This is the accepted version of a paper presented at 7th International Conference on Railway Operations Modelling and Analysis (RailLille 2017), Lille, France, April 4th-7th 2017.

Citation for the original published paper:

Högdahl, J., Bohlin, M., Fröidh, O. (2017) Combining Optimization and Simulation to Improve Railway Timetable Robustness In:

N.B. When citing this work, cite the original published paper.

Permanent link to this version:
http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-262872
Combining optimization and simulation to improve railway timetable robustness

Johan Högdahl\textsuperscript{a,1}, Markus Bohlin\textsuperscript{a,b}, Oskar Fröidh\textsuperscript{a}

\textsuperscript{a} Department of Transport Science, KTH Royal Institute of Technology
SE-100 44, Stockholm, Sweden
\textsuperscript{1} E-mail: jhogdahl@kth.se
\textsuperscript{b} SICS Swedish ICT Västerås AB
Kopparbergsvägen 10, SE-722 13 Västerås, Sweden

Abstract
The Train Timetabling Problem (TTP) is the problem of finding the timetable that utilizes the infrastructure as efficient as possible, while satisfying market demands and operational constraints. As reliability is important to passengers it is important that timetables are robust. In this paper we propose a method that combines optimization and simulation to find the timetable that minimizes the travel times and maximizes the expected punctuality. The core method consists of iteratively re-optimizing a bi-objective mixed integer sequencing timetable model, where both planned travel time and simulated delays are taken into account. Each generated timetable is validated and re-evaluated using the micro-simulation tool RailSys. The advantage of the method is that it captures both the uncertainty of a timetable at the planning stage and the validity of the generated timetable. The method is evaluated on a unidirectional track section of the Western Main Line in Sweden and shows promising results for future research.

Keywords
Railway timetabling, Robustness, Optimization, Simulation, Punctuality

1 Introduction

Increasing demand for passenger and freight transportation on the railway network creates new challenges for the supply of rail services at an acceptable level of quality. While improvements of the infrastructure to overcome capacity constraints is a medium to long-term issue, planning of the services is essential for optimal use of the resources in a short term. The timetables are the base for railway operations. The goal to achieve is railway timetables that satisfies market needs and utilizes the infrastructure efficiently. However, high capacity utilization or non-optimal solutions might lead to sensitiveness for disturbances which leads to low punctuality, lack of reliability and consequently low attractiveness of the railway services. Therefore, it is of interest to develop methods that can be used, in a large scale, to improve the robustness of railway services while maintaining a high level of service by better or optimal timetables.

"Robustness" is a key concept in this paper. Andersson (2014, p.11) define robustness as a "timetable in which trains should be able to keep their originally planned train slot despite small delays and without causing unrecoverable delays to other trains". Dewilde (2014,
p.35) includes a socio-economic component in the definition: "A railway system that is robust against the daily occurring, small disturbances minimizes the Real Weighted Travel Time (RWTT) of passengers". Both authors emphasize, in other words, the importance of margins in the timetable for handling smaller perturbations, which have a wide array of origins but nevertheless complicates railway planning and operations.

In this paper we propose a new method, combining micro-simulation and optimization to improve the robustness of a basically non-periodic timetable. Simulation is used to obtain punctuality data of a timetable. The punctuality data is then used as input to the optimization model, and a new timetable is calculated such that it minimizes the travel time and maximizes the punctuality.

The work in this paper is carried out in the following way. First, we propose a way to combine simulation and optimization into one method. This method is then implemented and applied to the timetable for a section of the Western Main Line in Sweden. Validation of the generated timetables are carried out with micro-simulations in Railsys.

The computational experiments conducted in this paper indicate that it is possible, in practice, to apply this method to increase robustness in rail operations.

This paper consists of six parts, including this introduction. In the next section we review some related work in this field. In the third section we present the train timetabling problem and show how it is possible to formulate it as a mixed integer program. The proposed method of this paper is described in the fourth section, and the computational experiments in the fifth. Finally, the discussion and conclusions rounds of the contribution in the sixth section.

2 Related Work

The train timetabling problem (TTP) aims to allocate train paths in a way that utilizes the infrastructure as efficient as possible (capacity utilization), while satisfying the operational constraints such as allocation of production resources, service and quality commitments. Abril et al. (2008) states that "there is a trade-off between capacity and reliability/robustness", which also could be interpreted as a difference between technical (theoretical) and feasible (practical) limits of capacity in terms of robustness.

