Microscopic Simulation of Decentralized Dispatching Strategies in Railways

R.N. van Lieshout[1,2], J.M. van den Akker[3], R. Mendes Borges[4], T. Druijf[3,4], and E. Quaglietta[4]

[1]Econometric Institute and ECOPT, Erasmus University Rotterdam, P.O. Box 1738, 3000 DR Rotterdam, The Netherlands

[2]Corresponding author. Email address: vanlieshout@ese.eur.nl

[3]Department of Information and Computing Sciences, Utrecht University, Utrecht, The Netherlands

[4]Department of Transport and Planning, Delft University of Technology, Delft, The Netherlands

Econometric Institute Research Papers, EI-1708

November 2021

Keywords: decentralized control, local dispatching, microscopic simulation, rescheduling.

Abstract

This paper analyzes the effectiveness of decentralized strategies for dispatching rolling stock and train drivers in a railway system. Such strategies give operators a robust alternative in case centralized control fails due to an abundance of infrastructure or rolling stock disruptions or information system malfunctions. We test the performance of four rolling stock and two driver dispatching strategies in a microscopic simulation. Our test case is a part of the Dutch railway network, containing eleven stations linked by four train lines. We find that with the decentralized dispatching strategies, target frequencies of the lines are approximately met and train services are highly regular without large delays. Especially strategies that allow rolling stock to switch between lines result in a high performance.

1 Introduction

When railway operations become heavily disrupted, central traffic control may lose the overview of the system and terminate all traffic. In such extreme events, decentralized dispatching strategies could provide a robust back-up plan, offering passengers a service that may not be as good as in regular circumstances, but is much preferred over the alternative of no service at all. In this paper, we analyze whether decentralized dispatching can indeed serve as a contingency plan when centralized dispatching is impossible. To do so, we develop an integrated platform that simulates decentralized dispatching strategies in a microscopic representation of the railway system.

The relevance of decentralized control of railway systems can be motivated by so-called *out-of-control* situations, which we define as situations "where dispatchers cease to have an overview of the system and consequently decide to terminate all railway traffic in the affected region, even though the required resources (infrastructure, rolling stock and crew) might be available" (Dekker et al., 2021). Out-of-control situations are often caused by extreme weather events, (possibly short-lasting) power outages, or malfunctioning of telecommunication systems. These situations are typically characterized by a large number of affected resources and incomplete information, yielding traditional rescheduling approaches ineffective. Instead, decentralized decision-making is more robust and better suited for out-of-control situations.

Theoretical justification for the good performance of decentralized dispatching is provided in Van Lieshout et al. (2021), which proved analytically that, under several assumptions, an easy-to-implement dispatching policy matches the performance of centralized dispatching in the long run. A first attempt in applying decentralized dispatching in railways is made in Van Lieshout et al. (2020), which proposed and tested decentralized strategies for dispatching rolling stock. A macroscopic simulation of a part of the Dutch railway network showed that the decentralized approach can quickly restore services to a reasonable level. In this paper, we take the next step in validating the adequacy of decentralized control by testing the rolling stock strategies proposed in Van Lieshout et al. (2020) in a full scope simulation of the railway network, including train drivers and infrastructure at a microscopic level of detail. As decentralized dispatching of train drivers has, to the best of our knowledge, not been considered before, we propose and assess two new driver dispatching strategies that can be applied in conjunction with the rolling stock strategies.

A microscopic railway simulation provides an accurate description of railway traffic by representing all details about the infrastructure (e.g. track gradients, curvature radii), the vehicles (e.g. mass, tractive-effort speed curves of the traction unit, braking rates), signaling (e.g. position and aspect of signals) and the interlocking (position of switches, interlocking rules to prevent conflicts at junctions) (Hansen and Pachl, 2008). Microscopic simulation allows for in-depth analysis of important timetable performance indicators, such as infrastructure occupation, feasibility, robustness and energy efficiency (Goverde and Hansen, 2013). Therefore, such simulations have primarily been used for assessing the quality of timetables or timetable rescheduling approaches, see e.g. Quaglietta et al. (2013), Schlechte et al. (2011), and Solinen et al. (2017). Our simulation framework makes use of the flexible microscopic railway traffic simulator EGTRAIN (Quaglietta, 2014), which features an API module that allows customization of built-in train control functionalities and the interface with external algorithms for real-time dispatching, as the one assessed in this research.

Experiments on a part of the Dutch railway network showcase the potential of decentralized dispatching approaches. Despite the lack of central control, it is possible to approximately meet the target frequencies of the lines in the network with a large degree of regularity and with only small delays. This also holds when crew is added into the mix, as long as we assume that all drivers are willing to work up to 2 hours longer than planned.

Summarizing, the contribution of this paper is two-fold. First, we propose two strategies for decentrally dispatching drivers along with the rolling stock. Secondly, we assess the performance of decentralized dispatching strategies using a microscopic simulation of a part of the Dutch railway network.

The remainder of this paper is structured as follows. In Section 2, we discuss the problem setting and the rolling stock and driver dispatching strategies. In Section 3, we discuss the simulation platform. In Section 4, we discuss the different performance measures. In Section 5, we discuss the results of a series of experiments. Finally, we conclude the paper in Section 6.

2 Problem description and dispatching strategies

In this section, we describe the problem we consider in this paper and discuss the rolling stock and driver dispatching strategies.

2.1 Problem description

In this paper, we consider the problem of decentrally operating a railway system. As timetables, rolling stock and crew schedules all require centralized control, this implies that we aim to operate the system without a centrally planned timetable, rolling stock and crew schedule. Instead, we use local policies that determine the next task incoming rolling stock and crew should perform. We assume that there is a line plan, specifying the lines and frequencies, that is known by all local dispatchers. Every line is operated in both directions. The objective of the local policies then is to execute this line plan as well as possible, i.e. the frequencies in the line plan should be met, delays should be avoided and the service should be regular.

We assume that the rolling stock is composed of self-powered train units and that there are no restrictions to the use of the rolling stock, so every piece of rolling stock can be used on every line. For the drivers, we assume that there are three constraints that should be taken into account: a break constraint, a planned end-of-duty constraint and a duty length constraint. The break constraint and the end-of-duty constraint are soft constraints, meaning that although it is undesirable to have drivers skip their breaks or work past their planned end-of-duty time, this is not strictly forbidden. We assume that drivers can take breaks at all stations. The duty length constraint is a hard constraint: it is strictly not allowed for a driver to operate any new trips after the driver has worked longer than a specified amount of time. Note that the planned end-of-duty time is earlier than the time at which the maximum duty length is reached. The duty length constraint is evaluated at the beginning of a trip using the minimum trip time. Hence, it may occur that a driver surpasses the maximum duty length while operating a trip because of delays. In such a case, the driver is allowed to finish the trip.

We also need to make assumptions with regards to the safety system to prevent decentralized strategies from causing deadlocks. To see why this is necessary, note that since train routes are not coordinated, such strategies could potentially lead to deadlock situations on single tracks or in station areas. On single tracks, this can simply be prevented by using an (electronic) token system. For station areas (or junctions), it is necessary that local traffic controllers are able to set a route for a train through the station to and/or from the platform, such that no other trains can cross the route until the train has either arrived at the platform or left the station.

2.2 Rolling stock dispatching

In this paper, we study rolling stock dispatching strategies that are proposed in Van Lieshout et al. (2020). For completeness, we explain them here as well.

The rolling stock dispatching strategies are used to determine the next service of a train when a train finishes a service. Hence, these strategies are only applied at the terminal stations of lines. At other stations, trains always continue their service with a dwell time that is as short as possible. The strategies comprise of two components: the *timing* component and the *turning* component. The timing component of a strategy determines the departure time of the train and the turning component determines the next line of the train.

There are two options for the turning component: *STAT* and *DYN*. STAT stands for static turning. When a strategy uses the STAT component, a train finishing a service of line l is instructed to perform a return trip of line l. In other words, if STAT is used, lines have dedicated vehicles. Conversely, when a strategy uses the DYN component, trains can be exchanged between lines. Then, a train finishing a service is assigned to the line with the earliest desired departure time, which is defined as the sum of the most recent departure time of the line and the desired interdeparture time of the line (e.g., 30 minutes for a line with frequency 2/h). Note that if all lines have the same frequency, an incoming train is always assigned to the line whose latest departure is the longest time ago.

