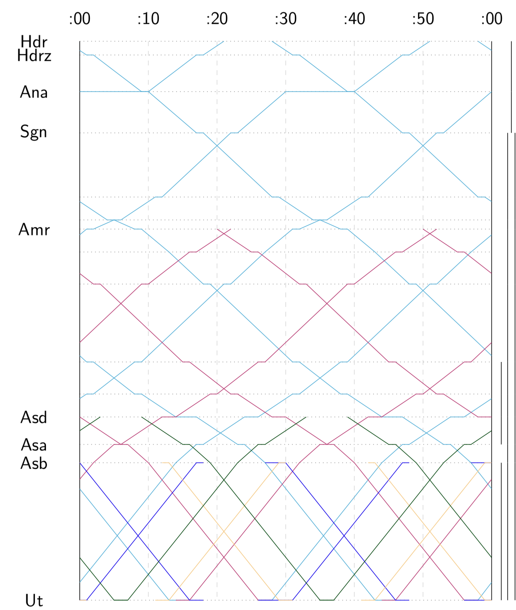
Identify OD-pairs. The first step is to identify the OD-pairs for which the evaluation contribution worsened the most. In order to do so, and to investigate the differences between



(a) Ideal timetable



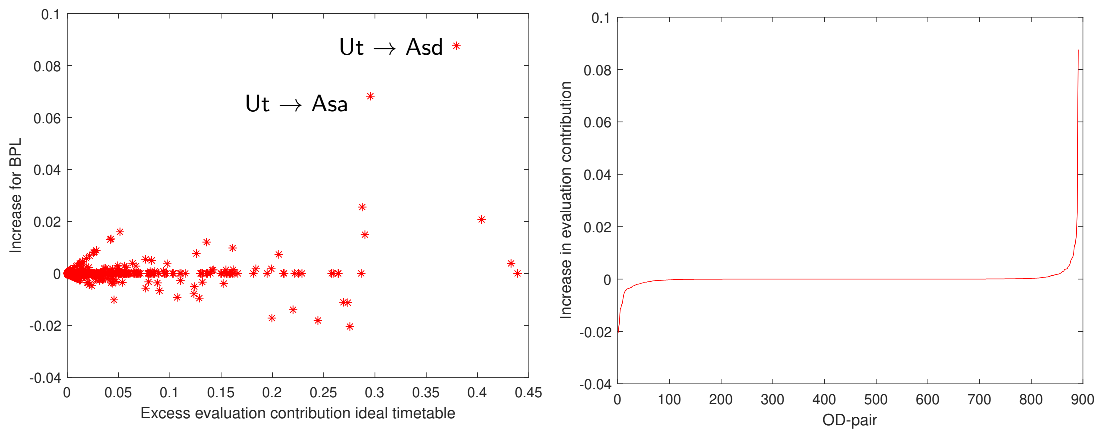
(b) BPL-timetable



(c) BF-timetable

Figure 5: Time space diagrams for A2 network

the ideal timetable π and the best found feasible timetable BPL, we compare the timetables with respect to the excess evaluation contribution (for the definition, see Section 3.3) of each individual OD-pair in Figure 6.



(a) Evaluation contribution comparison

(b) Evaluation contribution increase per OD-pair.

Figure 6: A2-corridor: ideal timetable vs BPL-timetable.

In Figure 6a, we see the difference in excess evaluation contribution for individual OD-pairs between timetables π and BPL. Each OD-pair is represented by a star, the stars are sorted from left to right according to the excess evaluation contribution of the corresponding OD-pair in the ideal timetable. As can be seen, for many OD-pairs the excess evaluation contribution is small in the ideal timetable, for only a few OD-pairs it is large. In this instance, the latter correspond to OD-pairs which have a high demand and a small irregularity in their departure pattern: recall that the evaluation contribution is weighted by the number of passengers, i.e., a large increase for only a few passengers can count less than a small increase for many passengers. The vertical coordinate indicates the difference in evaluation value between π and BPL. In particular, if a star lies above 0, the evaluation contribution of the OD-pair has increased after applying LH. But there are also some OD-pairs which have a lower evaluation contribution after applying LH, these can be found below 0. Two OD-pairs are labelled in the figure, these are the OD-pairs with the highest increase in perceived travel time that clearly stand out and on which we base the feedback. As can be seen, the OD-pair Ut-Asd has an excess evaluation contribution of 0.38 in the ideal timetable, and that evaluation contribution now increased to 0.47 in the BPL-timetable, due to a more irregular departure pattern at Ut.

In Figure 6b we see a different visualization of the differences between π and BPL with respect to the evaluation contribution. All 891 OD-pairs are shown on the horizontal axis sorted by their corresponding increase in evaluation contribution. As can be seen well in this figure, there are many OD-pairs for which the evaluation contribution hardly changed, only for a minority there are major changes. Hence, also in this figure it can be seen that the BPL timetable is very close to the ideal one, and only few OD-pairs were subject to an increase in perceived travel time.

Update profit structure. Two OD-pairs are identified as relevant: Ut-Asd and Ut-Asa. These two OD-pairs have the same set of travel options, as the trains from Ut to Asd first pass Asa (see Figure 3a). In fact, Asa is the only station where the trains from Ut to Asd stop. Passengers on these OD-pairs can choose from six different trains. Thus, to minimise adaption time upon departure in Ut, the headway times between consecutive trains would be 10 minutes, in that case the trains would be perfectly spread over time. In the ideal timetable we have headway times of 12 minutes (2 times) and 9 minutes (4 times), as is visible in Figure 5a. That means that in the ideal timetable the trains are not spread equally over time, and this causes the excess evaluation contribution of the relevant OD-pairs in the ideal timetable to be relatively large, in particular because these adaption times are weighted with the (high) passenger numbers. In the BPL-timetable, the departure pattern in Ut becomes even less regular. The headway times now are 15 minutes (twice), 9 minutes (twice) and 6 minutes (twice), see Figure 5b.

Based on these observations, we add intermediate shift penalties at Ut as described in Section 4.4, to improve the timetable for Ut-Asd and Ut-Asa. We start with the profit structure leading to the BPL timetable, and generate three new profit structures using values of 10, 20 and 30 as penalty values. Ut is intermediate station to 20 trains where these are added as intermediate shift penalties. No trains start their journey in Ut so no regular shift penalties have to be updated. We run LH with the three new profit structures and evaluate the resulting timetables. Unfortunately, in this case, none of these timetables give an overall improvement.

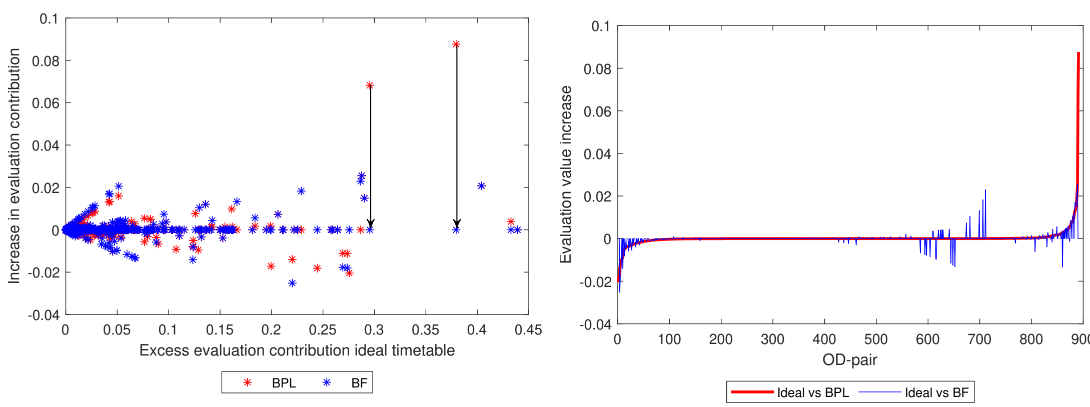
Identify OD-pairs. In all of the three timetables obtained using the updated profit structures, we see that the OD-pairs Ut-Asd and Ut-Asa are not improved, in fact, for the OD-pairs in the other direction (Asd-Ut and Asa-Ut) the situation becomes worse compared to the BPL timetable. The regularity at Asd (and in line with that also at Asa) is lost. Therefore, we identify these two OD-pairs as the new OD-pairs to provide additional feedback.

