EETTlib – Energy-Efficient Train Timetabling Library

Andreas Bärmann[1], Patrick Gemander[2], Lukas Hager[1], Frederik Nöth[3] and Oskar Schneider[2]

[1] Andreas.Baermann@fau.de
Lukas.Hager@fau.de
Lehrstuhl für Analytics & Mixed-Integer Optimization,
Department of Data Science / Department Mathematik,
Friedrich-Alexander-Universität Erlangen-Nürnberg,
Cauerstraße 11, 91058 Erlangen, Germany

[2] Patrick.Gemander@fau.de
Oskar.Schneider@fau.de
Gruppe Optimization,
Fraunhofer-Arbeitsgruppe für Supply Chain Services SCS,
Fraunhofer-Institut für Integrierte Schaltungen IIS,
Nordostpark 93, 90411 Nürnberg, Germany

[3] Frederik.Noeth@vag.de
Referent Technik und Innovation,
Ressort Technik & Marketing,
VAG Verkehrs-Aktiengesellschaft,
Südliche Fürther Straße 5, 90429 Nürnberg, Germany

Abstract

We introduce *EETTlib*, an instance library for the Energy-Efficient Train Timetabling problem. The task in this problem is to adjust a given timetable draft such that several key indicators related to the energy consumption of the resulting railway traffic are minimized. These include peak power consumption, total energy consumption, loss in recuperation energy, fluctuation in power consumption as well as weighted versions of these optimization goals. To this end, the departure times of the trains can be shifted slightly, and their velocity profiles on each trip can be modified, both according to which degrees of freedom the timetable planner allows. We provide real-world data originating from two research projects in this field, one with Deutsche Bahn AG, the most important railway company in Germany, the other with VAG Verkehrs-Aktiengesellschaft, the operator of public transport in the city of Nürnberg, Germany. In the first case, we provide 31 problem topologies representing local, regional and national subnetworks of German railway traffic. The second data set is from the underground system in Nürnberg, with timetable drafts corresponding to 12 different traffic scenarios. In both cases, our library contains representative data on the relevant operational constraints related to timetable construction such as dwell times, headway times or connection times. Furthermore, our instance library contains the data to support various possible choices for the objective function with respect to energy-efficiency. This results in several hundred benchmark instances for the EETT problem overall, which can be used by the scheduling and timetabling community to improve their models and algorithms. Many more instances can easily be constructed out of these by varying problem parameters such as the allowable shift in departure time or the considered planning horizon. The resources of our library can be found under `https://www.eettlib.fau.de`.

Keywords: Benchmark Library, Railway Timetabling, Energy Consumption, Project Scheduling Problem, Clique Problem, Multiple-Choice Constraints

1 Introduction

Railway timetabling has been one of the most actively researched applications of discrete optimization for the last 30 years. On the one hand, this is because it belongs to the central problems in the planning chain for railway operations. Indeed, it is one of the major revenue factors for a railway company to offer an effective and efficient transportation schedule, both in passenger and in freight traffic, and many subsequent planning steps depend on the resulting timetable. On the other hand, it leads to structurally interesting mathematical optimization problems with rich combinatorial and graph-based structures. Exploitation of these structures is key to developing well-performing optimization algorithms, both in terms of computation time and solution quality. We refer to [CT12, Lie08] for an overview of modern modelling and algorithmic approaches in timetable optimization.

For the applied optimization community as a whole, it is very important to have access to high-quality benchmark instances from the studied field of application. Prominent examples of available benchmark libraries include the Travelling Salesman Problem Library (TSPLIB, [Rei91]), the Survivable Network Design Library (SNDlib, [OPTW10]) the Project Scheduling Problem Library (PSPLIB, [KS96]), the library on Steiner problem in graphs (SteinLiB, [KMV01]), the Mixed-Integer Programming Problem Library (MIPLIB, [GHG+21]) or the benchmark data set for Liner Shipping Network Design (LINERLIB, [BAP+14]). Such libraries have great impact on the development of efficient optimization models and algorithms. For example, the MIPLIB is the de facto gold standard for improving and benchmarking MIP solver codes. However, in the very active research field of timetable optimization, one of the most impeding issues is the lack of high-quality real-world data. To this date, only few resources are available in the public domain when it comes to complete and well-curated data sets which can serve as a basis for developing and evaluating timetabling models and algorithms. To the best of our knowledge, there are only two libraries providing real-world railway timetabling data so far. The first one is the Train Timetabling Problem Library (TTPlib, [EKS+08]), which contains data on a certain region in central Germany (the city triangle Hannover, Kassel, Fulda – HAKAFU, for short). Secondly, there are 9 timetabling instances from public transportation networks available as part of the software toolbox for Integrated Optimization in Public Transportation (LinTim, [SAG+22]).

Contribution Our present work contributes to closing the described lack in publicly available research data for railway timetabling problems. To this end, we introduce the *Energy-Efficient Train Timetabling library* (*EETTlib*), which contains real-world benchmark instances based on original data from two prominent railway companies in Germany. These are *Deutsche Bahn AG*, Germany's largest railway operator, and *VAG Verkehrs-Aktiengesellschaft* (VAG), the operator of the underground system in the city of Nürnberg (in English: Nuremberg), Germany. In this library, we focus on a special kind of timetabling problem which has become more and more relevant over the past few years, namely *energy-efficient train timetabling*. While in the beginning, timetable optimization has been mostly addressed from the point of view of time-efficiency, over the last decade, the energy-efficiency of the timetable has received increased attention as the recent, comprehensive survey [SGK17] shows. This is natural, since railway traffic is among the big energy consumers globally, with a cumulative consumption of about 290 TWh of electricity, which corresponds to more than 1% of worldwide electricity demand according to [Int20]. Furthermore, traction energy is the most important cost factor in the electricity bill of a railway undertaking. Via the timetable, railway companies have huge leverage over their electricity costs. This is due to the pricing schemes according to which large consumers of electricity are typically charged by electricity providers. They consist of a price component proportional to overall electricity consumptions of all trains, responsible for about 75% of the electricity bill, and a load-dependent component with a cost factor for the peak power consumption drawn from the railway power grid at any point during the billing period, typically accounting for around 25% of the total cost. Via energy-efficient timetabling, a railway company can decrease both cost components significantly.

Over the past eight years, we have compiled vast amounts of data collected in various timetabling research projects with our two application partners. During this time, we have carefully curated the raw data provided by them and, what is crucial to the scientist, we have extracted exactly those components of the data which are actually necessary to feed the optimization models. This means, in particular, separating the important from the unimportant. From our project with Deutsche Bahn AG, the largest railway company in Germany, we can provide data for the core operational constraints making up a feasible energy-efficient timetable adjustment based on their real-world passenger timetable as well as simulated data on the power consumption of the trains. In the case of VAG, we were able to go significantly further. Here, we could compile a complete data set on all operational requirements required for a practically realizable timetable for the Nürnberg underground network. Moreover, we have performed all necessary estimations and conversions between units of power consumptions of the trains based on actual, physical long-term measurements. This way, we can provide all this data in an easily accessible format which can be read in by simple routines. With their consent, we can make this data available to the public. Altogether, our library includes energy-efficient timetabling benchmark instances for 31 subnetworks from German local, regional and national railway transport – including a very large instance comprising the entire passenger traffic operated by Deutsche Bahn AG – as well as 12 traffic scenarios from underground transport in the German city of Nürnberg, provided by VAG. All benchmark instances are freely available for download on our EETTlib homepage under `https://www.eettlib.fau.de` and can be used under the CC-BY-SA-4.0 license (see Appendix A for details). Finally, we would like to point out that energy-efficient train timetabling (EETT) is actually a special case of the resource-constrained project scheduling problem (RCPSP); see [SZ15] for an extensive introduction. Therefore, we hope that the real-world problem instances in EETTlib will be of interest to the RCPSP community as well.

Structure of the Article The remainder of this article is structured as follows. In Section 2, we give a summary of the two research projects from which the benchmark instances in EETTlib stem and highlight the importance of energy-efficient timetabling to a railway provider. Section 3 describes the data format used in EETTlib to encode instances of the energy-efficient train timetabling problem, comprising both timetabling and energy consumption data. We also introduce all constraint types which are incorporated in the timetable optimization models we use in our research work with Deutsche Bahn AG and VAG. Furthermore, we explain parsing routines written in Python which allow for straightforward access to our chosen data format. In Section 4, we discuss several important optimization goals in energy-efficient timetabling and the way to model them using the data from EETTlib. Section 5 contains an overview of the problem instances themselves, for which some reference computational results are shown in Section 6. We then give our conclusions in Section 7. The article is rounded off with several appendices featuring additional information. Appendix A explains the licensing of the instances in EETTlib. In Appendix B, we define the mathematical models we use to solve the EETT problems posed by our partners. They are based on the clique-problem with multiple-choice constraints which is described and investigated in more detail in the references we list there. Finally, Appendix C contains an in-depth description of the instances, including their real-world sources and the assumptions we took when compiling them.