Harrod (2012) characterize four methods for solving the TTP, namely Mixed Integer Sequencing Linear Programs (MISLP), Binary Integer Occupancy Programs (BIOP), the Hypergraph Formulation and Periodic event scheduling problems (PESP), where the three first mentioned are suitable for aperiodic scheduling and PESP for periodic. MISLP and PESP are event-based, while BIOP and the Hypergraph Formulation also explicit takes track layouts into account.

One underlying assumption in the models above is that all the trains will run according to the timetable without any disturbances. In real operation, this is not a valid assumption as trains will get delayed and the delays will propagate in the network and affect other trains as well, i.e. dynamics in terms of operational perturbations occur. Methods in the literature that deals with uncertainty in data (in the context of optimization) can be divided into stochastic programming methods and robust programming methods. However, Fischetti and Monaci (2009) concludes that stochastic programming and robust programming often tend to become too complicated and to give too conservative solutions, respectively.

Salido et al. (2012) propose analytical and simulation methods to measure robustness in a single railway line. Sels et al. (2012) designed an optimal passenger train timetable
by means of journey time, also taking into account typical train delays but treat production factors as circulation time constant. The authors analytically derive total stochastic expected passenger time as a closed formula, linearize it and use it as an objective function for optimizing the timetable using a mixed integer linear programming (MILP) model.

Kroon et al. (2008) describes a two-stage stochastic optimization model, that combines a timetabling model with a simulation model, to improve the robustness of a timetable by allocating time supplements and buffer times.

Hassannayebi et al. (2014) developed a two-stage GA-based simulation optimization approach in order to minimize the expected passenger waiting times. The optimization is intended to adjust headways through simulation experiments to achieve robust timetables for operation of an urban transit rail system. A further developed methodology with robust against variations in the demand, multi-objective stochastic programming models for train timetabling is presented by the same author (Hassannayebi et al., 2016).

Fischetti and Monaci (2009) have proposed a method called Light Robustness to solve linear programming (LP) problems with uncertainty in data. In this approach the maximum objective value deterioration is fixed and a "robustness goal" is modeled using a classical robust optimization framework. Compared to some stochastic programming models, of various complexity, they conclude that Light Robustness seem to be the most suitable tool to solve large-scale real scenarios.

Forsgren et al. (2012) introduced the planning approach of successive allocation of train paths (referred to as Successiv tilldelning in Swedish). They developed a method for optimization of timetables by redistribution of buffer time to minimize train running times by reallocation of train passings, with regards to robustness. The same thoughts on flexibility in timetables appear in D’Ariano et al. (2008). The authors claim less (rigid) timetable planning in favor of operational decisions within the buffer time would benefit overall punctuality. Jovanović et al. (2016) focused on optimal distribution of buffer times based on priority of events, modeled as a knapsack problem, not to consume too much capacity.

Khoshniyat and Peterson (2015) observed that delays seem to increase as the travel time increases. To compensate for this, they proposed a strategy where the headway increases along with the journey of each train.

Robustness in Critical Points (RCP) is a concept for improving the punctuality for single trains. Critical points refer to very time-sensitive dependencies between different pairs of trains at different locations in the network (Andersson et al., 2013 p.7). The amount of and distribution of margin time in the timetable has impact on the overall robustness. A mixed integer linear programming (MILP) model was proposed to redistribute the margin time in an existing timetable. Applied to a realistic example the model achieved a 28% reduction in total delay at the final destination compared to a not optimized timetable (Andersson et al., 2015).

Some examples of linear programming (LP) approaches includes Pouryousef et al. (2016) who uses a multi-objective linear programming model called “Hybrid Optimization of Train Schedules” (HOTS) together with a rail simulation tool (i.e. RailSys) to improve capacity utilization or level of service. The HOTS model uses both conflict resolution and timetable compression techniques but has no strong focus on dynamics in the operations, i.e. to minimize delays. Another recent example, that contrary to the previous one aims at improving punctuality, is the “Timetable Improvement Heuristic” (TIH) proposed by Lee et al. (2017). The TIH utilizes micro-simulation and a linear program to iteratively reallocate existing
buffer times and time supplements of a timetable. Chen et al. (2014) developed a timetable generator which optimizes the travel time, with consideration of train circulation including service times. Sensitivity analysis shows the effects in case of delays. A similar approach did Xie and Li (2012) who used integer linear programming to solve the scheduling problem with respect to circulation issues and connecting trains.