Update profit structure. As described in Section 4.4 we modify the initial profit structure that was used to find the BPL timetable and add shift penalty values at Ut *and* Asd. Asd is intermediate station to 8 trains, and for 2 trains the start station. This leads to 12 new profit structures: 3 for adding the three different penalty values 10, 20, and 30 at

Asd and 9 for the combinations of Ut and Asd. Note that a shift penalty at Asd can also improve the timetable at Ut, since these two stations are close to each other and there is only one stop in between.

Evaluation. After evaluating the 12 additional timetables, we find that the best evaluation value among the 12 created timetables now is 100.10, which is an improvement with respect to the BPL timetable. The time space diagram for this timetable is shown in Figure 5c. Note the improved regularity between Ut and Asd. The best timetable is found in the second feedback-iteration, and we refer to it as the *Best after feedback* (BF).

We now extend Figure 6 by adding blue stars, showing the evaluation contributions in the BF-timetable. Note that the OD-pairs Ut-Asa and Ut-Asd improve in the second iteration, as the blue stars representing these OD-pairs are now on the horizontal axis (the indicated arrows show the link between the two red and the corresponding blue stars). However, there are also some OD-pairs for which the perceived travel time increases in comparison to the BPL timetable. This is best seen in Figure 7b. Here, the same red line as in Figure 6b is shown. Now, the excess evaluation contributions for the OD-pairs in the timetable after feedback are added in blue. It is visible that some OD-pairs for who the evaluation contribution did not change in the BPL timetable are now changed: for some the perceived travel time decreases, but for others it increases.



(a) Evaluation contribution comparison

(b) Evaluation contribution increases per OD-pair.

Figure 7: A2-corridor: Timetable comparisons after feedback.

Summary. A summary of the progress of our approach on the A2-corridor is given in Table 1. The table displays the evaluation values for the best timetables found in the steps of the algorithm. The last row of the table shows the lower bound that is found by CPLEX when solving the integer programming model for finding an ideal timetable, which is, of course, also a lower bound on the objective value of a feasible timetable.

Timetable	Evaluation value
Ideal	100
Best Pure Lagrangian	100.18
Feedback step 1 (FB-1)	100.23
Feedback step 2 (FB-2)	100.10
Lower bound	97.00

Table 1: Evaluation values for A2-corridor

A visual summary of our approach is displayed in Figure 8. Here, the evaluation values of all computed timetables are displayed. The horizontal axis displays the step in the algorithm. The vertical axis shows the evaluation value of the timetable. First, the evaluation value of the ideal timetable is shown at the bottom left. Then, the blue lines and dots link this evaluation value to the evaluation values of the nine timetables computed by LH. Next, the red lines and dots display the values of the timetables computed during the feedback process. The evaluation values of the three timetables computed in the first and second feedback step (FB-1 and FB-2) are shown, with lines linking this evaluation value to the evaluation value of the BPL timetable. It is clearly visible that the evaluation value of the BF timetable is lower than that of the BPL timetable.

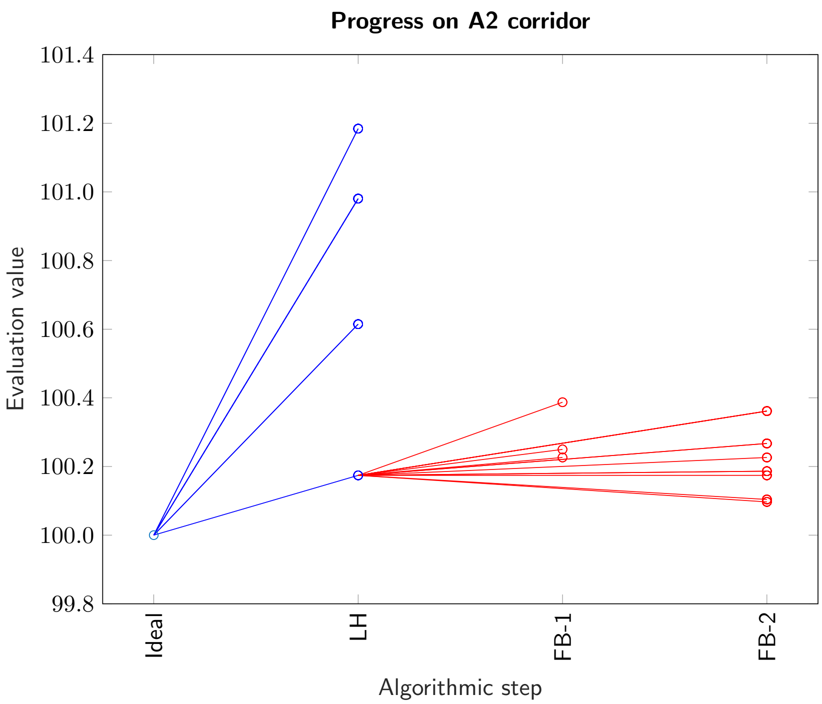


Figure 8: Overview of the progress in A2-corridor

5.3.2 Rotterdam-Groningen instance

Make an ideal timetable. Since this instance is more challenging than the previous one, we set a time limit of 4 hours to solve the model for constructing the ideal timetable. The best solution that we find within the time limit has a normalized objective value 100. The lower bound that is proven by CPLEX is 92.69% of the objective value, hence the remaining gap is 7.31%.

The ideal timetable is shown in terms of time-space diagrams in Figure 9 for two corridors that play a role in the description of the results: Rotterdam (Rtd) to Utrecht (Ut) and Leiden Central (Ledn) to The Hague Central (Gvc). See Figure 3b for the location of the stations.