2 Two Real-World Projects in Energy-Efficient Timetabling

In energy-efficient train timetabling, the task is to create or adapt a schedule for a railway system such that the resulting traffic leads to a beneficial energy consumption pattern. This may involve different targets according to the type of system considered, the technical and operational requirements which need to be taken into account as well as priorities of the company and its planners. In the following, we present two projects which Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) and Fraunhofer-Institut für Integrierte Schaltungen (IIS) have been working on jointly in the area of energy-efficient timetabling together with industrial partners. The insights and the data collected in these two projects form the basis for the timetabling instances we publish here in EETTlib. However, as we point out, they can be used in much more general contexts, comprising EETT problems in different countries and companies.

2.1 Minimizing Peak Power Consumption in the German Railway Network

Together with *Deutsche Bahn AG*, we studied minimizing peak power consumption in the German railway network as part of research project *E-Motion* (see [Pro22]) funded by the German Ministry of Education and Research (BMBF). Our intention here was to bring down the 15-minute average peaks in power consumption cumulated over all trains running in the system. This is a major cost factor in the electricity bill of a railway company in Germany. Besides the cost of the total energy consumption of the trains, the railway companies are also charged for their maximum power consumption, either over an annual or a monthly scope. This fee is determined by the maximum average peak power consumption of all trains over any 15-minute interval in the billing period. This average consumption is calculated by summing up the measured power consumptions of all trains occurring during this 15-minute interval and then dividing by the length of the interval. To be precise: it is the natural four 15-minute intervals of the clock which count, not any arbitrary 15-minute interval.

The railway company then pays a certain rate per kilowatt, which is multiplied with the highest such average value occurring over the whole month or year, depending on the supply contract. The reason for including a load-dependent charge – which is usual for contracts with big consumers of electricity – is that it covers the costs for the layout of the infrastructure. Further, railway companies are rewarded for feeding back energy to the catenaries with a fixed price per kilowatt-hour of recuperated braking energy. It is typically somewhat lower than what they pay to buy the energy. However, recuperated energy which cannot immediately be used by other trains is lost, such that a synchronization of braking trains with accelerating trains is necessary. At the same time, it is an important goal to ensure stability in the railway power supply system. To this end, it is also necessary to keep instantaneous peaks in consumption, lasting up to several seconds, under control. Details about the technical background and typical pricing schemes can be found in [BMS21].

A way to reduce the power-dependent energy costs of a railway company can be to adjust the timetable such that simultaneous departures in overloaded time windows are desynchronized. This can already be achieved by slight shifts in the departure times of the trains by few minutes in both directions – making use of remaining degrees of freedom in the timetable draft before finally publishing it. Figure 1 shows the effect this can have on the power consumption profile of the railway company at the example of a local subnetwork around the city of Nürnberg in the German state of Bavaria (cf. instance *Nürnberg* in Section 6).

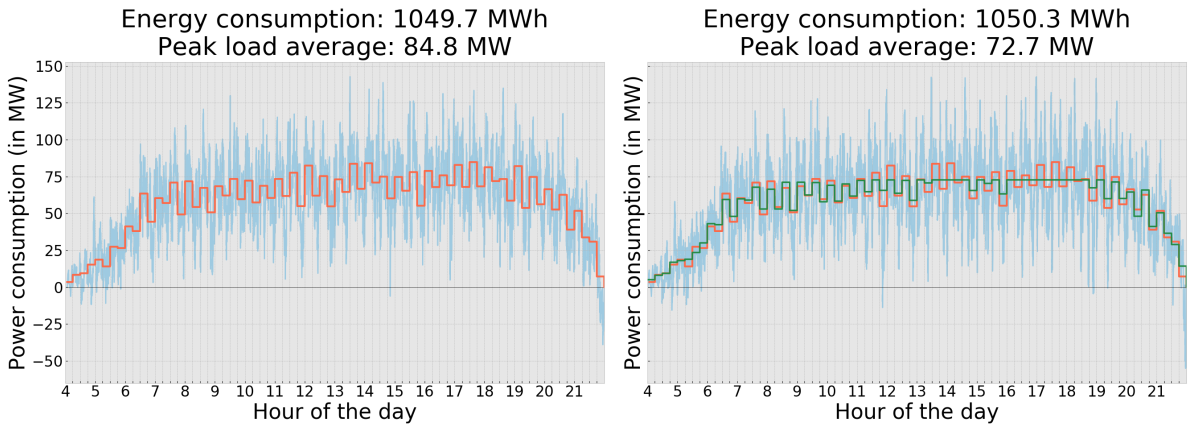

Figure 1: Power consumption profile for traffic in the local subnetwork around Nürnberg, Germany (instance *Nürnberg*, cf. Section 6) with 951 trains between 4 a.m. and 10 p.m., before (left) and after (right) optimization – instantaneous consumptions in *blue*, 15-minute averages in *red*/*green*. Peak power consumption decreases by 14% as a result of the optimization.

It shows the power consumption profile of all regional trains running in the state between 4 a.m. and 10 p.m. – both before (left) and after (right) an optimal adjustment of the timetable. In *blue*, we see the combined power consumption of the trains in each second, in *red*/*green* we see the resulting 15-minute averages. For this instance, timetable adjustment can balance power consumption throughout the whole day, where peak consumption can be reduced by 12.1 MW or more than 14%. According to the official price sheet by DB Energie GmbH for 2022 (see [DB 22]), a railway company is charged 105.02 € per kilowatt in peak demand over a given year. The 84.8 MW in peak consumption in the unoptimized timetable would thus amount to a power-dependent cost of 8.86 million € for the current year of planning, compared to 7.65 million € for the 72.7 MW in the optimized timetable, which represents major savings for this local subnetwork alone. This result was achieved shifting the departures of the trains by at most $\pm$ 3 minutes around their currently-planned departure times.

2.2 Minimizing Energy Consumption in a German Underground Network

In addition to avoiding high peak consumptions and making efficient use of recuperated braking energy, the energy-efficiency of a railway system is significantly influenced by the manner in which the trains are driven. Thus, a significant reduction in energy consumption can be achieved by choosing energy-efficient velocity profiles. This includes making use of pure rolling phases, the so-called *coasting*, as far as possible. A train consumes no traction energy at all in this phase and, due to the low rolling friction, only slowly loses speed. In Figure 2, the effects of choosing between different driving modes of a train are shown schematically for a train in an underground network. These data show that by slightly slowing down the fastest possible velocity profile on a given track, the train may consume up to 30% less in energy. In this respect, it is especially beneficial to extend the coasting phases of the train as much as possible.

Altogether, choosing the optimal velocity profile for each train on each *leg* (= timetabled run between two adjacent stations) with respect to given total line travel times entails a huge leverage for bringing down the consumption of the overall underground system. This finding motivated our joint research project *Energy-Efficient Timetable Optimization* with *VAG Verkehrs-Aktiengesellschaft*, the local operator of public transport in the German city of Nürnberg (see [Pro21]). The aim is to create the necessary flexibility in the timetable to enable choosing the best-possible velocity profile on each leg. In Figure 3, we exemplarily show for a Sunday

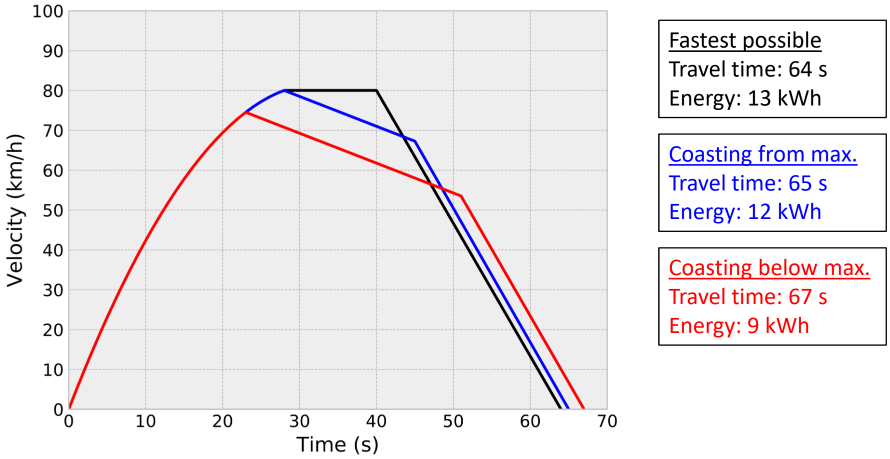

Figure 2: Schematic representation of the effect of choosing between one of the three driving modes "always driving as fast as possible" (*black*), "accelerating to maximum velocity followed by coasting" (*blue*) and "accelerating to below maximum velocity followed by extended coasting" (*red*) on a sample underground leg

the energy-saving potential in the underground system of Nürnberg by energy-efficient timetabling.