There are also two options for the timing component: *ASAP* and *SYNC*. When a strategy uses the ASAP component, a train finishing its service is always instructed to depart as soon as possible. When a strategy uses the SYNC component, the departure time is determined based on the most recent departure time of the selected line. For example, if a train finishes its service at 09:15 and is assigned to a line with frequency 2 per hour and a most recent departure time of 9:05, the train is instructed to depart at 9:35, to meet the desired interdeparture time of 30 minutes. If instead, the most recent departure time would be before 8:45, the train is instructed to depart as soon as possible.

2.3 Driver dispatching

In this paper, we propose two strategies that can be used to dispatch drivers in a decentralized manner. Similar to the rolling stock strategies, these strategies determine the next service to be performed by a driver, whenever a driver finishes a service. Moreover, as the rolling stock strategies, the driver strategies also require little information and computation. The driver strategies use the concept of *availability score*. This score indicates to what extent a driver can perform the service within the labor regulations. The availability score is computed based on the characteristics of the service (departure time and destination) and also takes into account for example break and duty length constraints. The lower the availability score, the more labor constraints are (likely) violated. An availability score of 0 indicates that a driver cannot perform a service. The idea of the driver dispatching strategies is to swap drivers at terminals whenever someone with a higher availability score is available.

We next describe how the driver dispatching, in conjunction with the rolling stock dispatching, works in more detail. When a train finishes a service, the rolling stock dispatching strategy proposes a tentative next service. If either the driver currently on the train or any of the drivers that are present at the corresponding station is able to perform this service (i.e. has an availability score strictly larger than 0), the line and departure time are fixed. If none of the drivers is able to perform the service, the departure time and line are adjusted, until either a driver is available or all options are exhausted. Once the line and departure time of the next service are fixed, the crew dispatching strategy determines which driver should operate the service. By default, this is the driver that is currently on the train. If there is a driver at the station with a higher availability score, this driver is assigned to the service and the driver on the train stays at the station.

The availability score can be computed for any combination of a service and a driver. The score is based on a driver's last break time, planned end-of-duty time and crew base, and of the departure time and destination of the service. A lower availability score corresponds to a violation of a more important labor constraint. The exact interpretation of the availability score is stated in Table 1. Since drivers are dispatched decentrally and dynamically, it is not possible to predict with certainty whether performing a service will later lead to the violation of a constraint. Therefore, we propose two strategies that have different ways of performing this prediction. In the first strategy, OneStepAhead, only violations during the service are considered. For example, if a driver's planned end-of-duty time is 16:30, the OneStepAhead strategy will only give an availibility score of at most 1 to services that end later than 16:30, regardless of the destination of a service. In contrast, the TwoStepAhead strategy also takes into account the time required to travel from the destination of a service to the driver's crew base. Only, if a driver is able to return to his/her crew base before the planned end-of-duty time, this strategy will give an availability score larger than 1. This works similarly for the other constraints.

Table 1: Interpretation of the availability score

Availability Score	Performing the service...
0	... causes a violation of the duty length constraint
1	... causes the driver to work past his/her planned end-of-duty time
2	... causes a violation of the break constraint
3	... does not violate any constraints

We also need to specify when drivers are sent on to having a break or can sign off completely. In both strategies, any driver that is idle at a break station is assumed to be having a break. Furthermore, whenever a driver is present at his/her crew base and it is past the planned end-of-duty time, the workday of the driver is ended. If a driver is idling at a station other than his/her crew base and is no longer able to perform any services without violating the planned end-of-duty time (i.e. the availability score is always 0 or 1), we assume that the driver travels to the crew base as a passenger. The workday is only ended upon arrival at the crew base.

In principle, any rolling stock dispatching strategy can be combined with any driver dispatching strategy. However, the added flexibility of the DYN strategies is especially useful when drivers are also considered, as they allow a driver at the end of a shift to operate a train towards his/her crew base. On top of that, if there are lines that connect two stations that are not crew bases, a STAT strategy may be ineffective, since a driver operating a train on this line will at some point have to

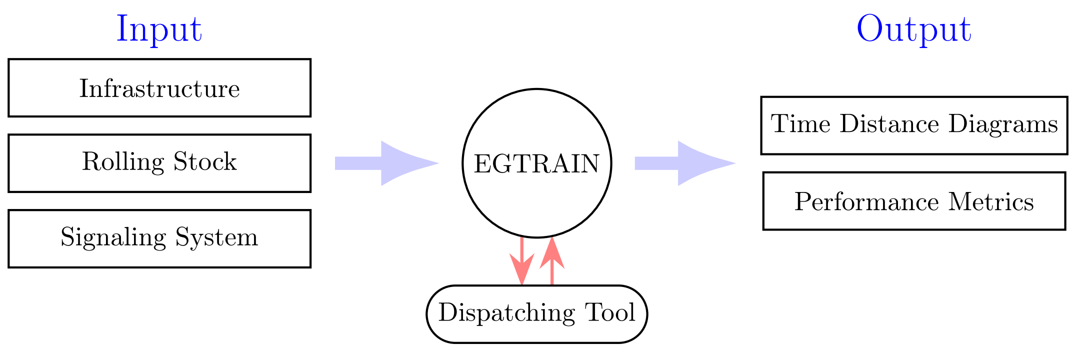


Figure 1: Visualization of the integrated platform.

abandon the train and travel to his/her crew base and it is unlikely that there is a driver available who can start operating the train on this line.

3 Simulation platform

We simulate the dispatching strategies using a platform that integrates a dispatching tool and a simulator and manages the continuous communication between these entities. The dispatching tool determines dispatching decisions based on incoming messages that specify departure and arrival times of trains and communicate these decisions back to the simulator. The simulator simulates the railway traffic that follows from the dispatching decisions and communicates all departure and arrival times back to the dispatching tool, which is in turn processed by the dispatching tool to make new dispatching decisions. Besides managing the continuous communication between the dispatching tool and the simulator, the platform also has an editor for setting up experiments. Such an experiment is sent to a server, which processes it by running the Simulator and dispatching tool. Once an experiment is finished, the realized time distance diagrams of all trains can be visualized, and a variety of metrics can be chosen to display for the assessment of one, or multiple simulation runs.

The simulation platform has a modular design and is, therefore, able to simulate any type of strategy using any type of simulator. In this paper, the dispatching tool determines dispatching decisions based on the decentralized rolling stock and crew strategies explained in the previous sections. Moreover, as a simulator, we use the microscopic simulator EGTRAIN. The platform is visualized in Figure 1. In the remainder of this section, we discuss this simulator in more detail.

3.1 EGTRAIN

EGTRAIN (Environment for the desiGn and simulaTion of RAIlway Networks) is a C++ object-oriented model for simulating railway operations at a microscopic level of detail by relying on time-driven processing of traffic events (Quaglietta, 2014). Input data are grouped within four main interacting modules, namely:

- The infrastructure module, which builds on a weighted directed-graph representation of the network where nodes represent physical infrastructure elements like switches, signals, balizes and station platforms while links are rail tracks connecting those elements. Node weights describe geographical coordinates of corresponding infrastructure elements, while link weights depict physical track characteristics such as gradients, speed limits and curvature radii.

- The rolling stock module, that collects physical and mechanical features of trains, including train masses lengths and car composition, as well as braking rates, tractive effort-speed curves and motion resistance coefficients.

- The signaling system module, which stores data about operational principles and rules of both the signaling and the interlocking system. Dependencies between signal aspects, speed codes of the Automatic Train Protection (ATP) system are modeled. Several signaling systems can be simulated ranging from traditional multi-aspect fixed-block signaling (e.g. the Dutch ATB/NS '54, the Italian BACC) to the three levels of the interoperable European signaling standard ERTMS/ETCS (Theeg and Vlasenko, 2009) for which the communication of Movement Authority and train position updates between trains and the RBC are specifically modelled. Additional functionality has been recently added to the signaling module to describe train operations under the next-generation signaling concept of Virtual Coupling (Quaglietta et al., 2020).