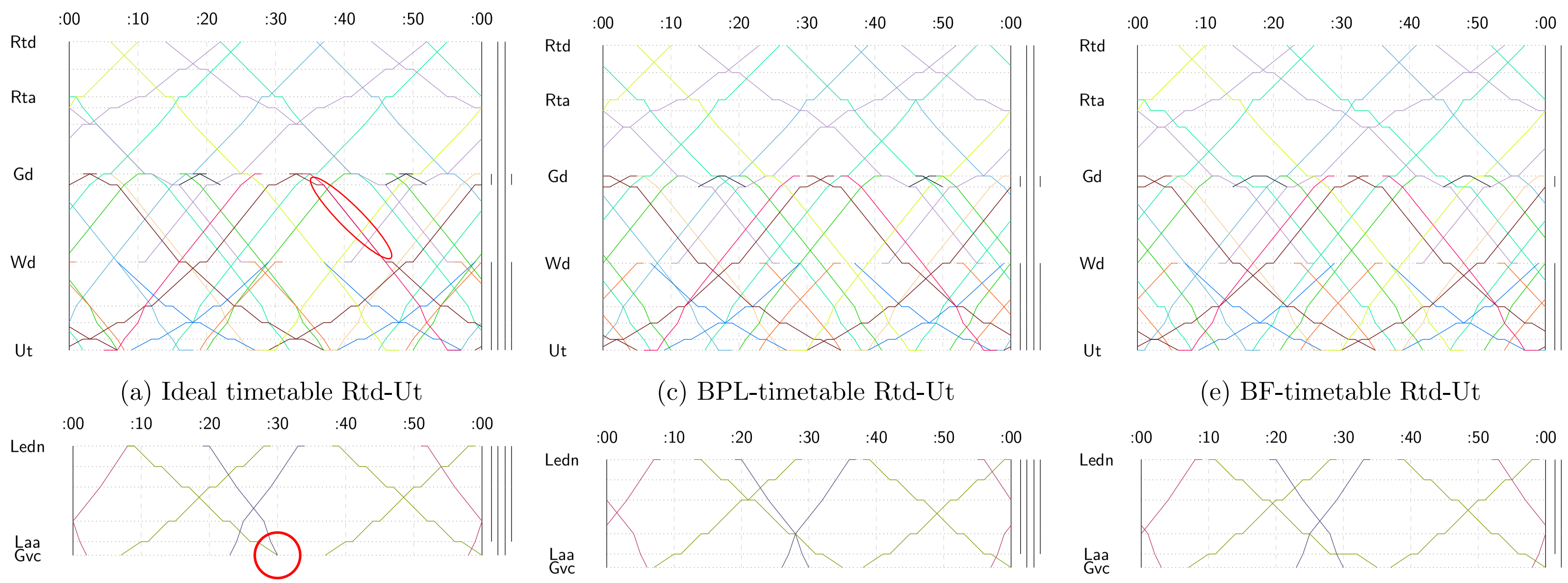
The ideal timetable does not satisfy the headway restrictions on the two mentioned corridors. For example, between stations Gd and Wd, two trains are scheduled at the exact same time on the same track and hence do not satisfy the headway restrictions (see the circled area in Figure 9a). Also on the other corridor violations of the headway restrictions occur. For example, two trains from the same direction arrive in Gvc at the same time, while there is only one track available for them, so the headway restriction upon arrival of two trains is not satisfied (see Figure 9b).

Make a feasible timetable. In order to find a timetable that is feasible with respect to current infrastructure, we run LH with the standard parameters. In seven out of the nine resulting timetables, all trains are scheduled. In two of them, two trains are cancelled. However, also in these timetables there are still travel options for all passengers. The best evaluated timetable has an evaluation value of 100.59 and has all trains scheduled. The time space diagrams for this BPL-timetable on the two aforementioned corridors are displayed in Figures 9c and 9d, which clearly show that the conflicts are resolved.

Evaluate & update profit structure.

Identify OD-pairs. In order to investigate the differences between the ideal timetable π and the BPL-timetable, we make the same plots as for the previous instance, showing the differences in evaluation contribution per OD-pair. Figure 10a displays each OD-pair as a star, where the excess evaluation contribution to the ideal timetable is the horizontal coordinate and the increase in evaluation contribution is the vertical coordinate. These increases in the evaluation contribution are summarised in Figure 10b, where all 3810 OD-pairs are shown on the horizontal axis, sorted by their corresponding increase in evaluation contribution. Note that for many OD-pairs the evaluation contribution does not change.

In the BPL-timetable, there are three OD-pairs that stand out, as is indicated in Figure 10a: Ledn-Laa and Ledn-Gvc are two OD-pairs that have very similar routes, they have



(a) Ideal timetable Rtd-Ut

(c) BPL-timetable Rtd-Ut

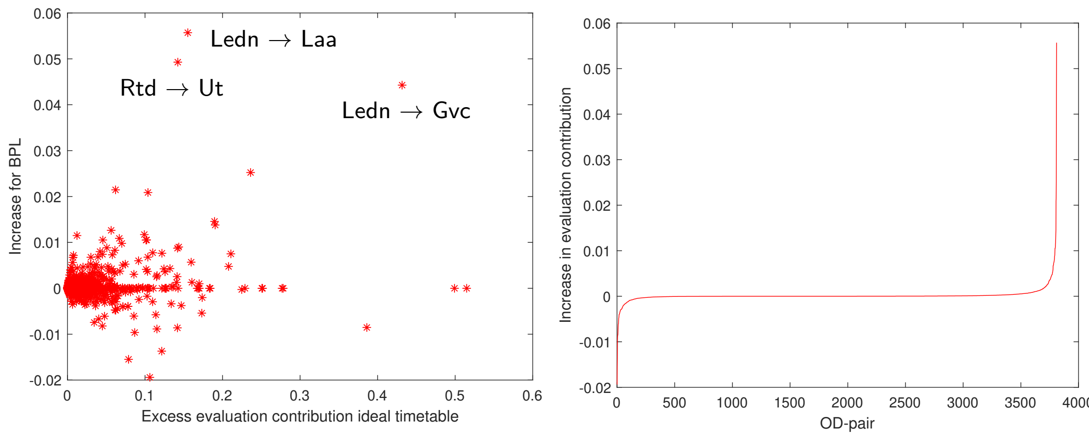
(e) BF-timetable Rtd-Ut

(b) Ideal timetable Ledn-Gvc

(d) BPL-timetable Ledn-Gvc

(f) BF-timetable Ledn-Gvc

Figure 9: Time space diagrams for the Rotterdam-Groningen instance



(a) Evaluation contribution comparison

(b) Evaluation increases per OD-pair.

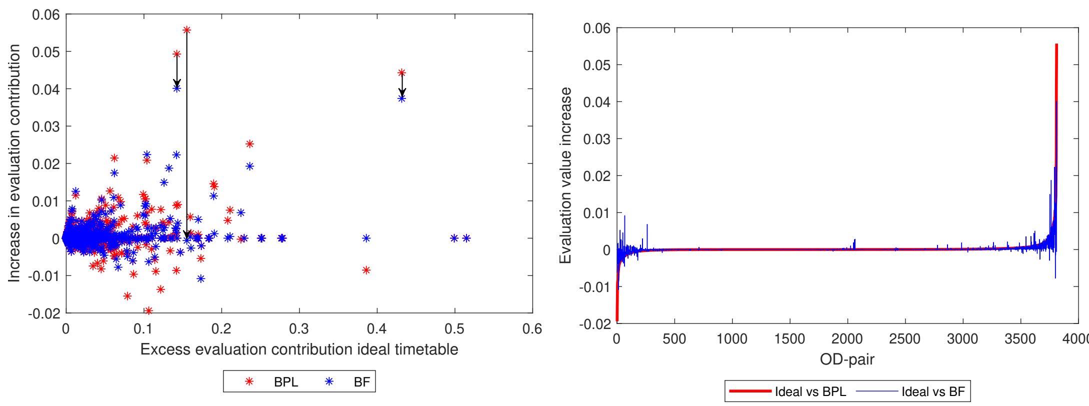
Figure 10: Rotterdam-Groningen: Ideal timetable vs BPL-timetable.

the same origin and their destinations are close to each other, see also Figure 3b. The corresponding excess evaluation contributions increased by +0.056 and +0.044 respectively with respect to the ideal timetable. The third OD-pair for which the evaluation value increased significantly is Rtd-Ut, its excess evaluation contribution increased by +0.050 with respect to the ideal timetable. This OD-pair is located on a different part of the network than the other two OD-pairs.