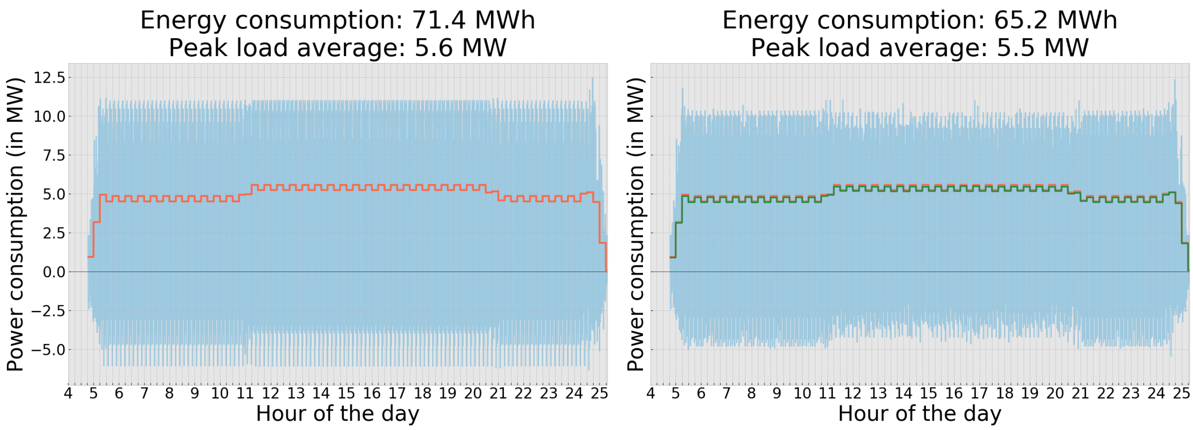

Figure 3: Power consumption profile for the railway traffic in the underground system of Nürnberg, Germany (instance *sunday, all lines*, cf. Section 6) with 841 trains for the whole day, before (left) and after (right) optimization – instantaneous consumption in *blue*, 15-minute averages in *red*/*green*. The optimization decreases total energy consumption by almost 9%.

We see that total energy consumption is reduced from 71.4 MWh to 65.2 MWh, to which both energy-efficient driving and increasing the use of recuperated braking energy contribute (note that the negative area under the *blue* curve is reduced, which means that less braking energy is wasted). Overall, there is a considerable potential not only for energy saving, but also for cost saving; for an operator like VAG, it can be a 6-figure amount in Euros every year. A more detailed description of these results can be found in [BGM+21a].

2.3 Representativeness of the Instances in EETTlib

This benchmark library comprises data sets compiled in two different research projects on energy-efficient timetabling in railway transport together with industry. Reducing the consumption of electricity via timetabling has been studied extensively in the literature over the past years; for example, see [YBG19, WBGC22, RDLLP+17, SG97, ZRH+17, CZ17, YNLT14, SWM+18, MGM20], among others. Indeed, these are all examples for research work that could have benefited from using our library to test the methods developed there on real-world data, as indicated in the following.

The minimization of maximum power peaks over 15-minute intervals as described above is a commonly studied optimization goal in train timetabling all across Europe; for example, see [YBG19, WBGC22] for the Dutch railway system. On a global scale, the length of the averaging intervals – or the peak load pricing scheme in general – may depend on the considered country and railway system. For instance, in Australia it is common to subdivide the day into differently priced time windows representing typical power demand scenarios (peak, shoulder and off-peak – in decreasing order of price per kilowatt-hour of energy drawn; see [Wri22]). We refer to [RDLLP+17] for an overview and a study of the effects of different pricing schemes on optimal timetabling patterns. The data we provide with EETTlib can easily be used for any time frame of averaging or for other pricing schemes relevant to peak load reduction. This also applies to reducing peak demand in order to stabilize railway power supply, as studied in [SG97]. Here the authors focus on minimizing instantaneous, i.e. second-wise, peak loads for the underground system of Montréal, Canada.

Apart from peak minimization, it is also an oft-considered goal to reduce the total energy consumption induced by a railway timetable, as it is performed, for instance, in [ZRH+17] and [CZ17]. Finally, other objective functions, such as minimizing passenger waiting times are considered in [YNLT14] and [SWM+18], while [MGM20] provides a model to reduce carbon emissions. For all the cases modelled in these works and many others, the data in our library can be used as well – possibly coupling it with data from other sources in order to incorporate further objectives and constraints which are specific to a certain application.

3 Data Format and Constraints for Feasible Timetable Adjustments

We now describe the data format for the problem instances available in EETTlib. The chosen data format represents all necessary input to solve the two real-world energy-efficient timetabling problems outlined in the previous section as well as many more related problems. It has been designed to make it as easy as possible to use it as input for an optimization model. In particular, it is not necessary any more to go back to the much more complex raw data with which we were provided by our industry partners. The data comprises the initial timetable, the power consumption profiles of the trains, a description of the rules for a feasible timetable adjustment and the solutions referred to in Section 6. We also provide routines in Python to parse the instances.

3.1 Initial Timetable and Adjustments

The initial timetable and its possible adjustments in a problem instance are given as the comma-separated-value (CSV) file *timetable.csv*. Each row in this file represents a single *leg*, i.e. one planned departure of a given train from a specific station to the next. A row consists of the unique identifier *leg_id* together with additional leg data. Namely, it features the *train_id* of the corresponding train, the *start_station_id* and *end_station_id* of the stations the leg departs from and arrives at and the *track_id* of the track being used. Further, the row includes the *nominal_departure_configuration*, which encodes the departure time and travel time for this leg according to the initial timetable, and finally it contains all alternative *departure_configurations*, i.e. all other combinations of departure times and travel times to choose from for a given leg.

The departure configurations are represented as strings of the format j_d_p, where j is the departure time of the leg in seconds since the beginning of the planning horizon, d is the time in seconds required to travel to the next station and p is the *profile_id* of the power profile used by the train on this leg. Each leg has exactly one nominal departure configuration, which together with those from the other legs form the initial timetable to be optimized. On the other hand, a leg may have multiple alternative departure configurations one can choose from, representing the possible timetable adjustments for this leg. The different possible departure configurations of a leg are separated by a whitespace character, and the nominal departure configuration is always included among the departure configuration alternatives. Table 1 summarizes the columns as well as the data types used in a timetable file and gives an example value for each column.

Column name	Data type	Example value
leg_id	int	1
train_id	int	1
track_id	int	1
start_station_id	int	1
end_station_id	int	2
nominal_departure_configuration	str	"17285_71_4489"
departure_configurations	str	"17285_67_5635 17285_71_4489 ..."

Table 1: Timetable data columns

3.2 Power Profiles

Similar to the legs in the timetable, each profile which can be chosen for a certain leg is stored as a row in a CSV file with two columns, called *profile.csv*. A row in this file contains the unique identifier *profile_id* of a profile together with the field *power_consumptions*, which is a value table with one power consumption value for each second of the trip. This value table is represented as a string consisting of floats which are rounded to three decimals and separated by whitespace characters. Each float represents the average power consumption of the train over one second in megawatts, where the first value represents the consumption during the second the leg departs. Accordingly, a profile corresponding to a travel time of t seconds will consist of t values, where negative values represent the potential for recuperated braking energy. Note that this potential can only be realized if the legs are suitably synchronized. Table 2 states the data and data types occurring in a profile as well as example values.

Column name	Data type	Example value
profile_id	int	5635
power_consumptions	str	"0.225 0.422 0.675 0.939 1.255 ..."

Table 2: Profile data columns

3.3 Constraints

The final building block of an instance is the set of timetabling constraints which need to be fulfilled in a feasible solution These constraints are stored in a JavaScript Object Notation (JSON) file named *constraints.json*. Table 3 gives an overview of all constraint types together with the data types used to represent them. The remainder of this section consists of dedicated paragraphs offering more detail on each type of constraint.

Constraint name	Key	Data type	Example value
headway_time_constraints	first_leg_id	int	3131
	second_leg_id	int	3157
	min_headway_time	int	100
single_track_headway_constraints	first_leg_id	int	6069
	second_leg_id	int	458
dwell_time_constraints	first_leg_id	int	1
	second_leg_id	int	2
	min_dwell_time	int	20
terminal_turnaround_constraints	first_leg_id	int	206
	second_leg_id	int	6149
	min_turnaround_time	int	420
connection_constraints	first_leg_id	int	8846
	second_leg_id	int	3110
	min_connection_time	int	180
	max_connection_time	int	240
	connection_type	str	"arrival_to_departure"
recuperation_subnets	subnet_id	int	1
	track_ids	List[int]	[1, 2, 3, ...]

Table 3: Constraint data dictionaries

Headway Time Constraints

```
{"first_leg_id": 3131,
 "second_leg_id": 3157,
 "min_headway_time": 100}
```

Any two trains which consecutively use the same track in the network in the same direction need to maintain a safety distance. For each such case, the instance contains a *headway time constraint*. Its field *min_headway_time* contains the time in seconds which is required as a safety buffer between the two trains represented by the first and second leg stated. According to our timetable adjustment model, the temporal order of departure between these two legs must not change. In other words, our model considers the order in which trains pass the tracks in the network as fixed. This is only a slight limitation in flexibility, since we only shift departure times and travel times within small intervals, but it allows us to work with fixed, a-priori known headway times between two legs.

We then need to ensure that the minimum headway time is retained between the departure times as well as between the arrival times of the two legs involved. Therefore, a valid timetable must respect the two constraints

$$j_1 + s \leq j_2,$$
$$j_1 + d_1 + s \leq j_2 + d_2,$$

where j_1, j_2 are the respective departure times of the first and second leg, d_1, d_2 their respective travel times and s the minimum allowed headway time between them. Note that depending on which train is faster, either only the first or only the second constraint need to be enforced explicitly.