Numerous timetabling studies for rail services in different aspects have been conducted, including optimization as well as simulation studies of rail capacity. A conclusion of the literature review is however that combined optimization and simulation techniques to achieve optimal performance could be found in other areas, but in rail applications to solve the TTP most are focused on either optimization or simulation, or limited approaches in combining the methods. When it comes to robust timetabling we have found two methods that combines simulation and optimization. The first method, by Kroon et al. (2008), differs from our method in the sense that they have integrated the simulation part in the optimization model, and the simulation model is on macro-level. The second method, by Lee et al. (2017), utilizes a separate micro-simulator like we do, but do not model disturbances explicit in the formulation of the optimization program. The potential benefit of combining optimization and micro-simulation including the infrastructure is the ability to create and evaluate robust timetables.

3 The train timetabling problem

The train timetabling problem can be described as follows: Consider a railway network, that consists of a number of stations (nodes) connected with tracks, and a set of railway lines that we wish to timetable. The problem is then to find a timetable, such that it satisfies the operational constraints, and utilizes the infrastructure as efficient as possible.

We will focus on a one-direction single track line (i.e. one of the tracks on a double-track line), connecting two endpoint stations, with a number of intermediate stations. Several models exists to formulate this as a optimization problem. We base our formulation on a similar notation as used by Fischetti et al. (2009) and formulate the problem as follows.

Let \( S \) be the set of stations and let \( T \) be the set of trains. For each station \( s \in S \) there is a set of arrival events, \( A_s \), and a set of departure events, \( D_s \). For each train \( h \in T \), let \( E_h \) be the set of all events, \( E_h^a \) be the set of all arrival events and \( E_h^d \) be the set of all departure events.

Let the events for each train \( h \) be arranged in an ordered sequence of arrival and departure events, such that \( t_i^h \leq t_{i+1}^h \), where \( t_i^h \) denotes the time of event \( i \) for train \( h \). Let the last event of train \( h \) be denoted with \( N_h \).

We define a timetable, \( t \), as the vector of all events, such that

\[
    t = (t_1^1, t_2^1, \ldots, t_{N_1}^1, \ldots, t_1^H, t_2^H, \ldots, t_{N_H}^H),
\]

where \( H \) is the number of trains in \( T \).

The operational constraints that must be considered are summarized in the list below.

1. The time difference between the departure time from one station and the arrival time to the following station must be larger than the technical minimum running time.
2. Two trains arriving to the same station must be separated with a minimum headway.
3. Two trains departing from the same station must be separated with a minimum headway.
4. Trains can only overtake each other at stations.

The fourth operational constraint can also be expressed as: if one train departs earlier than another train from one station, then the first train must also arrive earlier than the latter train at the following station.

The objective function, denoted \( f(t) \), should be a linear function (since it is a MILP model) expressing the cost or efficiency of the timetable \( t \).

Based on the description above the train timetabling problem is now formulated as the following optimization problem.

\[
\begin{align*}
\text{minimize} & \quad f(t) \\
\text{s. t.} & \quad t_{i+1}^h - t_i^h \geq d_{i,i+1}^h, \quad i = 1, \ldots, N_h - 1, \forall h \in T \\
           & \quad |t_i^h - t_j^h| \geq \Delta, \quad \forall t_i^h, t_j^h \in A_s, \forall s \in S \\
           & \quad t_{i+1}^h < t_{j+1}^k \\
           & \quad \Leftrightarrow t_i^h < t_j^k, \quad \forall t_i^h, t_j^k \in D_s, \forall s \in S \\
           & \quad l \leq t_i^h \leq u, \quad \forall t_i^h \in E_h, \forall h \in T
\end{align*}
\]

(P1)

The objective function is a linear function and the numbering of the constraints coincides with the numbering of the constraints in the list above. Due to the absolute value constraints this formulation is not linear. To obtain a linear formulation of the problem, it is also necessary to reformulate constraint (4) into a linear equation.

Fischetti et al. (2009) simulates the effect of the absolute value by utilizing binary variables and Big-M constraints, which is a common technique for linearizing absolute value constraints. By this approach we obtain

\[
|t_i^h - t_j^k| \geq \Delta \Leftrightarrow \begin{cases} 
  t_i^h - t_j^k \geq \Delta - M x_{i,j}^{h,k} \\
  t_j^k - t_i^h \geq \Delta - M x_{j,i}^{k,h} \\
  x_{i,j}^{h,k} + x_{j,i}^{k,h} = 1,
\end{cases}
\]

where \( M \) is a sufficiently large constant and \( x_{i,j}^{h,k} \) is a binary variable that is equal to 1 iff event \( i \) for train \( h \) occurs before event \( j \) for train \( k \).