- The timetabling module, which contains data about the train schedule such as planned departure/arrival times and minimum dwell times at stations. This module also takes as input stochastic distributions of entrance delays and station dwell times to assess the impact of disturbances on planned operations. In this paper, trains are dispatched according to decentralized strategies instead of a timetable, so this module is not used.

The core of EGTRAIN simulates train movements by integrating Newton's motion formula over time. At each time step, the speed and position of trains are calculated based on track and vehicle characteristics and the status of the signaling system is updated accordingly to respect safety constraints. Output from the simulation consists of train diagrams (e.g. time-distance, speed-time), delay statistics, mechanical energy consumption and blocking time diagrams. EGTRAIN features an API module for customizing functions, modifying model parameters and interface simulated railway operations with external applications such as sensitivity analysis toolboxes or traffic rescheduling algorithms. The decentralized dispatching strategies presented in this research have hence been interfaced with the EGTRAIN API module to reschedule train services, in real-time during the simulation. The impact of different decentralized dispatching strategies are assessed in simulation in terms of relevant performance measures pertaining to both train services (i.e. frequency, regularity, and delay) as well as crew duty planning (e.g. number of violations to planned lengths of break and duty times of the train crew).

In the remainder of this section, we discuss relevant implementation details.

Signaling system

The signaling of the simulation consists of a three-aspect fixed-block signaling system that resembles the Dutch railway signaling. The control of single tracks is based on a token system, such that a single track can only be occupied by a train at a time. This implies that even trains traveling in the same direction are not allowed to cross the single track simultaneously. This approach also prevents the single track to be used by consecutive trains in the same direction, which would lead to the single track being used for a long time in the same direction. Such a strategy is applied because train services are not known in advance, due to the characteristics of the dispatching strategies.

Additionally, there are limitations at interlocking areas, since the simulation does not prevent the usage of conflicting routes, which might lead to deadlocks. To mitigate such a problem, we use a conservative approach where trains cannot enter station interlocking areas if trains are leaving the station, therefore preventing deadlocks. Similarly, trains are not allowed to depart while trains are entering the station. This applies to terminal stations only.

Maneuvers at terminal stations

At terminal stations, trains always depart from the platform where they have arrived. An minimum time of 5 minutes is required before a train can depart in the opposite direction.

Dynamic platform allocation

By default, the simulation uses static train routes, i.e. trains are dispatched to a specific platform at their destination. However, considering the characteristics of the dispatching algorithms, such static behavior is not desired. If a train approaches a station and the assigned platform is occupied, the train will have to stop and wait for it to become available. This leads to delayed trains and queues around terminal stations. A solution to mitigate this problem is accomplished by the introduction of a dynamic platform allocation. Whenever a train is approaching a terminal station and its assigned platform is occupied, the simulation checks the availability of other platforms and adjusts the train route accordingly, if a suitable route exists. Otherwise, the train waits until a platform is available. By implementing such a strategy, queues are reduced and the usage of platforms is improved.

Initial position of trains

For every experiment, trains always start at terminal stations. In some cases, the number of trains starting at a given station exceeds the number of platforms available. When that is the case, the second train assigned to a given platform only enters the station after the first train departs. A maximum of two trains can be assigned to a platform when defining the initial position of trains.

Timestep

For all experiments, a timestep of one second is used, which provides a high level of detail when computing train motion dynamics.

4 Performance measures

To measure the performance of the rolling stock and crew dispatching strategies, we use a set of performance measures. We use the three operational measures proposed in Van Lieshout et al. (2020) to assess whether the line plan is executed satisfactorily. These measures are *frequency, regularity* and *delay* and consider the realized frequencies of the lines, the regularity of interdeparture times of lines and the delays, respectively. Besides the operational measures, we also discuss how we assess the performance of our strategies with respect to the constraints for train drivers.

4.1 Operational measures

The metrics are defined for the operation of a line in one direction. Let h denote the target interdeparture time of the line in minutes (so the hourly frequency is $60/h$) and τ the minimum trip time from one terminal to the other in minutes. Let the departures be labeled as 1, 2, ..., n, with departure times (in minutes from the start of the simulation) $d_1 \leq d_2 \leq \cdots \leq d_n$ and realized trip times $t_1, t_2, ..., t_n$. We first define the average realized interdeparture time, which we denote as $\mathcal{H}$:

$$\mathcal{H} = \frac{1}{n-1}\sum_{i=1}^{n-1}(d_{i+1} - d_i)$$

Frequency: The frequency metric, denoted as $\mathcal{F}$, measures the realized frequency, relative to the target frequency (or equivalently, the average realized interdeparture time relative to the target interdeparture time):

$$\mathcal{F} = \frac{h}{\mathcal{H}}.$$

Regularity: The regularity metric, denoted as $\mathcal{R}$, measures to which extent the interdeparture times vary with respect to the average realized interdeparture time:[1]

$$\mathcal{R} = 1 - \frac{1}{(n-1)\mathcal{H}}\sum_{i=1}^{n-1}|d_{i+1} - d_i - \mathcal{H}|.$$

Delay: The delay metric, denoted as $\mathcal{D}$, captures the average delay of the line, measured

[1]This definition sligtly deviates from Van Lieshout et al. (2020). In this paper, the deviations are measures with respect to the realized average interdeparture time, whereas Van Lieshout et al., 2020 measure the deviations with respect to the target interdeparture time.

relative to the theoretical minimum trip time:

$$\mathcal{D} = \frac{\sum_{i=1}^{n} t_i}{n\tau}.$$

Note that all metrics are normalized, such that a value of 1.00 for all metrics indicates the scenario where the target frequency is exactly met, all interdeparture intervals are constant and there are no delays. To assess the performance of the decentralized strategies on the complete line plan, we take an unweighted average over all lines in both directions.

4.2 Crew measures

To analyze the performance of the crew dispatching strategies, we simply count how many times the break and duty length constraints are violated, and how often drivers need to work past their planned end-of-duty time.

5 Results

5.1 Instances

Network and line plan

For the experiments, we use a part of the Dutch railway network. This network is depicted in Figure 2. The network contains four lines, all of which should be operated with a frequency of 2 per hour. The largest station in this network is Ut (Utrecht Centraal), which also serves as the crew base in our experiments. The part between Dld (Den Dolder) and Brn (Baarn) is single track, with a passing loop at St (Soest).

The infrastructure input data for the microscopic simulation is built from a very detailed database of the Dutch railway network, provided by ProRail, the Dutch infrastructure manager. It was necessary to convert the original data from the database into the specific format of input data used by EGTRAIN. After the conversion, the model of the network is still detailed but with some approximations. These include, for example, the assumption of a fixed block section length

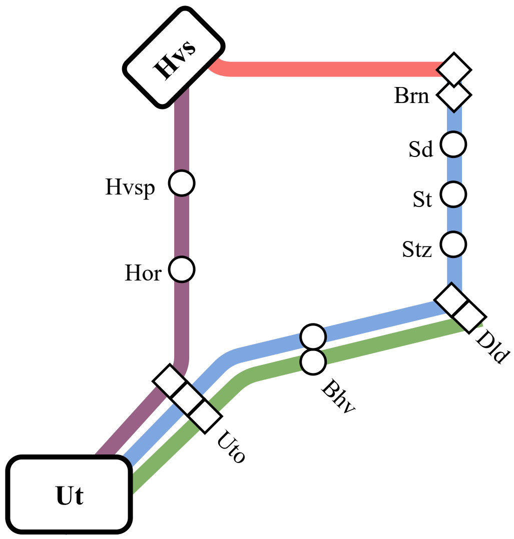


Figure 2: The network considered in the experiments.

and approximated track gradients and speed limits.

To compute the motion dynamics, we use the characteristics of a 6-wagon SLT train, which is an often-used rolling stock type in the considered area. The input data of rolling stock includes the tractive effort-speed curves and other characteristics, e.g. maximum speed, length and mass.

Finally, the default values of dwell times are based on the real timetable of train services running across the area.