Single-Track Headway Constraints

```
{"first_leg_id": 6096,
 "second_leg_id": 458}
```

Between some pairs of adjacent stations in the network, there exists only one physical track which is thus used in both directions. This situation requires a *single-track headway constraint* between any two trains which use such a track successively and in opposite directions. In this case, the two stated legs represent the two trains involved. A valid timetable then has to make sure that the first train arrives at its destination before the second train departs, i.e. it has to hold

$$j_1 + d_1 \leq j_2,$$

where j_1, j_2 are the respective departure times of the first and second leg and d_1 is the travel time of the first leg.

Dwell Time Constraints


```
{"first_leg_id": 1,
 "second_leg_id": 2,
 "min_dwell_time": 20}
```


The third constraint type, the *dwell time constraints*, require any train to stop long enough in a station so that passengers can get on and off. It uses the field *min_dwell_time* to state the required amount of seconds between the arrival of a train at a station and its subsequent departure towards the next station. Hence, a valid timetable must ensure

$$j_1 + d_1 + c \leq j_2,$$

where j_1, j_2 are the departure times of the first and second leg, d_1 is the travel time of the first leg and c is the minimum required dwell time of the corresponding train in the station involved.

Terminal Turnaround Constraints


```
{"first_leg_id": 206,
 "second_leg_id": 6149,
 "min_turnaround_time": 420}
```


In the underground instances in this library, the trains turn after completing a line and proceed to service the same line in the opposite direction. The required time to perform this turnaround between the completion of the last leg in one direction (*first_leg_id*) and the first leg in the opposite direction (*second_leg_id*) must be respected. To this end, we introduce the *terminal turnaround constraints*, which are structurally identical to the dwell time constraints, such that a valid timetable needs to ensure

$$j_1 + d_1 + c \leq j_2$$

for the departure times j_1 and j_2 of the first and second leg respectively, the travel time d_1 of the first leg and the minimal turnaround time f between the two legs.

Connection Constraints


```
{"first_leg_id": 8846,
 "second_leg_id": 3110,
 "min_connection_time": 180,
 "max_connection_time": 240,
 "connection_type": "arrival_to_departure"}
```


The *connection constraints* use the fields *min_connection_time* and *max_connection_time* to state the minimal and maximal connection time in seconds as well as the field *connection_type* to state the constraint type. In particular, "arrival_to_departure" states that the connection time is measured between the arrival of the first leg and the departure of the second leg, whereas "departure_to_departure" means the time between departures. Therefore, a valid timetable must ensure that

$$\begin{cases} j_1 + d_1 + \underline{h} \leq j_2 \leq j_1 + d_1 + \bar{h}, & \text{if constraint type is “arrival\_to\_departure”} \\ j_1 + \underline{h} \leq j_2 \leq j_1 + \bar{h}, & \text{if constraint type is “departure\_to\_departure”} \end{cases}$$

holds, where j_1, j_2 are the departure times of the first and second leg respectively, d_1 is the travel time of the first leg and $\underline{h}$, $\bar{h}$ are the minimal and maximal connection time respectively.

Recuperation Subnet Constraints

```
{"subnet_id": 1,
 "track_ids": [1, 2, 3, ...]}
```

In the underground application, the power network is subdivided into several power subnets, and recuperation is only possible between trains in the same subnet. This is not an operative constraint in the sense that it may render a timetable invalid, but rather limits recuperation possibilities and therefore affects what an energy-efficient timetable looks like. Each track is assigned to exactly one subnet and the *track_id* of a leg then determines in which subnet the corresponding trip takes place. The details of our underlying recuperation model are described in Section 4.

3.4 Solution Files

In addition to the instances themselves, we provide CSV files named *solution.csv*, which contain the solutions corresponding to the computational results presented in Section 6. These files consist of two columns, where the first column contains the unique identifier *leg_id* and the second column contains the chosen *departure_configuration*. The fields are of the same format as in *timetable.csv*; see the first and last row in Table 1.

3.5 Parsing Routines

The file formats were designed to be easy to read in using Python 3.7 ([VRD09]) and the pandas 1.1.3 data analysis framework ([pdt20, WM10]). The following routines can be used to transform the CSV files into easy-to-use dataframes and the JSON file into a constraint dictionary. Note that these routines already parse the input to integers and floats respectively. In particular, they parse the string representations of the departure configurations to a list containing the three integer values for departure time, travel time and power profile ID.

Parsing timetable.csv

```
import pandas

def parse_dc(s):
    m = map(lambda x: [int(i) for i in x.split('_')],
              s.split(' '))
    return list(m)

dtypes = {'leg_id': 'Int64',
          'train_id': 'Int64',
          'track_id': 'Int64',
          'start_station_id': 'Int64',
          'end_station_id': 'Int64'}

converters = {
    'nominal_departure_configuration': lambda x: parse_dc(x)[0],
```

```
    'departure_configurations': parse_dc}

df = pandas.read_csv('timetable.csv',
                     dtype=dtypes,
                     converters=converters)
```

Parsing profiles.csv

```
import pandas

dtypes = {'profile_id': 'Int64'}
converters = {
    'power_consumptions':
    lambda x: [float(f) for f in x.split(' ')] if x != '' else []}
df = pandas.read_csv('profiles.csv',
                     dtype=dtypes,
                     converters=converters)
```

Parsing constraints.json

```
import codecs
import json

with codecs.open('constraints.json', 'r') as infile:
    constraint_dict = json.load(infile,
                                encoding='utf-8',
                                parse_int=int)
```

Parsing solution.csv

```
import pandas

dtypes = {'leg_id': 'Int64'}
converters = {'departure_configuration': lambda x: parse_dc(x)[0]}

df = pandas.read_csv('solution.csv',
                     dtype=dtypes,
                     converters=converters)
```

4 Objective Functions for Energy-Efficient Timtabling

The data on energy consumption of the trains included in *EETTlib* allows for the adjustment of the provided railway timetables according to many different possible objective functions with respect to energy efficiency. We will now discuss several indicative optimization goals which are important to either a train-operating company (TOC) or an infrastructure manager (IM). A broader outline can be found in [BMS17].

Technical Background and Assumptions The traction energy for the trains is provided via so-called *substations*, which transform the power from the power supply network onto the catenaries, which in turn power the trains. Newer trains can feed energy back to the catenaries when braking. This is called *recuperation*. Recuperated energy can be used by nearby trains for their acceleration, which reduces the overall energy consumption. The closer the trains are to each other, the less energy is lost in the transmission. However, the recuperation energy of one train can only be used if there is a second train accelerating at the same time. In the following exemplary outline of energy-related objective functions, we assume that lossless energy transmission is possible between trains in the same (local) subnetwork of the railway network. Note that it is straightforward to consider degrees of effectiveness in the transmission of braking energy.

Notation Let the set $T = \{0, \ldots, |T| - 1\}$ denote the planning horizon of the timetabling problem, with a resolution in seconds. Further, let A denote the set of all legs to be scheduled. Each leg $a \in A$ has a set of permitted departure times $J_a \subseteq T$ and a set of admissible travel times $D_a \subseteq \mathbb{N}$. We also introduce the set B of subnetworks into which the considered railway network is subdivided. Accordingly, we denote by $A_b \subseteq A$ the set of legs belonging to subnetwork $b \in B$. Note that any leg $a \in A$ can be attributed to exactly one subnetwork $b \in B$ via its *track_id* (see Section 3.3). The consumptions and recuperations of the trains are thus aggregated and balanced within these subnetworks for each instant $t \in T$ in the planning horizon (i.e. for each second in a given sample day). Moreover, let us denote by $p^i_{ad} \in \mathbb{R}$ the power consumption induced by the choice of a travel time $d \in D_a$ for some leg $a \in A$ at i seconds after its departure, $0 \leq i \leq d$. Varying the departure time of a train at a station results in shifting the occurrence of its power consumption profile corresponding to the chosen travel time forward or backward in time. More precisely, the individual power consumption of a leg $a \in A$ at instant $t \in T$ is given by

$$\bar{p}^t_{ajd} := \begin{cases} p^{t-j}_{a,d}, & \text{if } 0 \leq t - j \leq d \\ 0, & \text{otherwise} \end{cases}$$

if departure time $j \in J_a$ and travel time $d \in D_a$ are chosen. Finally, we represent a timetable as a set $X \subseteq \{(a, j, d) \in A \times T \times \mathbb{N} \mid j \in J^a, d \in D^a\}$ with $|X| = |A|$ and $a_1 \neq a_2$ for any two distinct triples $(a_1, j_1, d_1), (a_2, j_2, d_2) \in X$. This means that each leg needs to be assigned a unique departure time and a unique travel time. In order to be feasible, the chosen triples in such a timetable X together need to fulfil all timetabling constraints from Section 3.

We will now define several possible metrics according to which the energy-efficiency of a railway timetable can be optimized. First, we introduce the objective functions we used in our projects with Deutsche Bahn AG for the German railway network and VAG for the Nürnberg underground network as we studied them in [BGM20] and [BMS21] respectively. For these two fine-tuned objective functions, we will provide reference computational results in Section 6. Furthermore, we present a few alternative choices from [BMS17, BMS21] which can be combined with the first two objectives as seen fit by the planners of a railway undertaking. In Appendix B, we show that they can all easily be stated as mixed-integer linear programs (MIPs). Note that the computational complexity of the overall problem may vary significantly with the chosen optimization objective.