Update profit structure. The origin stations for the aforementioned OD-pairs are Ledn and Rtd. At both of these stations, the regularity of departure times has worsened when shifting from the ideal to the BPL-timetable. Therefore, we make new profit structures, based on the one leading to the BPL-timetable, where we add the intermediate shift penalties of values 10, 20 and 30 to Ledn and Rtd. Ledn is intermediate station for 8 trains and no relevant trains start their journey in Ledn. Rtd is for no train an intermediate station, instead, 6 trains start their journey there so we update the regular shift penalty. This leads to 15 new profit structures: 3 with the penalty only at Ledn, 3 with the penalty only at Rtd, and 9 for all combinations.

Evaluation. We run LH on these new profit structures and evaluate the resulting timetables. We find the best timetable to have an evaluation value of 100.55. This timetable is referred to as the *best after feedback* (BF).

Like for the previous instance, we extend Figure 10 by plotting the evaluation contributions of the OD-pairs in the BF-timetable, in order to get insight into the differences with respect to the BPL-timetable. The result is displayed in Figure 11. Figure 11a shows for each OD-pair the increases in evaluation contribution with respect to the ideal timetable. The red stars correspond to the BPL-timetable, and the newly added red blue stars correspond to the BF-timetable. To increase visibility, we connected the red and blue stars which correspond to the relevant OD-pairs. As can be seen in this figure, the evaluation contribution for the three OD-pairs is improved. The increase in evaluation contribution with respect to the ideal timetable of Ledn-Laa reduces to -0.001, i.e., the timetable for this OD-pair is better than the ideal timetable. The increase in evaluation contribution for Ledn-Gvc now reduces to $+0.037$ and Rtd-Ut reduces to $+0.040$.



(a) Evaluation value comparison

(b) Evaluation increases per OD-pair.

Figure 11: Rotterdam-Groningen: Timetable comparisons after feedback.

To understand how the evaluation contribution of OD-pair Ledn-Laa is improved in the feedback step, we examine the time-space diagrams in Figures 9b–9f. In the ideal timetable for our instance, passengers can either take a local train directly from Ledn to Laa. The alternative is to take an Intercity train that does not stop at Laa, but travels to Gvc, where passengers can transfer to a local train back to Laa.

In the ideal timetable, the Intercity train from Ledn arrives in Gvc at :02, and passengers have a 5 minute transfer connection to a local train back to Laa. In the BPL-timetable though, the Intercity train comes in four minutes later and arrives in Gvc at :06, while the local train back to Laa is not shifted in time. A transfer time of one minute is too short and hence passengers have to wait half an hour, thus leading to a very high evaluation value contribution.

In the feedback, trains get an intermediate shift penalty at Ledn. This causes the Intercity train still to arrive in Gvc later than in the ideal timetable, but earlier than in the BPL-timetable: it now arrives at :03. Passengers can again make the connection to the local train and we are in a similar situation as in the ideal timetable.

This illustrates how the additional penalties lead to a different timetable and how feedback can be used to improve the timetable that is found.

Note that the 2019 NS line plan [NS] contains more trains between Ledn and Laa that we did not include in our instance (see Section 5.1.2). The observed effect is thus particular to our instance and would not be observed in the actual Dutch network.

Summary. Table 2 summarizes the results of our approach on the Rotterdam-Groningen instance. The evaluation values are displayed for each step in our algorithm.

Timetable	Evaluation value
Ideal	100
Best Pure Lagrangian (BPL)	100.59
Feedback step 1 (FB-1)	100.55
Lower bound	92.69

Table 2: Evaluation values for Rotterdam-Groningen instance

Also for this instance, we summarize the evaluation values found in the different steps of the algorithm in Figure 12. On the left, we see the evaluation value of the ideal timetable. Right of it, we see the evaluation values of the nine timetables found by the Lagrangian heuristic. Two lines are drawn with a dash-dotted line, these correspond to the timetables where some trains are cancelled. We observe that cancelling trains does not automatically lead to bad timetables, as long as all OD-pairs can still travel. This is because the cancelling of trains gives us more freedom to schedule other trains.

5.3.3 Extended A2-corridor

Make an ideal timetable. Also for this instance, we use a time limit of four hours to solve the model for constructing the ideal timetable. Normalizing the best timetable to an evaluation value of 100, the best lower bound is 93.00, i.e., the remaining gap is 7.0%.

Time-space diagrams displaying the ideal timetable are shown in Figure 13 for three corridors in the network: Arnhem (Ah) to Nijmegen (Nm), Amsterdam Central (Asd) to Schiphol (Shl) and Zaandam (Zd) to Utrecht (Ut). See Figure 3c for the location of these stations. In this dense network, there are numerous conflicts between trains that have to be resolved in the next step.

Make a feasible timetable. Using the same initial profit structures as in the earlier cases, in the first step of LH nine feasible timetables are computed. In three of these timetables, all trains are scheduled. In the other six, trains are cancelled in a way that causes some passengers not to have a travel option any longer. This is highly undesirable and highly penalized in the evaluation function (see Section 3.2.3). This leads to these six timetables having a bad

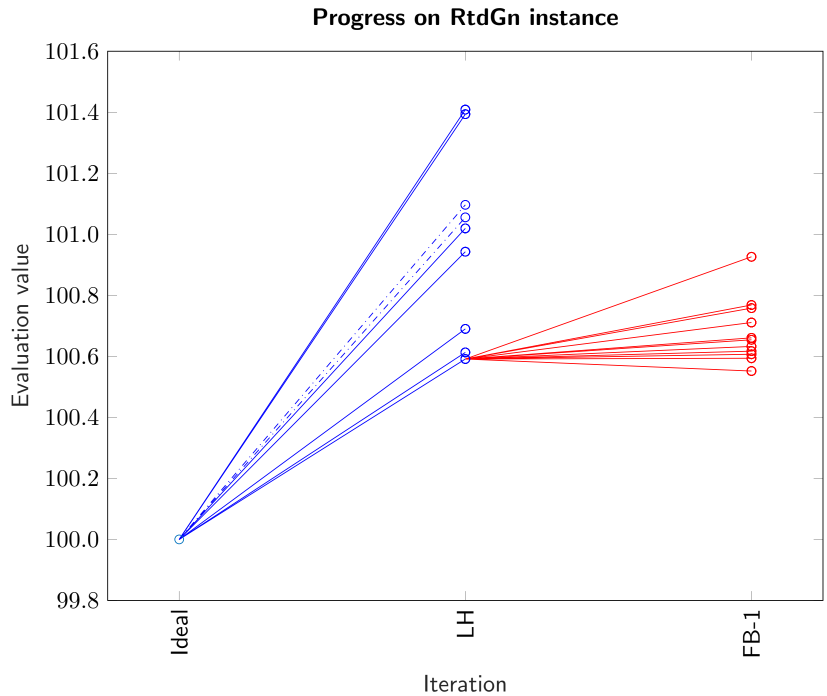


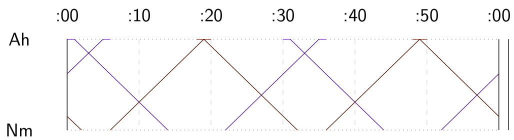
Figure 12: Overview of the progress in Rotterdam-Groningen instance

evaluation value. Consequently, the feasible timetable with the best evaluation value is one in which all trains are scheduled. Its evaluation value is 101.51. The corresponding time-space diagrams are shown in Figure 13.