By using the binary \( x \)-variables it is also possible to replace constraint (4) with the following equation:

\[
x_{i,j}^{h,k} = x_{i+1,j+1}^{h,k}, \quad \forall t_i^h, t_j^k \in D_s, \forall s \in S
\]

4 A bi-objective optimization approach combined with simulation

The model described in Section 3 assumes that the trains are operated according to the timetable, without any delays. A highly efficient timetable (in terms of capacity utilization) will likely be sensitive to disturbances and therefore not be robust.

A simple way to add robustness to a timetable is to add supplements to the running times and at the stations, and extra buffer times between the trains. However, this comes with the
Table 1: A method that combines simulation and optimization to improve the robustness of a timetable.

<table>
<thead>
<tr>
<th>Step 0 (Initialization):</th>
<th>Choose an initial feasible timetable $t^0$ and let $k = 0$.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 (Termination):</td>
<td>If a termination criterion is satisfied, then stop.</td>
</tr>
<tr>
<td>Step 2 (Simulation):</td>
<td>Run $K$ simulations of timetable $t^k$. Calculate punctuality data.</td>
</tr>
<tr>
<td>Step 3 (Optimization):</td>
<td>Use the punctuality data as input to problem (P2), see below. Then, find the optimal timetable $t^*$ by solving (P2).</td>
</tr>
<tr>
<td>Step 4 (Update):</td>
<td>Let $t^{k+1} = t^*$, set $k = k + 1$ and go to step 1.</td>
</tr>
</tbody>
</table>

price of reduced capacity and possibly less attractiveness of the railway due to increased travel time.

When constructing a timetable, the supplementary time is added to reduce the risk that trains get delayed. Independently of how supplements are added (either by preponing the departure or by postponing the arrival) it will lead to smaller margins between adjacent trains. Therefore, if supplements is not added with care, it is possible that knock-on delays will spread more easily in the system.

To address the issues of distributing supplements, so that both the travel time and the expected delay is minimized, we propose a method, described in Table 1. This method utilizes a bi-objective optimization program (step 3), in which delay distributions are modeled using data from the simulations (step 2).

### 4.1 Bi-objective optimization model

The optimization problem in step 3, in Table 1, is based on the formulation given in Section 3. As in problem (P1), the decision variables is the time, $t^h_i$, of arrival and departure events $i$ for each train $h$. If we assume that the delays are only minor, so that the order of the trains are kept, then we can let the binary variables be fixed, so that

$$x^h_{i,j} = \bar{x}^h_{i,j},$$

where $\bar{x}^h_{i,j}$ is equal to 1 iff the $i$:th event of train $h$ occurred earlier than the $j$:th event of train $k$ in the input timetable $t$.

As described earlier the aim with the optimization step is to minimize the travel time and the expected delay for each train. Let $\alpha \in [0, 1]$ be a parameter that weights the importance of the two objectives. Let $\Omega^h_i$ be the set of all possible delays for train $h$ at event $i$, let $p^h_i(\omega)$ be the probability that train $h$ is delayed with $\omega \in \Omega^h_i$ at event $i$ and let $t^h_i(\omega)$ be the time of the $i$:th event for train $h$, with the delay $\omega$. Let $\lambda^h_i \in [0, 1]$ be the proportion of travelers, or freight, that are transported by train $h$ from the starting station to the station where the $i$:th
event occurs. Then we can formulate the new problem as follows:

\[ \text{(P2)} \]