Optimization Problem for the Railway Instances A railway company can reduce the price component of its electricity bill that depends on peak power consumption by avoiding too many simultaneous train departures (involving trains of only that company). In the German railway system, peak power consumption is calculated and priced over all trains running at a given point in time, independent of their location in the network. Thus, we assume $|B| = 1$ here. In the railway instances of *EETTlib*, we only consider the optimal choice of departure time, not that of the travel time on a leg, i.e. we also have $|D_a| = 1$ for all $a \in A$. Further, the peak consumption charge is not computed from the instantaneous consumption values arising in individual points in time, but from the average consumption values over 15-minute intervals.

To facilitate the exposition, we assume $|T|$ to be a multiple of 900 and introduce the set

$$T_{15} := \left\{ [900 \cdot i, 900 \cdot (i+1) - 1] \cap T : i \in \left\{ 0, \dots, \frac{|T|}{900} - 1 \right\} \right\}$$

of all consecutive 15-minute (i.e. 900-second) intervals in the planning horizon. For a given feasible timetable X, the highest such average value is then given by

$$\frac{1}{900} \cdot \max_{I \in T_{15}} \left(\sum_{t \in I} \max \left\{ \sum_{(a,j,d) \in X} \bar{p}^t_{ajd}, 0 \right\} \right).$$

In other words, we sum up all the individual power consumption profiles for each trip of the trains under consideration according to the chosen departure times. Note that energy recuperated by braking trains is lost if there are no other trains that would be ready to make use of it at the same time, thus the inner maximum with 0. The resulting value, multiplied by a certain cost factor, determines the load-dependent part of the electricity cost of a German railway company, which we would like to minimize.

Finally, it is an important consideration for a railway company not only to reduce average peak consumption but also to limit instantaneous power consumption in order to maintain stability of the power supply system. To this end, we incorporate an upper limit $U \in \mathbb{R}$ for the maximum power consumption occurring at any instant during the planning horizon. This leads to the following additional constraint for the feasibility of a timetable X:

$$\sum_{(a,j,d) \in X} \bar{p}^t_{ajd} \leq U \quad (\forall t \in T).$$

A reference MIP formulation for this optimization problem is given as model (2) in Appendix B.

Optimization Problem for the Underground Instances For the underground instances, our aim is to minimize the total power consumption over the planning horizon. Here, we also have to consider that the power supply network is separated into several subnetworks $b \in B$. For a feasible timetable X, we thus minimize

$$\sum_{b \in B} \sum_{t \in T} \max \left\{ \sum_{(a,j,d) \in X} \bar{p}^t_{aj_a d_a}, 0 \right\}.$$

Model (3) in Appendix B serves as a reference MIP formulation for this problem.

Further Possible Objectives A variety of further possible performance metrics for energy-efficient timetabling have been proposed in [BMS17]. We will sketch a few of them in the following in order to demonstrate the broad applicability of the data provided in *EETTlib*. For the sake of simplicity, we drop the subdivision of the power supply network and base our description of these alternatives on the case of the railway instances.

From the point of view of a train-operating company (TOC), there are pricing schemes which refund recuperated energy separately instead of balancing it against the energy drawn from the substations. Thus, the TOC has an incentive not to minimize the *gross* maximum average power consumption as above, but rather the *net* maximum average power consumption, which is given by

$$\frac{1}{900} \cdot \max_{I \in T_{15}} \left(\sum_{t \in I} \sum_{(a,j,d) \in X} \max\{\bar{p}^t_{ajd}, 0\} \right),$$

cf. the adaptation of model (3) described in Appendix B.

From the point of view of the IM, it might be an important target to minimize power fluctuation over the planning horizon to stabilize the power supply system. Generally, the power fluctuation is defined as the distance between the considered system-wide power profile and an ideally balanced, i.e. constant, power profile. As this distance can be measured in different norms, this approach leads to a family of optimization problems for which we give some examples in Appendix B.

5 Instances

We now describe the instances in our library *EETTlib*, which can directly be used as input for the energy-efficient timetabling optimization problems presented in the previous sections. All instances are based on real-world data from our industry partners Deutsche Bahn AG and VAG. Details on the data compilation from the industry projects described in Section 2 are given in Appendix C.

5.1 Railway Instances

The railway instances are based on data provided by our industry partner Deutsche Bahn AG. We received the German railway passenger timetable for the trains operated by DB Regio AG and DB Fernverkehr AG for the year 2015, which represents about 80% of passenger traffic in Germany. In the following, we describe this timetabling data in detail and outline which assumptions were used to generate power profiles for the trains and to fill in some of the missing parameters. In total, we compiled 31 instances from this data, which represent subnetworks of the German railway network, including one instance containing all German DB rail passenger traffic on a given sample day.

Instance Characteristics Our railway instances contain the traffic between 4 a.m. and 10 p.m. on a typical working day and differ in the trains selected from the complete timetable. The instances mostly describe different regions of Germany and can be grouped into three types according to their scope. The first type are instances where for a given station all trains with at least one trip starting or ending there within the planning interval are considered (both regional and long-distance traffic). We call these the *local* instances. The stations these instances are based on range from smaller stations to large hubs in the network. The second type are the *regional* instances, which comprise regional traffic according to the regional subdivisions of DB Regio AG as of 2015. Finally, we consider a set of *national* instances, which are one instance for all regional traffic in Germany (*Regionalverkehr*), one instance for the German long-distance traffic (*Fernverkehr*) and one instance for the complete German passenger traffic with both regional and long-distance traffic combined (*Deutschland*). Altogether, we have 18 local instances, 10 regional instances and 3 national instances, or 31 in total.

To find an optimal adjustment of the timetable, we allowed to shift each train departure by up to $\pm$ 3 minutes in steps of full minutes compared to the original departure time. However, we did not allow the departure or arrival of a train to be shifted outside of the considered planning interval. The departure times for trips starting before 4 a.m. and extending into the planning interval as well as those starting in the planning interval and extending past 10 p.m. were considered as fixed. In this case, the part of their power consumption which falls inside the planning interval was added to the consumption of the variable trains as a fixed base load.

Table 4 summarizes several relevant characeristics of the railway instances. It shows for each of the 31 instances the number of trains and legs as well as the number of pairs of legs that have a waiting, headway and connection relation respectively.

Instance	Trains	Legs	Number of leg dependencies		
			Headway	Dwell	Connection
Zeil	42	762	703	720	102
Bayreuth Hbf	68	327	275	259	47
Passau	75	1 040	918	965	149
Jena Paradies	78	1 102	933	1 024	205
Lichtenfels	113	1 650	1 398	1 537	398
Erlangen	142	2 969	2 716	2 827	1 217
Bamberg	209	3 644	3 271	3 435	1 037
Aschaffenburg	245	3 463	3 139	3 218	1 218
Kiel Hbf	297	2 130	1 929	1 833	747
Leipzig Hbf (tief)	369	6 810	6 565	6 441	10 647
Würzburg Hbf	371	4 456	3 873	4 085	1 932
Dresden	422	6 936	6 461	6 514	5 202
Ulm Hbf	468	5 729	5 008	5 261	2 273
Stuttgart Hbf (tief)	628	11 594	10 517	10 966	32 673
Berlin Hbf (S-Bahn)	639	16 114	12 735	15 475	77 762
Hamburg-Altona(S)	722	12 373	11 162	11 651	44 542
Frankfurt(Main)Hbf	728	8 626	7 651	7 898	5 124
Nürnberg	951	12 189	11 164	11 238	9 515
S-Bahn Hamburg	1 208	17 533	15 063	16 325	67 142
Regio Nord	1 476	13 379	12 671	11 903	8 061
Regio Nordost	1 494	16 496	15 269	15 002	8 636
Regio Hessen	1 547	25 092	23 070	23 545	52 289
Regio Südwest	1 863	24 191	22 724	22 328	13 136
Regio Südost	2 357	31 917	30 070	29 560	23 424
Regio Baden-Württemberg	2 382	30 172	27 562	27 790	43 892
S-Bahn Berlin	2 578	53 353	42 605	50 775	273 964
Regio NRW	2 826	47 026	43 656	44 200	63 231
Regio Bayern	3 554	49 262	43 599	45 708	99 305
Fernverkehr	667	7 053	5 788	6 386	4 702
Regionalverkehr	21 288	308 472	276 441	287 184	667 682
Deutschland	21 955	315 525	282 219	293 570	697 885

Table 4: DB instances

5.2 Underground Instances

The underground instances stem from a joint project with VAG Verkehrs-Aktiengesellschaft, the operator of underground traffic in the German city of Nürnberg. All data which went into the compilation of these instances is completely based on real-world data from the Nürnberg underground system, including actual measurements of the power consumptions of the trains, recorded via the tachographs of the trains. Part of this data as well as further information, for example on the underground network, can be found in the VAG-OpenData portal under `https://opendata.vag.de`.