Crew data and simulation duration

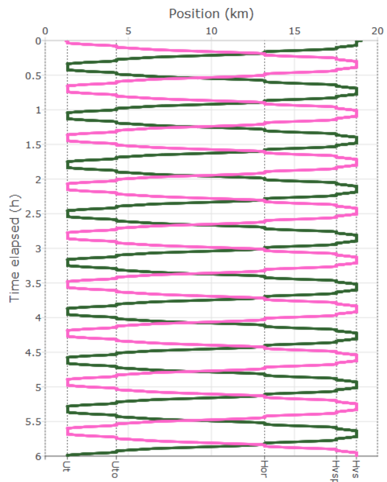
We use crew data that is based on crew schedules used by NS. The majority of duties of train drivers at NS can roughly be subdivided into morning shift duties, ending somewhere between 12pm and 2pm, and evening shift duties, starting somewhere between 12pm and 2pm. As it is interesting to simulate this period with many driver reliefs, we simulate a duration of 6 hours, from 10am until 4pm. Moreover, when we construct an instance with $2x$ drivers, we generate x morning shift drivers whose duties end between 12pm and 2pm and x evening shift drivers, whose duties start between 12pm and 2pm. The exact starting and end times are uniformly generated within this interval. For all drivers, Ut serves as the crew base. The planned duty lengths are all set equal to 8 hours. The maximum allowed working time is set equal to 10 hours. In other words, a driver is allowed to work at most 2 hours past his/her planned end-of-duty time. The maximum working time without a break is set equal to 4.5 hours.

5.2 Comparison of rolling stock strategies

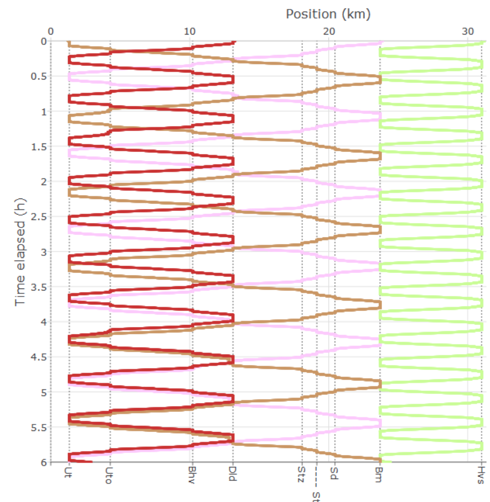
In the first experiment, we compare the performance of the four rolling stock dispatching strategies, without simulating the crew. Initially, we assume that there are six trains available in this region.

We first analyze the time-distance diagrams that visualize all train trajectories during the simulation. Figure 3 and Figure 4 present the time-distance diagrams for the four rolling stock dispatching strategies. The line Ut-Hvs is depicted in subfigures (a) and (c). The three other lines are depicted in subfigures (b) and (d). The different shades of gray in the diagrams represents a different train. When we compare the STAT strategies with the DYN strategies, it can be observed that in accordance with the definition of these strategies, when a STAT strategy is applied, trains stick with their initial line, whereas when a DYN strategy is applied, trains can switch between lines. Especially when the ASAP-DYN strategy is used, trains are often exchanged between lines, to serve the line that needs a departure most urgently. Note that there are no switches between different lines at station Brn, as this is prohibited by the infrastructure at that station. When we compare the ASAP strategies with the SYNC strategies, we find that the time-distance diagrams of ASAP-STAT and ASAP-DYN appear to be more cluttered and irregular compared to those of SYNC-STAT and SYNC-DYN. On the other hand, it appears that the ASAP strategies are able to achieve higher frequencies. In none of the diagrams, long delays or queuing of trains can be observed. Moreover, the time-distance diagrams do not give any signs of a long warm-up period required before a steady state is reached. Instead, the behavior of the system seems rather homogeneous over time.

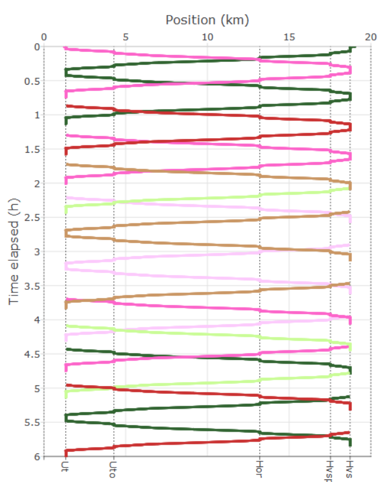
Figure 5 presents the operational measures obtained with the different strategies. This figure supports the observations made using the time-distance diagrams. The delay measure is very close to 1.00 for all strategies, indicating that there are hardly any delays. The frequency measure is slightly over 1.00 for the ASAP strategies and slightly below 1.00 for the SYNC strategies. This shows that the SYNC strategies lead to frequencies that are a bit below the target frequencies, while the ASAP strategies lead to frequencies above the target frequencies. This is caused by the fact that the ASAP principle instructs trains to depart as soon as possible, without regard to the desired



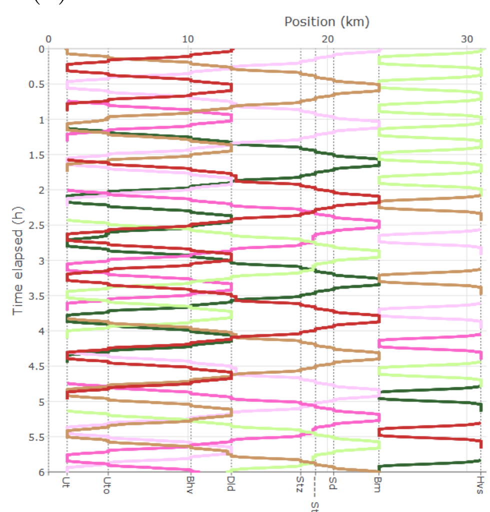
(a) Ut-Hvs with ASAP-STAT



(b) Ut-Brn-Hvs with ASAP-STAT

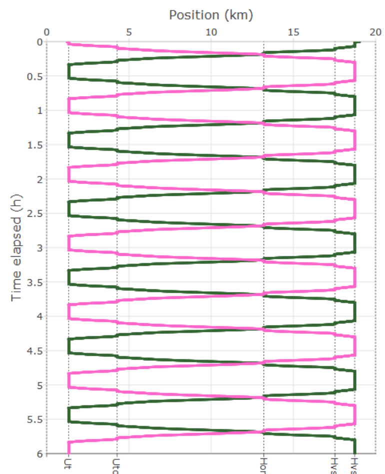


(c) Ut-Hvs with ASAP-DYN

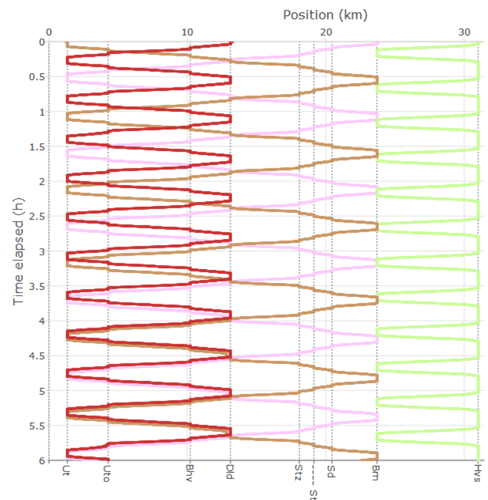


(d) Ut-Brn-Hvs with ASAP-DYN

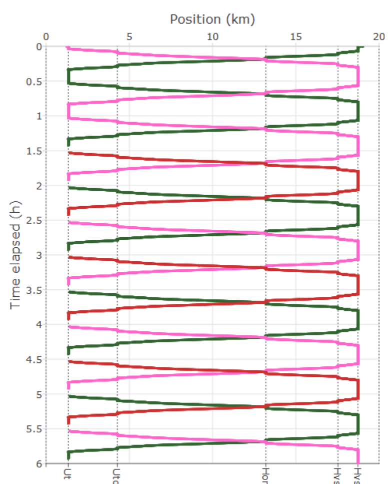
Figure 3: Time-distance diagrams obtained by simulating the ASAP strategies with six trains.