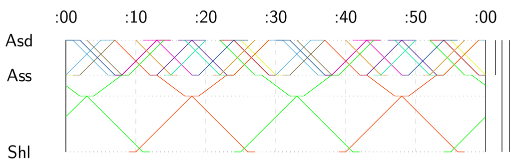
Evaluate & update profit structure.

Identify OD-pairs. As a first step, we identify the OD-pairs for which the shift from the ideal timetable to the BPL-timetable caused a high increase in the evaluation contribution. The changes in the evaluation contribution for each OD-pair are pictured in Figure 14, in a similar way as in the previous two cases. We observe that for many OD-pairs there are only few or little changes (see also Figure 14b). However, there are a few OD-pairs for which the evaluation contribution increases significantly as is visible in Figure 14a. The highest increase is visible for the the OD-pair Ah-Nm. Its evaluation contribution increases by +0.062. Besides Ah-Nm, Asd-Ut (+0.045), Asd-Zd (+0.039) and Asd-Shl (+0.038) account for the largest increases in the evaluation value.

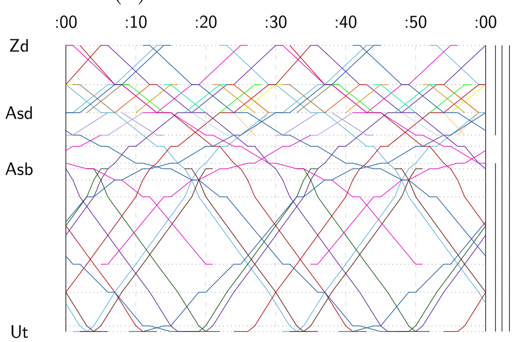
And indeed, inspecting the timetable differences (see the supporting time-space diagrams in Figure 13), we see that departure and arrival patterns have changed at Asd, trains now sometimes depart in a less regular pattern. Also the pattern of departures at Ah is less regular in the BPL-timetable than in the ideal timetable (compare also Figures 13a and 13d).



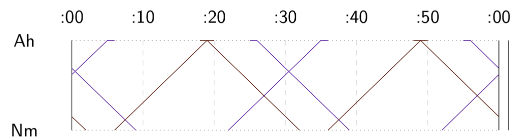
(a) Ideal timetable Ah-Nm



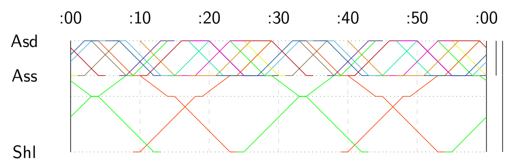
(b) Ideal timetable Asd-Shl



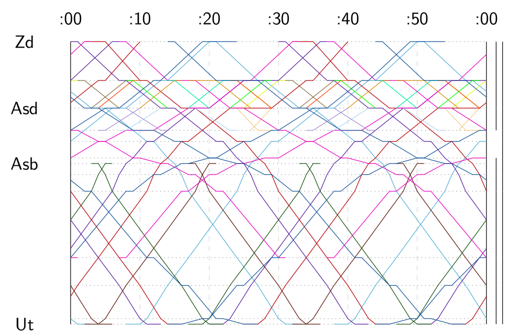
(c) Ideal timetable Zd-Ut



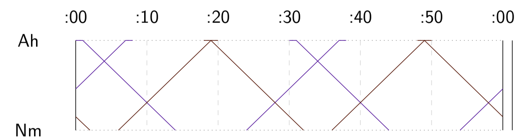
(d) BPL-timetable Ah-Nm



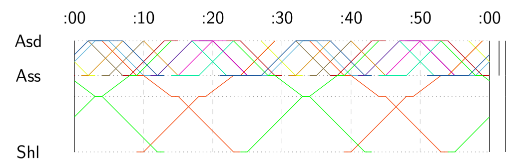
(e) BPL-timetable Asd-Shl



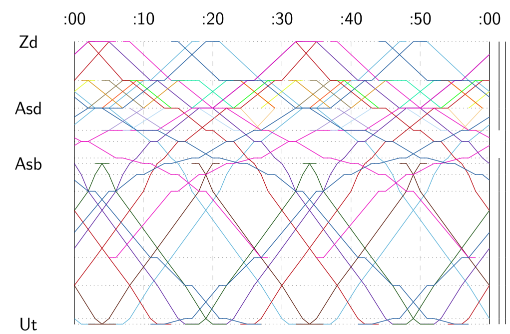
(f) BPL-timetable Zd-Ut



(g) BF-timetable Ah-Nm

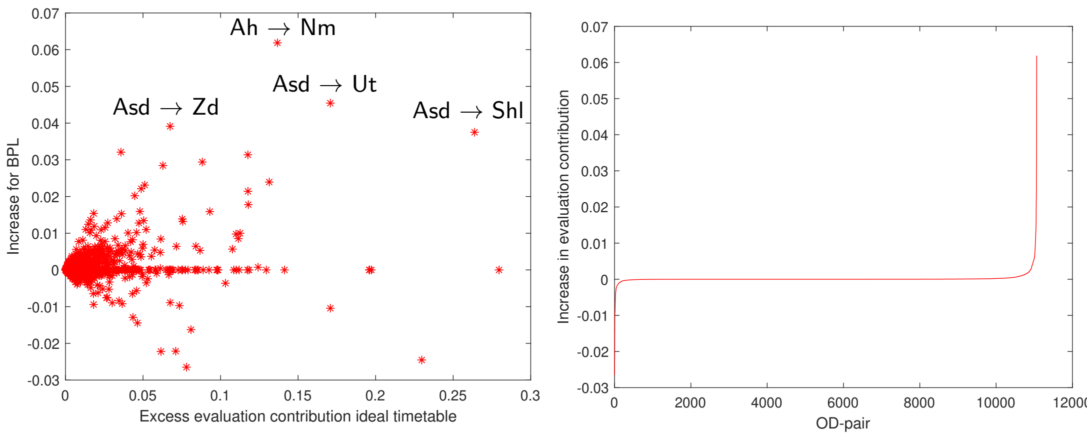


(h) BF-timetable Asd-Shl



(i) BF-timetable Zd-Ut

Figure 13: Time space diagrams for the extended A2-corridor instance



(a) Evaluation contribution comparison

(b) Evaluation increases per OD-pair.

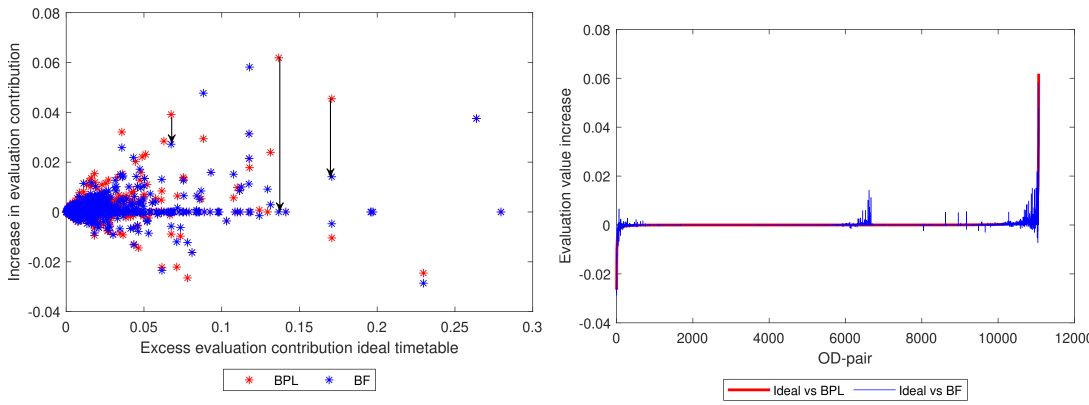
Figure 14: Extended A2-corridor: Ideal timetable vs BPL-timetable.