Instance Characteristics For the underground instances, we used four different timetables from the Nürnberg underground system for the year 2020. These are two workday timetables, one timetable for Saturdays and, finally, a timetable for Sundays and holidays. The workday timetables are both used Monday through Friday, but there is one version for regular workdays and another one for school holidays, where transportation in the morning and early afternoon are planned differently. Each timetable contains the scheduled trains of a whole operating day, which usually starts after 4 a.m. and ends before 2 a.m. the next morning. For the underground timetables, all constraint types described in Section 3.3 are relevant.

For each scheduled departure, we allow the departure time to be shifted by up to $\pm$ 15 seconds in 5-second intervals. Further, each leg may choose from three or four different travel times, each with their own power profile. The alternate choices represent the three most frequent travel times measured on the respective track as well as the originally scheduled travel time if it was not among the three most frequent ones.

An overview of the underground instances and their characteristics is given in Table 5.

			Number of leg dependencies			
Instance	Trains	Legs	Headway	Dwell	Turnaround	Connection
school, line U1	513	11 580	11 660	11 067	487	0
school, lines U2U3	992	12 963	12 919	11 971	954	1
school, all lines	1 505	24 543	24 579	23 038	1 442	152
school free, line U1	402	10 200	10 275	9 798	381	0
school free, lines U2U3	987	12 932	12 888	11 945	952	1
school free, all lines	1 389	23 132	23 163	21 743	1 334	152
saturday, line U1	309	7 917	7 922	7 608	295	0
saturday, lines U2U3	771	10 188	10 144	9 417	749	1
saturday, all lines	1 080	18 105	18 066	17 025	1 044	339
sunday, line U1	240	6 240	6 303	6 000	231	0
sunday, lines U2U3	601	8 122	8 078	7 521	583	0
sunday, all lines	841	14 362	14 381	13 521	814	772

Table 5: Underground instances

6 Reference Results

In this section, we present some reference computational results for solving the timetabling problems via the models described in Appendix B using the data provided in our instance library *EETTlib*. The computations have been run on machines with a Intel Xeon E5-2643 v4 CPU ("Broadwell", 6 cores, 3.4 GHz) and 256 GB RAM. We remark that this much RAM is not actually required as all underground instances can be run with 16 GB RAM, most railway instances can be run with 32 GB RAM, and only the two largest instances in the library (*Deutschland* and *Regionalverkehr*) require more than 64 GB RAM. Our implementation for these experiments is based on Python 3.7 ([VRD09]) with Gurobi 9.0.1 ([GO21]) as an MIP solver. We chose a time limit of one hour and set the number of parallel threads to four. For the railway instances, we additionally set the Gurobi parameter *LazyConstraints* to 1. This was required for the Benders decomposition used to solve them, which we implemented via a Gurobi callback. The underground instances use the default Gurobi parameters throughout.

The reference results for the railway instances were obtained using the totally unimodular dual-flow formulation introduced in [BGMS18, BMS21] for the polyhedron describing the feasible timetable adjustments. It is applicable because the corresponding compatibility graphs have the *staircase property* studied in [BGMS18]. Further, we use the Benders decomposition scheme proposed in [BMS21] in order to efficiently solve the resulting model. For the maximum power consumption U, we chose the highest consumption at any point in time during the planning horizon according to the reference timetable. The results are summarized in Table 6.

On this instance subset, we see a large variance in terms of solvability. Some instances, for example *Lichtenfels* or *Erlangen*, can be solved within minutes; for other instances, like *Regio Hessen*, the generic MIP solver achieves decent quality solutions within the first hour of computation, and some instances, such as *Zeil* or *Regionalverkehr*, remain very hard to solve. Interestingly, the latter two instances are difficult for different reasons. The instance *Zeil* is structurally hard, as it contains few trains for which the MIP solver has to search within a relatively small solution space, but among relatively many solutions of similar quality. On the other hand, instance *Regionalverkehr* instance is difficult to solve due to its sheer size; Gurobi does not even finish preprocessing within the first hour of computation. Finally, we point out that the instance *Deutschland*, containing both the energy-intensive high-speed trains and all regional trains, is structurally easier than instance *Regionalverkehr*, although it is slightly larger. This is due to the fact that the high-speed trains with their much higher energy consumption largely dominate the consumption of the regional trains.

Instance	Base (MW)	Objective (MW)	Bound (MW)	Time/Gap	Saving
Zeil	2.2	1.4	0.8	44.7%	34.0%
Bayreuth Hbf	2.1	1.5	1.3	17.9%	25.9%
Passau	5.4	3.9	3.4	13.2%	26.4%
Jena Paradies	24.1	20.4	20.4	0.1%	15.3%
Lichtenfels	18.6	13.7	13.7	218.4 s	26.3%
Erlangen	18.0	13.3	13.3	364.6 s	26.0%
Bamberg	27.7	21.9	21.9	0.0%	20.8%
Aschaffenburg	25.8	21.8	21.7	0.2%	15.8%
Kiel Hbf	17.4	13.2	13.1	0.5%	24.1%
Leipzig Hbf (tief)	5.7	4.6	4.6	613.1s	20.1%
Würzburg Hbf	62.3	50.9	50.8	0.1%	18.4%
Dresden	22.3	17.3	17.1	1.0%	22.4%
Ulm Hbf	35.3	29.2	29.1	0.2%	17.4%
Stuttgart Hbf (tief)	4.3	3.9	3.9	68.5 s	9.9%
Berlin Hbf (S-Bahn)	2.8	2.6	2.6	0.1%	9.2%
Hamburg-Altona (S-Bahn)	3.4	3.0	3.0	0.0%	11.1%
Frankfurt(Main)Hbf	116.5	98.5	98.5	649.3 s	15.5%
Nürnberg	84.8	72.7	72.7	2675.4 s	14.2%
S-Bahn Hamburg	5.4	4.6	4.6	2126.9 s	13.8%
Regio Nord	18.2	17.0	13.3	21.7%	6.6%
Regio Nordost	24.2	24.2	10.3	57.7%	0.0%
Regio Hessen	15.5	13.1	12.9	1.5%	16.0%
Regio Südwest	21.3	19.8	16.3	17.7%	7.1%
Regio Südost	32.0	32.0	19.2	39.8%	0.0%
Regio Baden-Württemberg	30.0	30.0	21.4	28.6%	0.0%
S-Bahn Berlin	8.9	8.5	8.5	0.6%	4.8%
Regio NRW	35.9	30.3	28.9	4.7%	15.5%
Regio Bayern	42.0	33.3	32.6	2.1%	20.7%
Fernverkehr	254.5	228.4	225.5	1.3%	10.2%
Regionalverkehr	217.8	217.8	-	-	0.0%
Deutschland	452.8	452.8	364.7	19.4%	0.0%

Table 6: Performance of the Benders approach on the railway instances

For the underground instances, we used the stable-set formulation for the timetable adjustment polyhedron proposed in [BGM20]. It completely describes the convex hull of feasible timetable adaptations, but can be very large in terms of the number of constraints in general. However, the corresponding upper bounds derived in [BGM20] state that the number of constraints will stay modest for these instances as there are only 3–4 different travel times to choose from per leg. The results for the underground instances are summarized in Table 7.

Instance	Base (MWh)	Objective (MWh)	Bound (MWh)	Time/Gap	Saving
school, line U1	71.3	60.0	56.1	6.4%	15.9%
school, lines U2U3	45.2	42.0	37.0	12.0%	7.1%
school, all lines	116.5	108.5	93.1	14.2%	6.9%
school free, line U1	60.6	57.2	49.7	13.2%	5.6%
school free, lines U2U3	43.3	40.4	35.5	12.1%	6.6%
school free, all lines	103.9	97.9	85.3	12.9%	5.8%
saturday, line U1	53.0	45.4	38.8	14.6%	14.4%
saturday, lines U2U3	27.3	27.2	22.2	18.5%	0.4%
saturday, all lines	80.4	74.7	61.1	18.2%	7.1%
sunday, line U1	47.5	33.3	30.7	7.9%	29.8%
sunday, lines U2U3	24.0	21.4	19.2	10.2%	10.8%
sunday, all lines	71.4	65.2	50.0	23.4%	8.7%

Table 7: Performance of the stable-set formulation on the underground instances

For the schedules *school* and *sunday*, it is more difficult to solve those instances which contain the traffic of more underground lines. For the other two schedules *school free* and *saturday*, this is not clearly the case. Overall, the instance *school, line U1* seems to be the easiest one, and *sunday, all lines* the hardest instance to solve.

Finally, the above results show that energy-efficient timetabling by slight departure time and travel time adjustments is a worthwhile consideration for a railway company. The approach of improving an existing timetable draft yields a savings potential of up to 10% for the nationwide railway instances, and 5.8 – 8.7% for the underground instances containing all lines. For a full discussion of these savings potentials under varying circumstances and their technical background, we refer the interested reader to [BGMS18] as well as [BMS21] for the railway setting and to [BGM20, BGM+21a] for the underground setting.