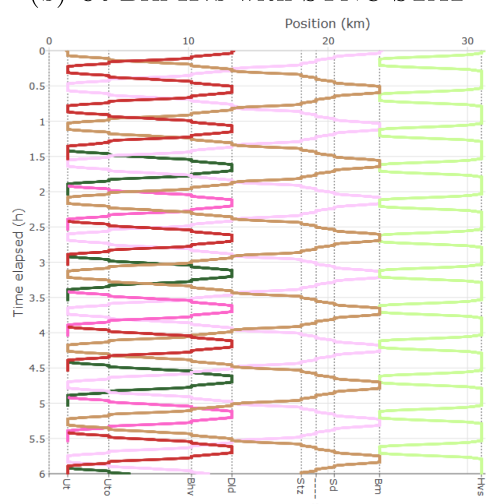
(a) Ut-Hvs with SYNC-STAT



(b) Ut-Brn-Hvs with SYNC-STAT



(c) Ut-Hvs with SYNC-DYN



(d) Ut-Brn-Hvs with SYNC-DYN

Figure 4: Time-distance diagrams obtained by simulating the SYNC strategies with six trains.

interdeparture. We find that the SYNC strategies perform better for the regularity measure, with values very close to 1.00. This confirms the observation that the services realized by these strategies are almost perfectly regular. The ASAP strategies score worse in terms of regularity, especially ASAP-DYN. There are only minor differences in the measures obtained with SYNC-STAT and SYNC-DYN. This could be caused by the number of trains available in these experiments, as in the absence of delays, six trains are sufficient to meet the target headways.

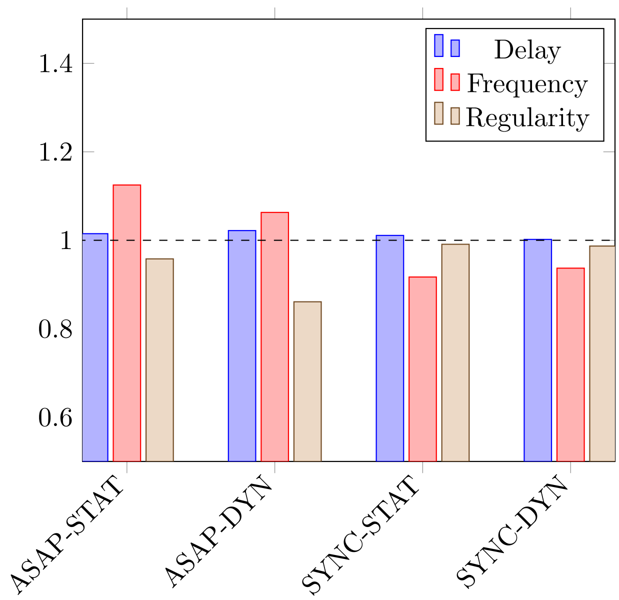


Figure 5: The operational measures obtained by simulating the strategies with six trains, averaged over all lines in the network.

Larger differences in the performance of the strategies become apparent when we analyze the measures per line, especially in terms of frequency. Figure 6 presents the frequency measure per line for the four strategies. The static turning principle can be seen to lead to a much larger dispersion in the frequency. For example, with the ASAP-STAT strategy, half the lines experience a frequency much larger than 1 (up to 1.5), and the other half experience a frequency smaller than 1. This occurs since the lines Ut-Hvs and Hvs-Brn are assigned more trains relative to their trip time. With the ASAP-DYN strategy, the differences in frequency between lines is much smaller, as trains are swapped between lines. To a lesser extent, the same holds for the SYNC-STAT strategy and the SYNC-DYN strategy. The dynamic turning principle hence leads to a more balanced division of resources over the lines.

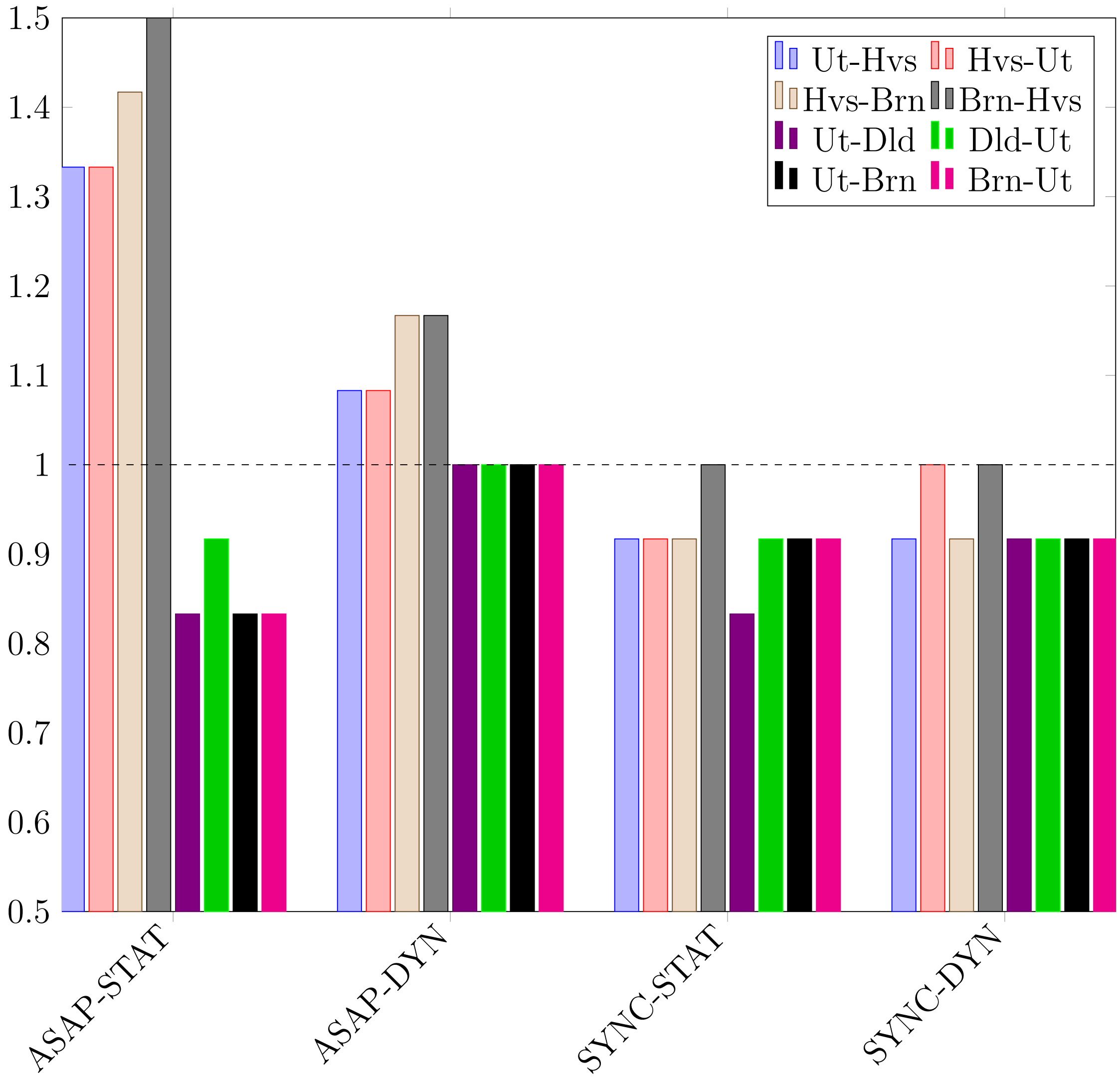


Figure 6: Frequency measure per line with six trains.