Update profit structure. As prescribed in our approach from Section 4.4, we update the profit structure for trains at the stations Ah and Asd with penalty 10, 20, or 30, leading two 15 new profit structures (3 for the shift penalties at Asd, 3 for the shift penalties at Ah, and 9 for the combinations).

Evaluation. We run LH with the new profit structures and obtain 15 new timetables. When evaluating these solutions, we find an improved timetable with an evaluation value of 101.28, i.e., a reduction of 0.23 with respect to the BPL-timetable. This new timetable is referred to as the BF-timetable.

Figure 15 displays the new evaluation contributions, both for the BPL-timetable and BF-timetable. The red stars correspond again to the increase in evaluation contribution when comparing the ideal timetable with the BPL-timetable. The blue stars show the same result when comparing the ideal timetable with the BF-timetable. The OD-pairs that had a high increase in evaluation contribution now improved significantly: the increase in evaluation contribution for Ah-Nm reduced from +0.062 to 0 (see also the time space diagram in Figure 13g). For Asd-Ut, the increase in evaluation contribution reduced from +0.045 to +0.014 and for Asd-Zd it reduced from +0.039 to +0.027. The increase for Asd-Shl remained the same. The improvements for the first three mentioned OD-pairs are shown by means of arrows in Figure 15a.

Although the overall evaluation value of the new timetable improved and the contributions of the aforementioned three OD-pairs improved as well, now other OD-pairs have a higher evaluation contribution, as already indicated in Figure 15b. An improvement for some OD-pairs can indeed imply a worsening for others. In the BF-timetable, there is an OD-pair



(a) Evaluation value comparison

(b) Evaluation value increases per OD-pair.

Figure 15: Extended A2-corridor: Timetable comparisons after feedback.

(Hlm-Asd) which now has an excess contribution which is similar to that of Ah-Nm in the BPL-timetable.

Summary. Table 3 summarizes the main results of our approach on the extended A2-corridor instance in terms of evaluation values for each of the steps.

Timetable	Evaluation value
Ideal	100
Best Pure Lagrangian (BPL)	101.51
Feedback step 1 (FB-1)	101.28
Lower bound	93.00

Table 3: Evaluation values for extended A2-corridor instance

A visual summary of the computations is shown in Figure 16. For each computed timetable, the evaluation value is plotted. On the left the evaluation value of the ideal timetable is shown. Next, the results of the timetables after running LH are indicated by the blue dots and lines. If in some timetable not all trains are scheduled, this is indicated by the dash-dotted line. As can be seen, many timetables have a bad evaluation value, they are not even visible in the figure. The reason for this is that in these timetables not all passengers have a travel option. In red, the evaluation values of the timetables after feedback are shown. Again, we use dash-dotted lines for the timetables where not all trains are scheduled.

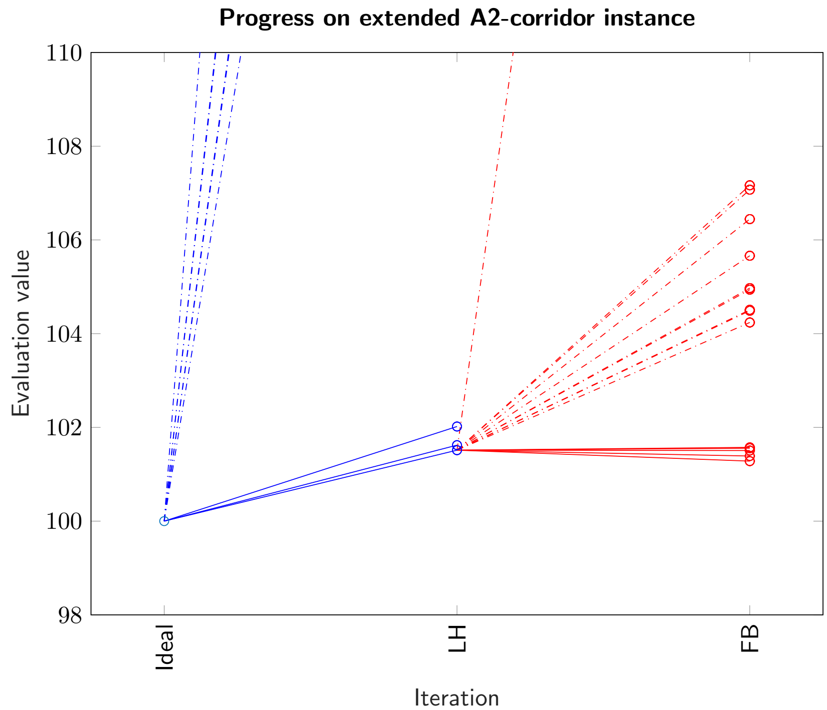


Figure 16: Overview of the progress in the extended A2-corridor instance

5.4 Comparison to a benchmark approach

In order to further evaluate our approach to solving POT, we compare it to the benchmark approach of modeling POT as a mixed-integer linear program. To make this comparison, we model POT as a mixed-integer linear program as described in Polinder et al. [2020], but additionally include infrastructure capacity constraints in the there-described model for strategic timetabling. We then solve it with the approach described in Polinder et al. [2020]. Note that the extended mixed-integer programming approach from Polinder et al. [2020] does not allow to cancel trains, as does our model. However, since there are no trains cancelled in any of the best timetables we found in the three considered instances, a fair comparison of results is possible. The results are displayed in Table 4.

In this table, for each instance, the result of several approaches to find a feasible timetable are shown. The first result is the best timetable that is obtained after computing an ideal timetable, and then running LH. The table reports the evaluation value of the timetable in the third column. The fourth column shows the time it took to compute this timetable. Where the time is split up in two parts, the first number shows the time spent on computing the ideal timetable, the other number shows the time spent on LH and feedback. Note that the time for inspection of the timetable and adjustment of the profit structure in the feedback loop is not included, because this is a manual process. The second result that is shown for each

Instance	Approach	Evaluation value	Time (hours)
A2	Ideal + LH	100.18	$2 + 0.03$
	Ideal + LH + FB	100.10	$2 + 0.11$
	POT		
	- After 2.11 hours	105.80	2.11
	- After 8 hours	104.88	8
	Lower bound CPLEX	97.09	
Rotterdam Groningen	Ideal + LH	100.59	$4 + 0.06$
	Ideal + LH + FB	100.55	$4 + 0.18$
	POT		
	- After 4.18 hours	105.64	4.18
	- After 16 hours	103.69	16
	Lower bound CPLEX	92.72	
Extended A2	Ideal + LH	101.51	$4 + 0.14$
	Ideal + LH + FB	101.28	$4 + 0.49$
	POT		
	- After 4.49 hours	-	4.49
	- After 16 hours	-	16
	Lower bound CPLEX	93.00	

Table 4: Benchmark results

instance is the evaluation value of the best timetable found after applying feedback. Third, the value of the best timetable after solving the integer programming formulation for POT including infrastructure capacity restrictions is shown. Next to the value that is obtained when reaching the time limit, we also show the evaluation value of the best timetable found in the time it took the iterative approach to compute the solutions listed. That means, for the A2-corridor instance, computing an ideal timetable, running LH and including feedback took 2.11 hours. In the same time, the full POT model found a solution of value 105.80. Finally, we mention the lower bound as computed by CPLEX when solving the POT model until the time limit is reached. Note that the CPLEX lower bounds are stronger than those mentioned in Section 5.3 where we report lower bounds on the MILP model for finding the *ideal* timetable, because the POT model is more restrictive and therefore, combined with a longer computation time, a stronger lower bound is more likely.