7 Conclusions

Timetabling, and especially energy-efficient timetabling, has become a very active field of research in discrete optimization, as the recent review [SGK17] underlines. It offers both interesting mathematical structures to investigate and leads to significant benefits when applied in practice. However, the possibilities to obtain real-world data to validate the optimization models have always been scarce. In this article, we have introduced *EETTlib*, an instance library for energy-efficient train timetabling problems. Its intention is to fill an important gap in the field of timetabling research: the availability of high-quality, easy-to-use, real-world timetabling data in the public domain.

This library originated from two long-standing research projects with prominent providers of public transport in Germany. Together with the practitioners at the planning departments of our industry partners, we took extensive care in collecting, curating and compiling the data from the very diverse software and controlling systems used at these companies. The result is a set of 43 timetabling instances representing real-world problem structure. In the case of the railway instances, they lead to a basic model which can be used to study the interdependencies that optimization decisions lead to. In the case of the underground instances, the data provided here was even proved to allow for optimized timetable adaptations which can reliably be used in practice, as test runs of our optimization results in the underground system of Nürnberg have shown. Notably, our problem data can be used in conjunction with different energy-related optimization goals to be studied, as our discussion of possible objective functions in Section 4 outlines. A further important point is that the mathematical problem underlying energy-efficient timetable optimization is actually a resource-constrained project scheduling problem (RCPSP, cf. [SZ15]). Thus, the data provided here also leads to interesting RCPSP instances to be studied.

Everybody is invited to download and use the instances in *EETTlib* in publications as well as model and algorithm development. We hope that our library will be useful for researchers and practitioners in the timetabling domain and also for the operations research community as a whole.

References

[BAP+14] Berit D. Brouer, J. Fernando Alvarez, Christian E. M. Plum, David Pisinger, and Mikkel M. Sigurd. A base integer programming model and benchmark suite for liner-shipping network design. *Transportation Science*, 48(2):281–312, 2014.

[BGM20] Andreas Bärmann, Patrick Gemander, and Maximilian Merkert. The clique problem with multiple-choice constraints under a cycle-free dependency graph. *Discrete Applied Mathematics*, 283:59–77, 2020.

[BGM+21a] Andreas Bärmann, Patrick Gemander, Alexander Martin, Maximilian Merkert, and Frederik Nöth. *German Success Stories in Industrial Mathematics*, volume 35 of *Mathematics in Industry*, chapter Energy-Efficient Timetabling in a German Underground System. Springer, 2021.

[BGM+21b] Andreas Bärmann, Patrick Gemander, Maximilian Merkert, Ann-Kathrin Wiertz, and Francisco Javier Zaragoza Martínez. Algorithms for the clique problem with multiple-choice constraints under a series-parallel dependency graph. Submitted; available under: `http://www.optimization-online.org/DB_HTML/2021/05/8393.html`, 2021.

[BGMM21] Andreas Bärmann, Patrick Gemander, Alexander Martin, and Maximilian Merkert. On recognizing staircase compatibility, 2021. Submitted; available under: `http://www.optimization-online.org/DB_HTML/2020/12/8138.html`.

[BGMS18] Andreas Bärmann, Thorsten Gellermann, Maximilian Merkert, and Oskar Schneider. Staircase compatibility and its applications in scheduling and piecewise linearization. *Discrete Optimization*, 29:111–132, 2018.

[BMS17] Andreas Bärmann, Alexander Martin, and Oskar Schneider. A comparison of performance metrics for balancing the power consumption of trains in a railway network by slight timetable adaptation. *Public Transport*, 9(1-2):95–113, 2017.

[BMS21] Andreas Bärmann, Alexander Martin, and Oskar Schneider. Efficient formulations and decomposition approaches for power peak reduction in railway traffic via timetabling. *Transportation Science*, 55(3):747–767, 2021.

[CC a] CC BY-SA 4.0. Creative Commons Attribution-ShareAlike 4.0 International Public License. Available under: `https://creativecommons.org/licenses/by-sa/4.0/legalcode`.

[CC b] CC BY-SA 4.0. Creative Commons Attribution-ShareAlike 4.0 International Public License – Human-Readable Summary. Available under: `https://creativecommons.org/licenses/by-sa/4.0`.

[CCMR10] A. Ceselli, R. Cordone, Y. Melzani, and G. Righini. Optimization algorithms for the max edge weighted clique problem with multiple choice constraints. In *Proceedings of the 9th Cologne-Twente Workshop on Graphs and Combinatorial Optimization*, pages 37–41, 2010.

[CT12] Valentina Cacchiani and Paolo Toth. Nominal and robust train timetabling problems. *European Journal of Operational Research*, 219(3):727–737, 2012.

[CZ17] David Canca and Alejandro Zarzo. Design of energy-efficient timetables in two-way railway rapid transit lines. *Transportation Research Part B: Methodological*, 102:142–161, 2017.

[DB 22] DB Energie GmbH. Preisblatt für die Nutzung des 16,7-Hz-Bahnstromnetzes (Bahnstromnetz) gültig ab 01.01.2022. Available under: `https://www.dbenergie.de/dbenergie-de/16-7-Hz-Preisblaetter-Netznutzung-4571178`, 2022.

[EKS+08] B. Erol, M. Klemenz, T. Schlechte, S. Schulz, and A. Tanner. *Computers in Railways XI*, chapter TTPlib 2008 – A Library for Train Timetabling Problems, pages 605–614. WIT Press, 2008.

[GHG+21] Ambros Gleixner, Gregor Hendel, Gerald Gamrath, Tobias Achterberg, Michael Bastubbe, Timo Berthold, Philipp M. Christophel, Kati Jarck, Thorsten Koch, Jeff Linderoth, Marco Lübbecke, Hans D. Mittelmann, Derya Ozyurt, Ted K. Ralphs, Domenico Salvagnin, and Yuji Shinano. MIPLIB 2017: Data-Driven Compilation of the 6th Mixed-Integer Programming Library. *Mathematical Programming Computation*, 13:443–490, 2021.

[GO21] LLC Gurobi Optimization. Gurobi optimizer reference manual, 2021.

[Int20] International Energy Agency (IEA). The future of rail. Available under: `https://www.iea.org/reports/the-future-of-rail`, 2020.

[KMV01] Thorsten Koch, Alexander Martin, and Stefan Voß. *Steiner Trees in Industry*, chapter SteinLib: An Updated Library on Steiner Tree Problems in Graphs. Kluwer Academic Publishers, 2001.

[KS96] Rainer Kolisch and Arno Sprecher. PSPLIB – a project scheduling problem library. *European Journal of Operational Research*, 96:205–216, 1996.

[Lie08] Christian Liebchen. The first optimized railway timetable in practice. *Transpotation Science*, 42(4):420–435, 2008.

[MGM20] Mahdi Hampaeyan Miandoab, Vahidreza Ghezavati, and Davood Mohammaditabar. Developing a simultaneous scheduling of passenger and freight trains for an inter-city railway considering optimization of carbon emissions and waiting times. *Journal of Cleaner Production*, 248:119303, 2020.

[OPTW10] S. Orlowski, M. Pióro, A. Tomaszewski, and R. Wessäly. SNDlib 1.0–Survivable Network Design Library. *Networks*, 55(3):276–286, 2010.

[Pac16] Jörn Pachl. *Systemtechnik des Schienenverkehrs: Bahnbetrieb planen, steuern und sichern*. Springer Vieweg, 2016.

[pdt20] The pandas development team. Pandas 1.1.3. Available under: `https://doi.org/10.5281/zenodo.3509134`, 2020.

[Pro21] Project Energy-efficient Timetable Optimization. Homepage: `https://www.scs.fraunhofer.de/de/referenzen/vag-fahrplanoptimierung.html`, 2021.

[Pro22] Project E-Motion: Energy-Efficient Mobility. Homepage: `https://en.www.math.fau.de/edom/projects-edom/energy/robust-power-load-balancing-in-railway-networks/`, 2022.

[RDLLP+17] David Roch-Dupré, Álvaro J. López-López, Ramón R. Pecharromán, Asunción P. Cucala, and Antonio Fernández-Cardador. Analysis of the demand charge in dc railway systems and reduction of its economic impact with energy storage systems. *International Journal of Electrical Power & Energy Systems*, 93:459–467, 2017.

[Rei91] Gerhard Reinelt. TSPLIB – a travelling salesman problem library. *ORSA Journal on Computing*, 3(4):376–384, 1991.

[SAG+22] A. Schiewe, S. Albert, V. Grafe, P. Schiewe, A. Schöbel, and F. Spühler. LinTim – integrated optimization in public transportation. Homepage: `https://www.lintim.net`, 2022.

[SG97] Brunilde Sansó and Pierre Girard. Instantaneous power peak reduction and train scheduling desynchronization in subway systems. *Transportation science*, 31(4):312–323, 1997.

[SGK17] Gerben M. Scheepmaker, Rob M. P. Goverde, and Leo G. Kroon. Review of energy-efficient train control and timetabling. *European Journal of Operational Research*, 257(2):355–376, 2017.

[SWM+18] Huijun Sun, Jianjun Wu, Hongnan Ma, Xin Yang, and Ziyou Gao. A bi-objective timetable optimization model for urban rail transit based on the time-dependent passenger volume. *IEEE Transactions on Intelligent Transportation Systems*, 20(2):604–615, 2018.