Varying the number of trains

Besides conducting the experiment with six available trains, we also repeat the experiment with four, five, seven and eight trains. Figure 7 presents the performance measures as a function of the number of trains. We find that with ASAP strategies every increase in the number trains translates to an increase in frequency. This is not the case with the SYNC strategies, where the frequency stops increasing after six trains. This aligns with the definition of these strategies, as the SYNC principle instructs trains to wait to meet the target interdeparture time, such that the frequency measure cannot be above 1.0 by definition. The difference in frequency between the STAT principle and the DYN principle again only becomes apparent when the frequency is analyzed per line, as the STAT principle leads to large differences in frequency per line, whereas the DYN principle leads to a more evenly distribution of services over the lines. How the number of the trains affects the other measures is less unambiguous. There appears to be a positive relationship between delay and the number of trains when one of the SYNC strategies is used. This delay can be attributed to the single track part between Dld and Brn, where the abundance of "slack" in the number of trains causes trains to have to wait for each other at the passing loop at Soest. There are no significant delays in the other parts of the network. As for the regularity, we find that all strategies have fairly a high regularity. The STAT strategies attain a higher or equal regularity than their DYN counterparts, except when there are eight trains. This is caused by the fact that the STAT strategies have a constant number of trains per line, leading to a higher regularity.

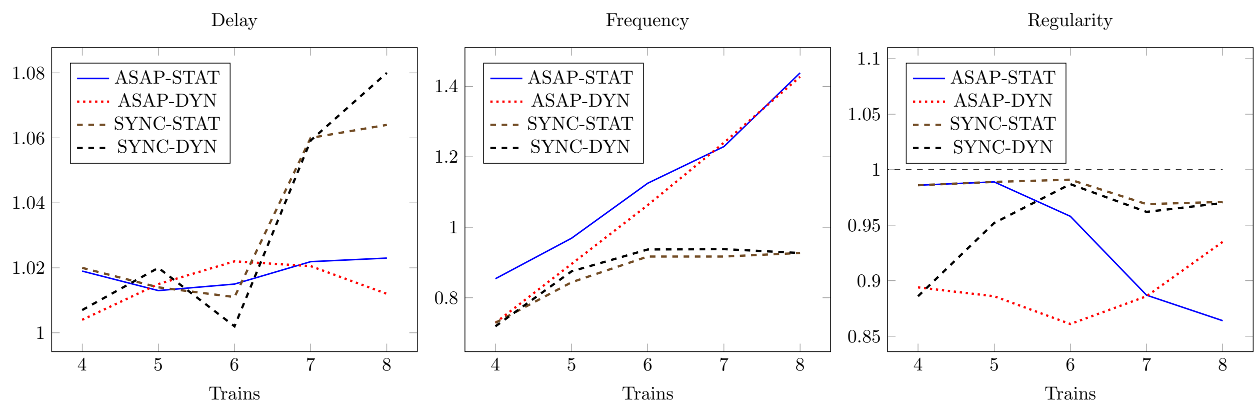


Figure 7: Performance measures with a different number of trains.

Higher line frequencies

Furthermore, we analyze the effect of increasing the frequencies of the lines in the network. Specifically, the frequency of the lines Ut-Dld, Ut-Hvs and Hvs-Brn is increased to 4 per hour. The frequency of the line Ut-Brn remains 2 per hour, as the single track cannot manage higher frequencies. We perform this experiment with ten trains. Figures 9 and 10 visualize the time-distance diagrams. The main finding is that the increased frequencies lead to a higher incidence of delays, which can be observed as vertical lines in the time-distance diagram. This happens occasionally when at the single track part of the network and also right before entering station Hvs. Still, there is no sign of queuing of trains and all delays remain relatively small. This is also reflected in the performance measures, presented in Figure 8. Especially the SYNC strategies experience larger delays compared to the case with lower line frequencies. With respect to the frequency measure, the ASAP strategies also outperform the SYNC strategies. On the other hand, the SYNC strategies do score much better on the regularity measure.

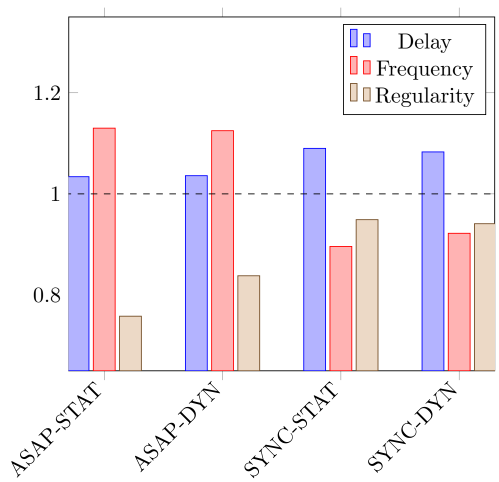
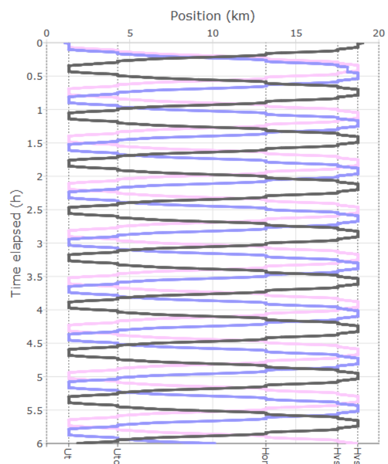
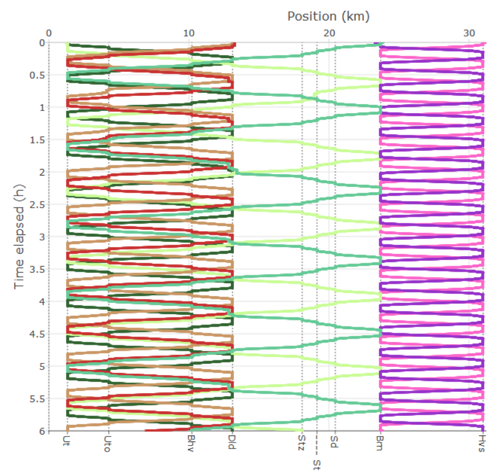


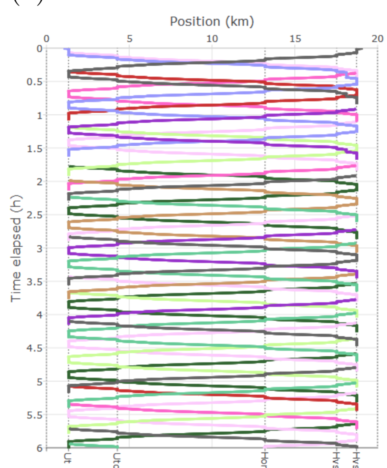
Figure 8: The operational measures obtained by simulating the strategies with ten trains, with increased line frequencies, averaged over all lines in the network.