We observe that our approach is able to find better solutions in less time, even when no feedback is included. In particular, for the extended A2-corridor instance, we were not able to find any feasible timetable within 16 hours using the MILP formulation for POT, while the approach of this paper generates a reasonably good one within a bit more than two hours.

6 Conclusion and further research

In this paper, we proposed an approach to solve the tactical timetabling problem. Hereby we specifically focused on the quality of the timetable for the passengers.

In order to find a feasible passenger-oriented timetable for challenging real world instances, for which the timetabling model itself already is challenging, we used variants of two existing approaches. These two approaches are combined into an algorithmic framework. First, an ideal timetable is computed, thereby neglecting infrastructure related restrictions. Next, through a Lagrangian heuristic, this timetable is modified to obtain a feasible timetable with respect to infrastructure. A feedback mechanism is used to improve the found solutions.

We showed that for real-life instances, based on the network operated by Netherlands Railway, we can obtain satisfying results. Furthermore, we show that the provided feedback indeed leads to (overall) better timetables.

Interesting further research would include the further automatisation of the feedback procedure. Although this procedure is formalized in Section 4.4, it can still require manual inspection of the results in order to find a good feedback option. Furthermore, it would be interesting to investigate effects of including station capacity in our models.

Further research is needed to close the optimality gaps found in our models.

Acknowledgements. We thank Erasmus Trustfonds for supporting this work. Furthermore, we thank Gábor Maróti for providing several figures used in this paper.

References

A. Ait-Ali, P. O. Lindberg, J. Eliasson, J.-E. Nilsson, and A. Peterson. A disaggregate bundle method for train timetabling problems. *Journal of Rail Transport Planning & Management*, page in press, 2020.

E. Barrena, D. Canca, L. C. Coelho, and G. Laporte. Single-line rail rapid transit timetabling under dynamic passenger demand. *Transportation Research Part B: Methodological*, 70: 134–150, 2014a. ISSN 0191-2615. doi: https://doi.org/10.1016/j.trb.2014.08.013.

E. Barrena, D. Canca, L. C. Coelho, and G. Laporte. Exact formulations and algorithm for the train timetabling problem with dynamic demand. *Computers & Operations Research*, 44:66–74, 2014b. ISSN 0305-0548. doi: https://doi.org/10.1016/j.cor.2013.11.003.

G. Belvaux, N. Boissin, A. Sutter, and L. A. Wolsey. Optimal placement of add /drop multiplexers static and dynamic models. *European Journal of Operational Research*, 108 (1):26 – 35, 1998. ISSN 0377-2217. doi: https://doi.org/10.1016/S0377-2217(97)00021-0.

R. Borndörfer, H. Hoppmann, and M. Karbstein. Passenger routing for periodic timetable optimization. *Public Transport*, 9(1-2):115–135, 2017. doi: 10.1007/s12469-016-0132-0.

U. Brännlund, P. O. Lindberg, A. Nou, and J.-E. Nilsson. Railway timetabling using lagrangian relaxation. *Transportation science*, 32(4):358–369, 1998.

V. Cacchiani and P. Toth. Nominal and robust train timetabling problems. *European Journal of Operational Research*, 219(3):727 – 737, 2012. ISSN 0377-2217. doi: https://doi.org/10.1016/j.ejor.2011.11.003. URL `http://www.sciencedirect.com/science/article/pii/S0377221711009908`. Feature Clusters.

V. Cacchiani, A. Caprara, and P. Toth. A column generation approach to train timetabling on a corridor. *4OR*, 6(2):125–142, 2008.

V. Cacchiani, A. Caprara, and P. Toth. Scheduling extra freight trains on railway networks. *Transportation Research Part B: Methodological*, 44(2):215 – 231, 2010. ISSN 0191-2615. doi: https://doi.org/10.1016/j.trb.2009.07.007.

G. Caimi, L. Kroon, and C. Liebchen. Models for railway timetable optimization: Applicability and applications in practice. *Journal of Rail Transport Planning & Management*, 6 (4):285–312, 2017. ISSN 2210-9706. doi: https://doi.org/10.1016/j.jrtpm.2016.11.002.

A. Caprara, M. Fischetti, and P. Toth. Modeling and solving the train timetabling problem. *Operations Research*, 50(5):851–861, 2002. doi: 10.1287/opre.50.5.851.362.

CBS, PBL, RIVM, WUR. Emissies naar lucht door verkeer en vervoer, 2018. `https://www.clo.nl/indicatoren/nl0129-emissies-naar-lucht-door-verkeer-en-vervoer?ond=20897`, 2019. indicator 0129, versie 34, 27 september 2019.

J. de Dios Ortúzar and L. G. Willumsen. *Modelling transport*. John wiley & sons, 2011.

B. de Keizer, M. Kouwenhoven, and F. Hofker. New insights in resistance to interchange. *Transportation Research Procedia*, 8:72 – 79, 2015. ISSN 2352-1465. doi: https://doi.org/10.1016/j.trpro.2015.06.043. Current practices in transport: appraisal methods, policies and models - 42nd European Transport Conference Selected Proceedings.

F. Farina. A heuristic for periodic train timetabling with integrated passenger routing. Submitted to European Journal of Operational Research, 2019.

GAMS. General algebraic modeling system. URL `http://www.gams.com`.

P. Gattermann, P. Großmann, K. Nachtigall, and A. Schöbel. Integrating Passengers' Routes in Periodic Timetabling: A SAT approach. In M. Goerigk and R. Werneck, editors, *16th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2016)*, volume 54 of *OpenAccess Series in Informatics (OASIcs)*, pages 3:1–3:15, Dagstuhl, Germany, 2016. Schloss Dagstuhl–Leibniz-Zentrum für Informatik. doi: 10.4230/OASIcs.ATMOS.2016.3. URL `http://drops.dagstuhl.de/opus/volltexte/2016/6527`.

M. Goerigk and C. Liebchen. An Improved Algorithm for the Periodic Timetabling Problem. In G. D'Angelo and T. Dollevoet, editors, *17th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2017)*, volume 59 of *OpenAccess Series in Informatics (OASIcs)*, pages 12:1–12:14, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum für Informatik. ISBN 978-3-95977-042-2. doi: 10.4230/OASIcs.ATMOS.2017.12.

R. M. Goverde, N. Bešinović, A. Binder, V. Cacchiani, E. Quaglietta, R. Roberti, and P. Toth. A three-level framework for performance-based railway timetabling. *Transportation Research Part C: Emerging Technologies*, 67:62 – 83, 2016. ISSN 0968-090X. doi: https://doi.org/10.1016/j.trc.2016.02.004. URL `http://www.sciencedirect.com/science/article/pii/S0968090X16000498`.