\[
\begin{align*}
\text{minimize} \quad & f(t) = \alpha \sum_{h \in T} \sum_{i \in E_h^b} \lambda_i^h \left( t_i^h - t_i^h \right) \\
& + (1 - \alpha) \sum_{h \in T} \sum_{i \in E_h^s} \lambda_i^h p_i^h(\omega) \max \left\{ 0, t_i^h(\omega) - \left( t_i^h - \phi_i^h(t_i^h, t_j^k) \right) \right\} \\
\text{s. t.} \quad & t_{i+1}^h - t_i^h \geq d_{i+1}^i, \quad i = 1, \ldots, N_h, \forall h \in T \quad (6) \\
& t_j^h - t_j^h \geq \Delta_a - M x_{i,j}^{h,k}, \quad \forall t_j^h, t_j^h \in A_s, \forall s \in S \quad (7) \\
& t_k^h - t_j^h \geq \Delta_a - M x_{k,j}^{h,k}, \quad \forall t_k^h, t_k^h \in A_s, \forall s \in S \quad (8) \\
& t_i^h - t_j^h \geq \Delta_a - M x_{i,j}^{h,k}, \quad \forall t_i^h, t_i^h \in D_s, \forall s \in S \quad (9) \\
& t_i^h - t_j^h \geq \Delta_a - M x_{k,j}^{h,k}, \quad \forall t_i^h, t_i^h \in D_s, \forall s \in S \quad (10) \\
& \phi_i^h(t_i^h, t_j^h) = \\
& \begin{cases} \\
& t_j^h - t_j^h, \quad \text{if } t_j^h - t_j^h < b \quad \text{such that } t_j^h \text{ is the previous event of } t_i^h, \\
& 0, \quad \text{else.} \end{cases} \forall h \in T \quad (11) \\
& t_i^h = t_i^h, \quad \forall h \in T \quad (12)
\end{align*}
\]

The objective function models the trade-off between travel time and expected delay. The first sum in the objective function models that we want to minimize the travel time to all stations where each train stops. The parameter \( \lambda_i^h \) weights the importance of each stop according to how many travelers or freight that have event \( i \) as their destination. The second sum in the objective function models how we expect that the delays are affected by the scheduled arrival times, and each term is positive since we want to avoid negative delays to improve the objective function value. The main idea is that if we postpone the arrival time of event \( j \) for train \( k \), then the probability that this train arrives late is decreased as if the probability distribution where shifted to the left. However, if the distance to the train behind, train \( h \), is not big enough, then it is likely that the punctuality of train \( h \) will be dependent on the punctuality of train \( k \), which is modeled with the function \( \phi_i^h(t_i^h, t_j^h) \). This dependency between two trains is explained in Figure 1.

Constraints (6) to (10) is the same as for problem (P1), but here the sequencing variables, \( x_{i,j}^{h,k} \), are fixed which allows us to drop constraint (4) of problem (P1). Constraint (11) defines the function \( \phi_i^h(t_i^h, t_j^h) \), which becomes non-zero when the distance between train \( k \) and the train immediately behind it, train \( h \), is less than some constant \( b > 0 \). The parameter \( t_j^h \) denotes the time of event \( j \) for train \( k \) in the input timetable.

The fixation of the first event for each train in constraint (12) is necessary to prevent all trains from just being shifted infinitely to the right (in a graphical timetable with the time on the horizontal axis), which would otherwise be the trivial solution to this problem.

The formulation of problem (P2) is not a mixed integer problem. To obtain a MIP we must replace the \( \max \)-function and \( \phi_i^h(t_i^h, t_j^h) \) in the objective with linear expressions.

First, we linearize the function \( \phi_i^h(t_i^h, t_j^h) \). This is done by introducing the binary variable \( z_i^h \), which is equal to one if \( t_i^h - t_j^h < b \) and otherwise zero, and a state variable \( \Delta_a^{h,k} \).
By utilizing $z_i^h$ and $\Delta_{i,j}^{h,k}$ we can now write $\phi_i^h(t_j^k)$ as
\[
\phi_i^h(t_j^k) = (t_i^k - t_j^k) z_i^h = \Delta_{i,j}^{h,k}.
\] (13)

Let $\Delta_{i,j}^{h,k}$ replace $\phi_i^h(t_j^k)$ in the objective function of problem (P2) and replace constraint (11) with the following set of constraints:
\[
\begin{align*}
t_i^h - t_j^k - b & \geq -M z_i^h \\
t_i^h - t_j^k - (b - \epsilon) & \leq M (1 - z_i^h) \\
\Delta_{i,j}^{h,k} & \geq -M z_i^h \\
\Delta_{i,j}^{h,k} & \leq M z_i^h \\
\Delta_{i,j}^{h,k} & \geq (t_i^k - t_j^k) - M (1 - z_i^h) \\
\Delta_{i,j}^{h,k} & \leq (t_i^k - t_j^k) + M (1 - z_i^h),
\end{align*}
\] (14)
where $\epsilon \geq 0$ is a small constant simulating the strict inequality in constraint (11) and $M$ is a sufficiently large, possibly very large, constant.