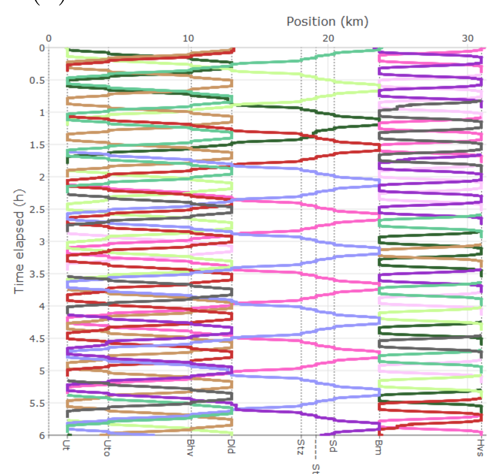
(a) Ut-Hvs with ASAP-STAT



(b) Ut-Brn-Hvs with ASAP-STAT

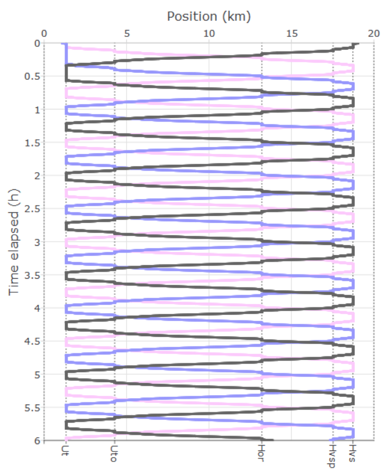


(c) Ut-Hvs with ASAP-DYN

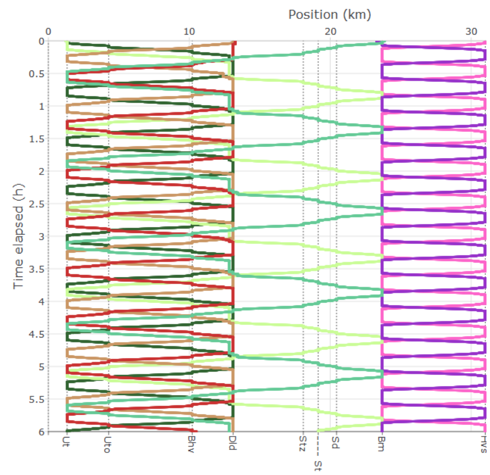


(d) Ut-Brn-Hvs with ASAP-DYN

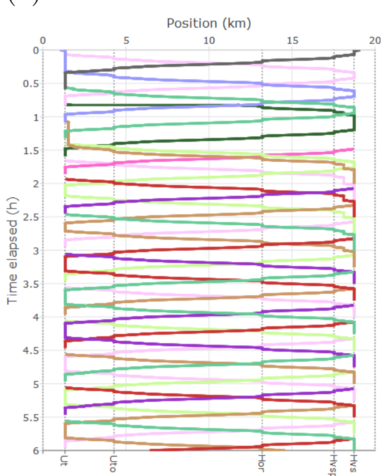
Figure 9: Time-distance diagrams obtained by simulating the ASAP strategies with ten trains, with increased line frequencies.



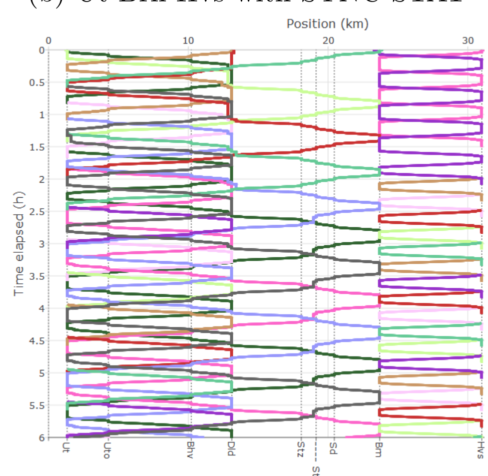
(a) Ut-Hvs with SYNC-STAT



(b) Ut-Brn-Hvs with SYNC-STAT



(c) Ut-Hvs with SYNC-DYN



(d) Ut-Brn-Hvs with SYNC-DYN

Figure 10: Time-distance diagrams obtained by simulating the SYNC strategies with ten trains, with increased line frequencies.

5.3 Comparison of crew strategies

In the second experiment, we compare the performance of the two crew dispatching strategies. As a static rolling stock strategy does not combine well with the flexible switching of drivers and the SYNC-DYN performed well without drivers, we choose the SYNC-DYN strategy for the rolling stock in this experiment. We use six trains, six drivers in the morning shift and six in the evening shift. We perform five runs for every setting, with different crew data.

First, we examine the impact of the inclusion of the crew dispatching in the simulation on the metrics. Figure 11 presents the operational measures for the OneStepAhead strategy, the TwoStepAhead strategy, and the case without driver dispatching. We observe that regardless of the crew strategy, the impact of including driver dispatching is small, with only minor differences in the obtained delay, frequency and regularity. Hence, we find that both strategies are successful in maintaining a high level of service. With the TwoStepAhead strategy, the frequency measure is even higher than without driver dispatching. A possible reason for this is that the TwoStepAhead strategy can instruct a train to leave before the desired departure time if that is required to avoid violating driver constraints.

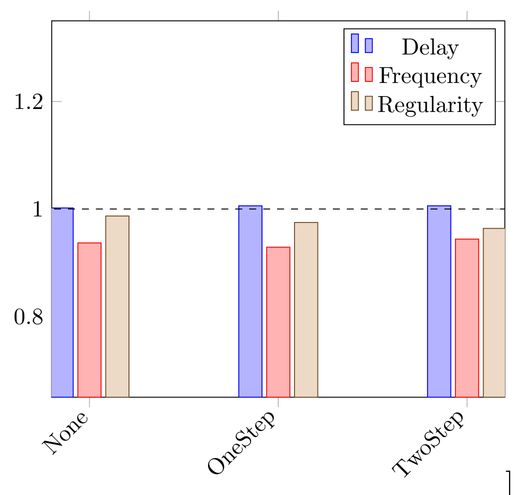
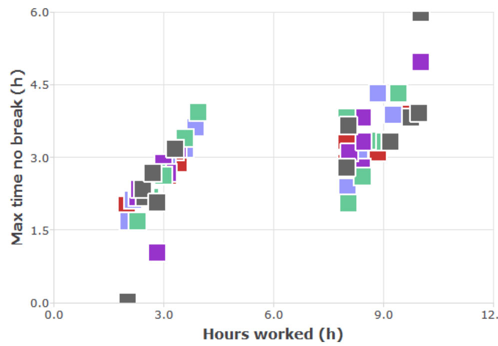


Figure 11: Performance measures with different crew dispatching strategies.

Next, we analyze the realized durations of the duty length of drivers and how long drivers have worked without having a break. Figure 12 visualizes these statistics for every driver in the five simulation runs. Every shade of gray corresponds to a different run. For both strategies, all

(a) OneStepAhead



(b) TwoStepAhead

Figure 12: Scatter plot of the realized duty duration and working time without break for the two strategies, with six trains and twelve drivers (six in the morning shift, six in the evening shift). The rolling stock strategy is SYNC-DYN. Every shade corresponds to a run.

points can be divided into two clouds, which correspond to the morning and evening shift drivers, respectively. Recall that the simulation starts at 10am, that the workday of the morning shift drivers starts between 4am and 6am and for the evening shift drivers between 12pm and 2pm. Hence, the left cloud corresponds to the evening shift drivers, who worked a couple of hours at most when the simulation ends. It is more interesting to consider the right cloud, corresponding to the morning shift drivers. We find that with both strategies, the majority of drivers needs to work between 8 and 9 hours. As the planned duty durations are 8 hours, this corresponds to at most 1 hour overtime. Moreover, for most drivers, the constraint that the maximum working time without a break is 4.5 hours is not violated. The difference between the strategies becomes apparent in the outliers, where we find that with the OneStepAhead strategy, five drivers worked more than 10 hours, including two drivers that worked over 11 hours. These drivers got stuck at stations other than the crew base Utrecht, and were required to travel back to Utrecht as a passenger. With the TwoStepAhead strategy, only two drivers worked more than 10 hours, but only with a maximum of 10 hours and 3 minutes, due to a driver operating a train that faced a delay. The TwoStepAhead strategy also leads to fewer violations of the constraint that drivers should have a break every 4.5 hours. Therefore, this indicates that the TwoStepAhead strategy is an effective strategy for avoiding severe violations of the break and duty length constraints.

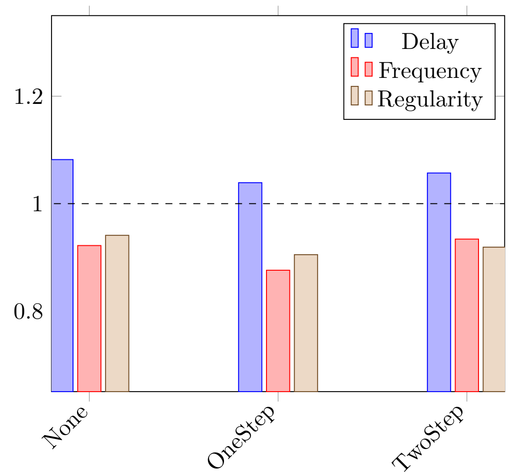
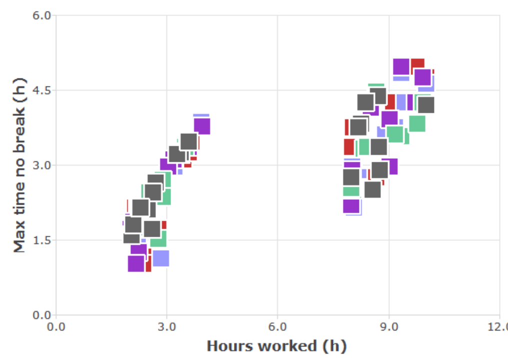


Figure 13: Performance measures with different crew dispatching strategies, with increased line frequencies.