P. Großmann, S. Hölldobler, N. Manthey, K. Nachtigall, J. Opitz, and P. Steinke. Solving periodic event scheduling problems with SAT. In H. Jiang, W. Ding, M. Ali, and X. Wu, editors, *Advanced Research in Applied Artificial Intelligence*, volume 7345 of *Lecture Notes in Computer Science*, pages 166–175. Springer Berlin Heidelberg, 2012. ISBN 978-3-642-31086-7.

J. Hartleb and M. Schmidt. Railway timetabling with integrated passenger distribution. *Available at SSRN 3505167*, 2019.

J. Hartleb, M. Schmidt, M. Friedrich, and D. Huisman. A good or a bad timetable: Do different evaluation functions agree? ERIM report series research in management Erasmus Research Institute of Management, 2019. URL `http://hdl.handle.net/1765/115831`.

IBM. IBM ILOG CPLEX Optimization Studio 12.9.0, 2019.

L. G. Kroon, L. W. P. Peeters, J. C. Wagenaar, and R. A. Zuidwijk. Flexible connections in pesp models for cyclic passenger railway timetabling. *Transportation Science*, 48(1): 136–154, 2014. doi: 10.1287/trsc.1120.0453.

C. Liebchen. The first optimized railway timetable in practice. *Transportation Science*, 42 (4):420–435, 2008. doi: 10.1287/trsc.1080.0240.

C. Liebchen and R. Möhring. The modeling power of the periodic event scheduling problem: Railway timetables – and beyond. In F. Geraets, L. Kroon, A. Schöbel, D. Wagner, and C. Zaroliagis, editors, *Algorithmic Methods for Railway Optimization: International Dagstuhl Workshop, Dagstuhl Castle, Germany, June 20-25, 2004, 4th International Workshop, ATMOS 2004, Bergen, Norway, September 16-17, 2004, Revised Selected Papers*, pages 3–40, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg. doi: 10.1007/978-3-540-74247-0_1.

C. Liebchen and L. Peeters. Integral cycle bases for cyclic timetabling. *Discrete Optimization*, 6(1):98–109, 2009. ISSN 1572-5286. doi: https://doi.org/10.1016/j.disopt.2008.09.003.

C. Liebchen, M. Proksch, and F. H. Wagner. Performance of algorithms for periodic timetable optimization. In M. Hickman, P. Mirchandani, and S. Voß, editors, *Computer-aided Systems in Public Transport*, pages 151–180, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg. ISBN 978-3-540-73312-6.

J. Lübbe. *Passagierrouting und taktfahrplanoptimierung*. Diploma thesis, Technische Universität Berlin, 2009.

R. Lusby, J. Larsen, and S. Bull. A survey on robustness in railway planning. *European Journal of Operational Research*, 266(1):1–15, 2018. ISSN 0377-2217. doi: https://doi.org/10.1016/j.ejor.2017.07.044.

B. Martin-Iradi and S. Ropke. A column-generation-based matheuristic for periodic train timetabling with integrated passenger routing, 2019.

G. P. Matos, L. M. Albino, R. L. Saldanha, and E. M. Morgado. Solving periodic timetabling problems with SAT and machine learning. In *Conference on Advanced Systems in Public Transport*, 2018.

K. Nachtigall. *A Branch and Cut Approach for Periodic Network Programming*. Hildesheimer Informatik-Berichte. Univ., Inst. für Mathematik, 1994. Technical Report 29.

K. Nachtigall and J. Opitz. Solving Periodic Timetable Optimisation Problems by Modulo Simplex Calculations. In M. Fischetti and P. Widmayer, editors, *8th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems (ATMOS'08)*, volume 9 of *OpenAccess Series in Informatics (OASIcs)*, Dagstuhl, Germany, 2008. Schloss Dagstuhl–Leibniz-Zentrum für Informatik. ISBN 978-3-939897-07-1. doi: http://dx.doi.org/10.4230/OASIcs.ATMOS.2008.1588.

H. Niu, X. Zhou, and R. Gao. Train scheduling for minimizing passenger waiting time with time-dependent demand and skip-stop patterns: Nonlinear integer programming models with linear constraints. *Transportation Research Part B: Methodological*, 76:117–135, 2015.

NS. Website nederlandse spoorwegen. `http://www.ns.nl`.

M. Odijk. A constraint generation algorithm for the construction of periodic railway timetables. *Transportation Research Part B: Methodological*, 30(6):455–464, 1996. ISSN 0191-2615. doi: http://dx.doi.org/10.1016/0191-2615(96)00005-7.

L. Peeters. *Cyclic Railway Timetable Optimization*. PhD thesis, Erasmus University Rotterdam, Jun 2003.

G. J. Polinder, M. Schmidt, and D. Huisman. Timetabling for strategic passenger railway planning. Technical Report ERS-2020-001-LIS, Erasmus Research Institute of Management, Jan. 2020. URL `http://hdl.handle.net/1765/123973`.

A. Radtke. Infrastructure modelling. In I. Hansen and J. Pachl, editors, *Railway Timetable & Operations*. DDV Media Group, 2014.

P. Schiewe and A. Schöbel. Periodic timetabling with integrated routing: An applicable approach. Accepted for publication in Transportation Science, 2019.

M. Schmidt and A. Schöbel. Timetabling with passenger routing. *OR Spectrum*, 37(1):75–97, 2015. ISSN 0171-6468. doi: 10.1007/s00291-014-0360-0.

A. Schrijver and A. Steenbeek. Spoorwegdienstregelingontwikkeling. Technical report, CWI, Amsterdam, 1993. In Dutch.

P. Serafini and W. Ukovich. A mathematical model for periodic scheduling problems. *SIAM Journal on Discrete Mathematics*, 2(4):550–581, 1989. doi: 10.1137/0402049.

M. Siebert and M. Goerigk. An experimental comparison of periodic timetabling models. *Computers & Operations Research*, 40(10):2251 – 2259, 2013. ISSN 0305-0548. doi: https://doi.org/10.1016/j.cor.2013.04.002.

The Mathworks. Matlab. URL `http://www.matlab.com`.

Y. Wang, T. Tang, B. Ning, T. J. van den Boom, and B. De Schutter. Passenger-demands-oriented train scheduling for an urban rail transit network. *Transportation Research Part C: Emerging Technologies*, 60:1–23, 2015.

L. Wolsey. *Integer Programming*. Wiley Series in Discrete Mathematics and Optimization. Wiley, 1998. ISBN 9780471283669. URL `https://books.google.nl/books?id=x7RvQgAACAAJ`.

J. Yin, L. Yang, T. Tang, Z. Gao, and B. Ran. Dynamic passenger demand oriented metro train scheduling with energy-efficiency and waiting time minimization: Mixed-integer linear programming approaches. *Transportation Research Part B: Methodological*, 97:182–213, 2017.

Y. Zhang, Q. Peng, Y. Yao, X. Zhang, and X. Zhou. Solving cyclic train timetabling problem through model reformulation: Extended time-space network construct and alternating direction method of multipliers methods. *Transportation Research Part B: Methodological*, 128:344–379, 2019.

Y. Zhu, B. Mao, Y. Bai, and S. Chen. A bi-level model for single-line rail timetable design with consideration of demand and capacity. *Transportation Research Part C: Emerging Technologies*, 85:211–233, 2017.