Next, we linearize the objective function of problem (P2). To begin, we note that
\[
\min f(x) = \max \{0, x\} \quad \Leftrightarrow \quad \begin{cases} 
\min y \\
\text{s. t.} \quad x \leq y \\
y \geq 0.
\end{cases}
\]

Thus, by introducing $y_{i,\omega}^h \geq 0$ and adding
\[
t_i^h(\omega) - (t_i^h - \Delta_{i,j}^{h,k}) \leq y_{i,\omega}^h
\] (15)
to the constraints of problem (P2), we can replace the objective function with the following linear function:

$$f(t) = \alpha \sum_{h \in T} \sum_{i \in A_h} \lambda^h_i \left( t^h_i - t^h_{i,1} \right) + (1 - \alpha) \sum_{h \in T} \sum_{i \in A_h} \sum_{\omega \in \Omega} \lambda^h_i p^h_i(\omega) y^h_{i,\omega}.$$  \hspace{1cm} (16)

The generation of input timetable and details regarding the implementation of this model is described in the following chapter.

5 Computational experiments

5.1 Implementational details

The simulation environment
The simulations have been carried out in Railsys (version 9.8.25), which is a commercial software developed by RMCon used to model, simulate, and evaluate railway traffic. The Swedish Transport Administration (Trafikverket) have provided us with a model over the Swedish railway network and the 2015 national timetable.

From the railway network we have extracted the part between Gnesta (63 km southwest of Stockholm) and Partille (near Gothenburg) on the Swedish Western Main Line, 383 km of total 455 km between endpoints, excluding multiple track sections and the Stockholm commuter services. The selected line is an electrified double track line with Centralized Traffic Control (CTC) and it has 53 nodes (stations and halts), including passing (or overtaking) tracks.

From the national Swedish timetable we have extracted the southbound trains that operates the selected route on September 23, 2015, between 5:00am and 12:00pm. This selection includes 87 trains which can be divided into the following categories:

- 9 freight trains (varying train weight, 70-100 km/h top speed)
- 17 commuter trains (frequent stops, Alingsås-Gothenburg only, 140 km/h top speed)
- 40 Intercity/regional trains (some stops, 160-200 km/h top speed)
- 21 fast trains (interregional, none or few stops, 200 km/h top speed).

To ensure that all trains arrive at their destination the simulation have been carried out between 5:00am and 6:00pm.

In the simulations perturbed timetables have been generated. The perturbation statistics have been provided by the Swedish Transport Administration. The applied perturbations can be divided into entry delays (which are added to the scheduled starting time), dwell time delays (which are added to the minimum dwell time) and runtime delay (which are added to the minimum technical run time).

The input timetable
The original timetable, that where provided by the Swedish Transport Administration, contained some trains with to small headways and some trains with scheduled runtimes that where less than the minimum technical runtime (computed by Railsys). Therefore, in order to make a fair comparison with the optimized timetables, it was necessary to construct a feasible input timetable. This timetable have been constructed by having the original timetable
as initial timetable while solving a variant of problem (P1), where the following principles have been adopted:

- The runtime between two stations shall be the largest value of the minimum technical runtime and the originally scheduled runtime
- The stopping times shall be equal to the stopping times in the original timetable
- The headway shall be larger than the minimum headways (see below)
- Minimize the translation in time of each train.

According to the recommendations by the Swedish Transport Administration for the 2015 national timetable (Trafikverket, 2014) the minimum headways shall be as follows:

- Gnesta (Gn) - Hallsberg (Hpbg): 5 minutes
- Hallsberg (Hpbg) - Alingsås (A): 4 minutes
- Alingsås (A) - Partille (P): 5 minutes

Figure 2 show the graphical timetable of the input timetable. From this figure can we see that we have mixture of trains operating on different parts of the line. In the square boxes we highlight some examples where neighboring trains operate with different speed. The circles shows where we have scheduled overtakings.
The optimization model

The optimization problem have been implemented in Python (version 2.7.12) with Gurobi (version 6.5.2) as an optimization solver. The parameters of problem (P2) have been chosen as follows.