(a) OneStepAhead

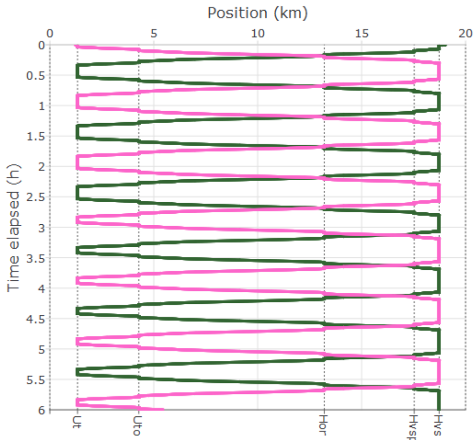


(b) TwoStepAhead

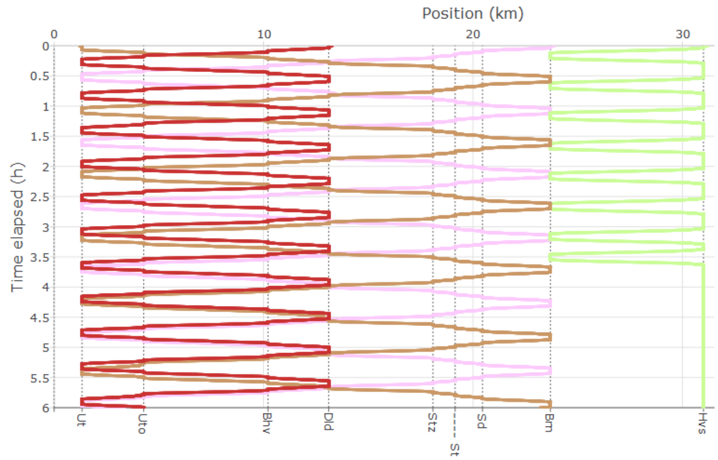
Figure 14: Scatter plot of the realized duty duration and working time without break for the two strategies, with ten trains and twenty drivers (ten in the morning shift, ten in the evening shift). The rolling stock strategy is SYNC-DYN. Every shade corresponds to a run.

Higher line frequencies

When we repeat the experiments with increased line frequencies of 4 per hour for all lines except Ut-Brn, we find the results presented in Figure 13 and Figure 14. We again find that the impact on the measures of including driver dispatching is small. Furthermore, the TwoStepAhead strategy results in fewer and less severe exceedances of the end-of-duty time of drivers and of the maximum time without a break than the OneStepAhead strategy.



(a) Ut-Hvs



(b) Ut-Brn-Hvs

Figure 15: Time-distance diagrams, with six trains and twelve drivers. The rolling stock strategy is SYNC-STAT and the crew strategy is TwoStepAhead.

Crew dispatching with a STAT strategy

In Section 2.3, we mentioned that rolling stock dispatching with static turning may be ineffective if crew is considered, since it may lead to drivers having to switch trains at stations that are not a crew base, where it is unlikely that there is a driver available to start operating the abandoned train. Experiments confirm that when crew is considered, static rolling stock strategies indeed perform badly. Figures 15a-b illustrate the issue. The figures show the obtained time-distance diagrams with SYNC-STAT as the rolling stock strategy and TwoStepAhead as the crew dispatching strategy. It can be observed that after about 3.5 hours, the train operating the Hvs-Brn line stops in Hvs and does not operate any more trips. The reason is that the driver originally operating this train, has to switch to the Ut-Hvs line at Hvs, in order to arrive at the crew base Ut before violating the duty length constraint. If the SYNC-DYN strategy would be used, the driver would stay on the train and simply operate a trip of the line to Ut. However, with the SYNC-STAT strategy, the train is not allowed to switch lines, such that the driver has to abandon the train and get on a different train to travel towards Ut. Moreover, as dispatching is done locally, the dispatcher at Ut is unaware of the driver shortage at Hvs. Of course, it is possible that the driver that abandoned the train at Hvs informs the dispatcher at Ut, but it would still take a long time for the replacement driver to arrive at Hvs.

6 Conclusion

In this paper, we tested the performance of decentralized strategies for dispatching rolling stock and drivers in a railway system. Such strategies could serve as a back-up plan when traditional dispatching approaches become infeasible due to disruptions. To analyze whether decentralized dispatching can be a viable alternative, we developed a simulation platform that is able to simulate dispatching strategies on a microscopic representation of the railway system. Experiments on a part of the Dutch railway system indicate that on small instances, easy-to-implement decentralized dispatching strategies can attain high performance, meeting target frequencies with a high degree of regularity and small delays. Strategies where trains are allowed to switch between lines attain the same average frequency as strategies where trains are fixed to lines. However, with these latter strategies, the frequency per line deviates much more strongly from the average frequency, indicating that dynamic switching leads to a more balanced performance. The advantage of dynamic switching strategies become even more clear when drivers are also considered: with static strategies, trains can be left without a driver because the driver needs to switch to a different line to travel to his/her crew base, which is avoided by using a dynamic strategy. Due to the complicatedness of microscopic railway simulation, we have only considered a relatively small instance with four lines and at most 10 trains. It would be interesting to scale up and investigate whether the performance of decentralized dispatching degrades in larger instances with multiple types of trains and lines.

Acknowledgements The integrated platform has been developed within the Software project of the bachelor Computer Science in the faculty of Science, Utrecht University (https://softwaregameprojecten.sites.uu.nl/ in Dutch, student team Shashi Chotkan, Teun Druijf, Simen van Herpt, Arvid Kater, Pieter Knops, Richard Koetschruyter, Lucas Meijer, Bas Nijmeijer, Ruben Vink, supervisor Michael Behrisch). This research was funded by NWO, the Dutch Research Council, as part of the research programme Complexity in Transport & Logistics (project number 439.16.111).

References

Dekker, M.M., R.N. Van Lieshout, R.C. Ball, P.C. Bouman, S.C. Dekker, H.A. Dijkstra, R.M.P. Goverde, D. Huisman, A.A.M. Schaafsma, and J.M. Van den Akker (2021). "A

Next Step in Disruption Management: Combining Operations Research and Complexity Science". *Public Transport*.

Goverde, R.M. and I.A. Hansen (2013). "Performance indicators for railway timetables". *2013 IEEE International Conference on intelligent rail transportation proceedings*. IEEE, 301–306.

Hansen, I.A. and J. Pachl (2008). *Railway Timetable and Traffic. Analysis, Modelling, Simulation.* Hamburg, Germany: Eurailpress.

Quaglietta, E. (2014). "A simulation-based approach for the optimal design of signalling block layout in railway networks". *Simulation Modelling Practice and Theory* 46, 4–24.

Quaglietta, E., F. Corman, and R.M. Goverde (2013). "Stability analysis of railway dispatching plans in a stochastic and dynamic environment". *Journal of Rail Transport Planning & Management* 3(4), 137–149.

Quaglietta, E., M. Wang, and R.M. Goverde (2020). "A multi-state train-following model for the analysis of virtual coupling railway operations". *Journal of Rail Transport Planning & Management* 15, 100195.

Schlechte, T., R. Borndörfer, B. Erol, T. Graffagnino, and E. Swarat (2011). "Micro–macro transformation of railway networks". *Journal of Rail Transport Planning & Management* 1(1), 38–48.

Solinen, E., G. Nicholson, and A. Peterson (2017). "A microscopic evaluation of railway timetable robustness and critical points". *Journal of Rail Transport Planning & Management* 7(4), 207–223.

Theeg, G. and S. Vlasenko (2009). "Railway signalling & interlocking". *International Compendium. Hamburg, Eurail-press Publ* 448.

Van Lieshout, R.N., P.C. Bouman, and D. Huisman (2020). "Determining and evaluating alternative line plans in out-of-control situations". *Transportation Science* 54(3), 740–761.

Van Lieshout, R.N., P.C. Bouman, M. Van den Akker, and D. Huisman (2021). “A Self-Organizing Policy for Vehicle Dispatching in Public Transit Systems with Multiple Lines”. *Transportation Research Part B: Methodological* 152, 46–64.