The OD-pair weights, $\lambda^h_{i}$, have been generated using two different principles: one for the passenger trains and one for freight trains (A more realistic model could have been adopted, but for the purpose of evaluating our method we claim that it is not necessary). For the passenger trains we have assumed that all passengers enters the train at the first station and that half of them travels to the end station. For the intermediate stops, we assume that equally many passengers leave the train at each stop. This gives, as an example, that if one train stops at six locations, then the final station will have the weight $\lambda^h_{N_h} = 0.5$ and the other stations the weight $\lambda^h_{i} = 0.1$, for $i \in E^h_a \setminus N_h$, which is illustrated in Figure 3. For the freight trains we have assumed that all freight is heading to the final station. This gives that $\lambda^h_{N_h} = 1$ and $\lambda^h_{i} = 0$, for $i \in E^h_a \setminus N_h$.

The delay distributions, $p^h_i(\omega)$, have been calculated from the simulation output generated by Railsys. Only the simulations were the largest delay is less than, or equal to, 10 minutes have been considered.

When $i$ and $i + 1$ are events at different stations for train $h$, we have used the minimum technical runtime as the value of $d^h_{i,i+1}$. The minimum technical runtime have been imported from Railsys. When $i$ and $i + 1$ are events at the same station (i.e. a stop) for train $h$, we have used the scheduled dwell time of the input timetable as the value of $d^h_{i,i+1}$.

The headway constants, $\Delta_a$ and $\Delta_d$, have been set to the minimum headways (see the input timetable-section above).

Finally, the train independence factor, $b$, have been set to 10 minutes and the Big-M constant have been set to $M = 10^5$. 

Figure 3: Example of how the number of passengers decrease as a train travels towards its destination. For each intermediate destination, number 1-5, the number of passengers decreases with 10%, i.e $\lambda^h_{i} = 0.1$, and at the final destination, number 6, the remaining 50% of the passengers leave the train, i.e $\lambda^h_{N_h} = 0.5$. 

Table 2: Percentage of passengers and freight arriving within a specified maximum delay. Each row show the percentage of passengers and freight that arrive within the maximum delay, specified in the leftmost column, for timetables generated with different values of $\alpha$, one per column. The rightmost column show the corresponding values for the input timetable, included as a reference.

<table>
<thead>
<tr>
<th>Value of $\alpha$</th>
<th>0.01</th>
<th>0.1</th>
<th>0.2</th>
<th>0.3</th>
<th>0.4</th>
<th>0.5</th>
<th>0.6</th>
<th>0.8</th>
<th>1</th>
<th>Input</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max delay 0</td>
<td>0</td>
<td>55.0</td>
<td>49.9</td>
<td>47.9</td>
<td>41.8</td>
<td>27.1</td>
<td>2.8</td>
<td>2.2</td>
<td>2.7</td>
<td>1.9</td>
</tr>
<tr>
<td>1</td>
<td>91.5</td>
<td>92.1</td>
<td>90.8</td>
<td>82.7</td>
<td>68.1</td>
<td>48.4</td>
<td>47.3</td>
<td>47.4</td>
<td>47.1</td>
<td>85.9</td>
</tr>
<tr>
<td>3</td>
<td>99.1</td>
<td>98.4</td>
<td>98</td>
<td>96.3</td>
<td>87.5</td>
<td>81.5</td>
<td>80.9</td>
<td>80.8</td>
<td>80.5</td>
<td>97.2</td>
</tr>
<tr>
<td>5</td>
<td>99.8</td>
<td>99.7</td>
<td>99.6</td>
<td>99.2</td>
<td>97.2</td>
<td>95.9</td>
<td>95.8</td>
<td>95.7</td>
<td>94.7</td>
<td>99.2</td>
</tr>
<tr>
<td>10</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>99.9</td>
<td>99.8</td>
<td>99.8</td>
<td>99.9</td>
<td>100</td>
</tr>
<tr>
<td>30</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
</tbody>
</table>

5.2 Results

Experiments have been carried out to evaluate the performance of the proposed method, described in Table 1. However, due to limitations in the simulation environment we have only performed one iteration.

First, we performed 200 simulations of the input timetable, with randomly generated disturbances in each of them. Then, we extracted data from the 186 simulations where the largest delay, at any event, is less than or equal to 10 minutes. The extracted data, that contains scheduled and actual arrival and departure times, where then used to calculate the delay distributions in the optimization model.

Optimized timetables have been generated for 100 different choices of $\alpha \in [0, 1]$. Of these 100 timetables, 11 have been selected for validation. The validation have been carried out by performing 50 simulations and measure the punctuality.

Table 2 show the percentage of passengers and freight that arrive within a specified maximum delay. This value is calculated by, for each train, computing the percentage of passenger and freight that arrive within the specified maximum delay, which gives the punctuality for the passengers (or freight) of each train. If, for instance, a passenger train stops at two stations, then half of the passengers are assumed to leave the train at each stop, respectively. If the delay at the first stop is 4 minutes and 2 minutes at the second stop, then 100% of the passengers arrive within a 5 minute maximum delay, but only 50% of them within a 3 minute maximum delay. The percentage of passengers and freight that arrive within a specified maximum delay is then taken as the average punctuality for each train.

In Table 2 we see that, for the input timetable roughly one third of the passengers and freight arrive at the scheduled time or earlier, and within a 5 minute delay nearly all passenger and freight have arrived. Compared to the input timetable, we observe an improved punctuality for the optimized timetables when $\alpha \leq 0.2$. In particular we observe a significant improvement when the maximum delay is either zero or one minute. We also note that, compared to the input timetable, the percentage of passengers and freight that arrive within a specified maximum delay is significantly worse for $\alpha \geq 0.4$. 

Figure 4: Comparison between the Pareto front and the validated timetables. The solid blue line is the Pareto front for the trade-off between travel time and expected delay, the red crosses marks the performance of the optimized timetables for different values of $\alpha$ when simulated, and the yellow dot marks the performance of the input timetable.

Figure 4, show the Pareto front for the trade-off between the total scheduled weighted travel time and the total weighted expected delay, computed as in the first and the second sum of equation (16), respectively. The blue line show the expected performance of the optimized timetables, where each dot marks a timetable with a specific value of $\alpha$. The red crosses show the results from the validation of the optimized timetables. Finally, the yellow dot show the performance of the input timetable (when all 200 simulations is considered), which is included as a reference. From this figure we see that decreasing either the travel time or the expected delay leads to an increase of the other aspect. It also appears that small changes in the scheduled travel time can have a large impact on the punctuality. For instance the third (i.e. $\alpha = 0.2$) and the forth (i.e. $\alpha = 0.3$) red cross, counted from the left, has a very similar scheduled travel time but the observed delay ranges from about 2000 to 3000 seconds. An observation in Figure 4 is that the input timetable actually lies very close to the Pareto front, which is further elaborated in the discussions.

6 Discussion and conclusions

In this paper we have proposed a method to improve railway timetable robustness by combining optimization and simulation. We have derived an optimization model that captures
the trade-off between travel time and expected delay, while simultaneously considering the headway between unidirectional trains of a basically non-periodic timetable.

The method have been implemented but we have only performed one iteration, which in principle is to simulate an input timetable, use the simulation data to model delays in the optimization problem, and then solving it to generate a new timetable. Validation of the generated timetables indicate that it is possible, in practice, to apply this method in order to reduce delays and increase robustness in rail operations. Only small perturbations less than 10 minutes were included.

The results have shown that the punctuality is sensitive to the choice of $\alpha$ and that small changes in the scheduled travel time may have a large impact on the punctuality of the timetable. Further research may reveal if it is possible to determine an appropriate value of $\alpha$ to, in general, obtain a well-balanced timetable, but it might as well be the case that $\alpha$ have to be tuned for each scenario.

In cases where small changes in the scheduled travel time gives major changes in the punctuality it is simple to determine which is the best scenario, namely the one with highest punctuality. However, it might as well be the case that a large improvement in punctuality is achieved at a relatively large cost in terms of longer travel times (compare for instance the input timetable with the optimized timetable for $\alpha = 0.01$, which is the leftmost red cross in Figure 4). In situations like this it could be appropriate to combine our method with a socio-economic analysis, but also other parameters such as commercial aspects might be necessary to take into account. However, these are aspects that lies outside the scope of this paper.

The input timetable was found to be relatively robust and very close to the Pareto front. Provided that the combined optimization and simulation method works as expected, it indicates that the current process of constructing national timetables is without obvious flaws. We have however made some adjustments to the input timetable before simulation since it had some built-in discrepancies in terms of less than minimum running times and headways, which improved the conditions. Further analysis includes an array of input timetables which will reveal if this is still true.

Future research could focus on the following areas: Evaluate the performance when the method is iteratively repeated, model more complex railway networks, or compare the performance of this method with other methods that aims to improve railway timetable robustness.

**Acknowledgments**

This research is conducted within KTH Royal Institute of Technology’s research project Flexibility and recovery capability as timetable measures (*Flex˚Ater - Flexibilitet och återställningsförmåga som tidtabellsmätt*), funded by the Swedish Transport Administration (Trafikverket).

**References**