[SZ15] Christoph Schwindt and Jürgen Zimmermann, editors. *Handbook on Project Scheduling* (*Vol. 1* + *Vol. 2*). Springer, 2015.

[VRD09] Guido Van Rossum and Fred L. Drake. *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA, 2009.

[WBGC22] Pengling Wang, Nikola Bešinović, Rob M. P. Goverde, and Francesco Corman. Improving the utilization of regenerative energy and shaving power peaks by railway timetable adjustment. *IEEE Transactions on Intelligent Transportation Systems*, 2022. DOI: 10.1109/TITS.2022.3145390.

[WM10] Wes McKinney. Data Structures for Statistical Computing in Python. In Stéfan van der Walt and Jarrod Millman, editors, *Proceedings of the 9th Python in Science Conference*, pages 56 – 61, 2010.

[Wri22] Kelseigh Wrigley. Peak and off-peak electricity times explained. Posted on `https://www.canstarblue.com.au/electricity/peak-off-peak-electricity-times/`, 2022.

[YBG19] Fei Yan, Nikola Bešinović, and Rob M. P. Goverde. Multi-objective periodic railway timetabling on dense heterogeneous railway corridors. *Transportation Research Part B: Methodological*, 125:52–75, 2019.

[YNLT14] Xin Yang, Bin Ning, Xiang Li, and Tao Tang. A two-objective timetable optimization model in subway systems. *IEEE Transactions on Intelligent Transportation Systems*, 15(5):1913–1921, 2014.

[ZRH+17] Ning Zhao, Clive Roberts, Stuart Hillmansen, Zhongbei Tian, Paul Weston, and Lei Chen. An integrated metro operation optimization to minimize energy consumption. *Transportation Research Part C: Emerging Technologies*, 75:168–182, 2017.

A Licensing of the Provided Data

The instances provided in EETTlib can be downloaded and used under the Creative Commons Attribution-ShareAlike 4.0 International Public License (CC BY-SA 4.0, see [CC a]).

A short, human-readable summary of the license terms as presented in [CC b] is given in the following. All users are free to copy and redistribute the material in any medium or format as well as to remix, transform and build upon the material for any purpose, even commercially. In turn, all users must give appropriate credit, provide a link to the license and indicate if changes were made. Users may do so in any reasonable manner, but not in any way that suggests the authors endorse them or their use. Further, if users remix, transform or build upon the material, they must distribute their contributions under the same license as the original. In doing so, they may not apply legal terms or technological measures that legally restrict others from doing anything the license permits. Users do not have to comply with the license for elements of the material in the public domain or where their use is permitted by an applicable exception or limitation. No warranties are given. The license may not give users all of the permissions necessary for their intended use. For example, other rights such as publicity, privacy or moral rights may limit how they use the material.

B Mathematical Models

In the following, we introduce a graph-based model for the set of feasible timetable adjustments. Note that the model provided here is not necessarily the most efficient formulation for the problem under consideration. Rather, the model formulation stated here shall serve as a reference for the precise definition of the energy-efficient timetabling problems covered in EETTlib. It is indeed the main purpose of this instance library to allow its users to develop and test their own model formulations and algorithmic approaches to find the most efficient way of solving energy-efficient railway timetabling problems.

B.1 Graph Model for Feasible Timetable Adaptations

We start by describing the feasible set of possible timetable adaptations. To model this set, we use a graph-based formulation as a clique problem with multiple-choice constraints (CPMC), as introduced in [CCMR10, BGMS18] and further investigated in [BGM20, BMS21, BGMM21, BGM$^+$21b].

CPMC consists in finding a clique of size $m \in \mathbb{N}$ in an m-partite graph $G = (V, E)$. Let $\mathcal{V} = \{V_1, \ldots, V_m\}$ be the corresponding partition of the node set V into subsets. Then the task is to choose exactly one node from each of these subsets. Further, let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be the so-called *dependency graph* of a CPMC-instance $(G, \mathcal{V})$, which contains an edge between two distinct subsets $V_i, V_j \in \mathcal{V}$ if and only if there are two nodes $v \in V_i$ and $w \in V_j$ such that $\{v, w\} \notin E$. The dependency graph is a very helpful structure, as the presence of an edge $\{V_i, V_j\}$ shows that the choice of one of the nodes in V_i may directly restrict the choice of the node from V_j, and vice versa.

With binary variables x_v stating whether a node $v \in V$ has been chosen, a generic formulation of CPMC as a binary feasibility problem may be stated as follows:

$$
\begin{aligned}
\text{find} \quad & x & & \\
\text{s.t.} \quad & \sum_{v \in V_i} x_v = 1 & & (\forall V_i \in \mathcal{V}) \qquad \text{(1a)} \\
& x_v \leq \sum_{\substack{w \in V_j: \\ \{v,w\} \in E}} x_w & & (\{V_i, V_j\} \in \mathcal{E})(\forall v \in V_i) \qquad \text{(1b)} \\
& x \in \{0,1\}^{|V|}. & & \qquad \text{(1c)}
\end{aligned}
$$

The multiple-choice constraints (1a) enforce that exactly one node v is chosen per subset $V_i \in \mathcal{V}$. The compatibility constraints (1b) ensure that the chosen nodes together form a clique in G. Finally, via constraint (1c), all x-variables have to be binary. Note that the decision variant of CPMC, which asks if G contains an m-clique, is NP-complete in general (see [BGM20]). The works [BGMS18, BGM20, BMS21] investigate polynomial-time solvable subcases, develop specialized compact linear programming formulations for these subcases and study their application in the context of energy-efficient timetabling.

To model the choice of a feasible timetable adaptation as an instance of CPMC, we need to define its compatibility graph and its dependency graph. For this purpose, we need to introduce some further notation in addition to what was introduced in Section 4. Let S be the set which contains all pairs of legs from A belonging to two different trains passing the same track in direct succession. For a pair $(a_1, a_2) \in S$, the trains on these two legs have to meet a safety distance which is given by the minimum headway time $s_{a_1a_2}$. We assume that it is sufficient to enforce this minimum headway between the respective times of departure and arrival of the two trains. In a similar fashion, the set W contains all pairs of trains between which a single-track headway relation exists. Further, we introduce the set C, which contains all pairs of directly successive legs of the same train. For any $(a_1, a_2) \in C$, the minimum dwell time between a train completing leg a_1 and departing for the subsequent leg a_2 is denoted by $c_{a_1a_2}$. The set F contains all pairs of legs between which a train has to turn; the respective turnaround time is f_{a_1,a_2} for any $(a_1, a_2) \in F$. Finally, we introduce the two sets H^1 and H^2 with all pairs of legs (a_1, a_2) belonging to two different trains where it has to be possible for the passenger to change from the first to the second train – the former set containing the arrival-to-departure leg pairs and the latter set containing the departure-to-departure leg pairs. In both cases, the corresponding minimal and maximal connection times are given by $\underline{h}_{a_1a_2}$ and $\bar{h}_{a_1a_2}$ respectively.

The set of feasible timetable adaptations can now be stated as an instance of (CPMC) by defining a suitable graph $G = (V, E)$ to model the compatibilities between the departure configurations of the legs to be scheduled. For each such departure configuration $(j, d) \in J_a \times D_a$ for a leg $a \in A$, we introduce a corresponding node in the graph. The set of all nodes V is thus given by

$$V := \{(a, j, d) \mid a \in A, j \in J_a, d \in D_a\}.$$

For ease of exposition, we describe the edge set E via the set of edges $\bar{E}$ of the complement graph $\bar{G}$ of G. This complement graph, also called *conflict graph*, models incompatibilities between the individual departure configurations of the legs. In the definition of the edges, we use the constraints introduced in Section 3. First, in order to enforce the choice of exactly one departure configuration per leg $a \in A$, we define the edge set

$$\bar{E}^{\mathrm{MC}}_a := \{\{(a, j_1, d_1), (a, j_2, d_2)\} \mid j_1, j_2 \in J_a, d_1, d_2 \in D_a, (j_1, d_1) \neq (j_2, d_2)\}.$$

For any two trains with a pair of successive legs $(a_1, a_2) \in S$ on the same track, we have to meet the minimum headway times. This leads to the following edges:

$$\begin{aligned}\bar{E}^{\mathrm{HW}}_{a_1a_2} := \{\{(a_1, j_1, d_1), (a_2, j_2, d_2)\} \mid j_1 \in J_{a_1}, j_2 \in J_{a_2}, d_1 \in D_{a_1}, d_2 \in D_{a_2},\\ j_2 < j_1 + \max\{d_1 - d_2, 0\} + s_{a_1a_2}\}.\end{aligned}$$

The single-track headway relations for the legs $(a_1, a_2) \in W$ are modelled via the edges

$$\bar{E}^{\mathrm{SHW}}_{a_1a_2} := \{\{(a_1, j_1, d_1), (a_2, j_2, d_2)\} \mid j_1 \in J_{a_1}, j_2 \in J_{a_2}, d_1 \in D_{a_1}, d_2 \in D_{a_2}, j_2 < j_1 + d_1\}.$$

For each pair of successive legs $(a_1, a_2) \in C$ of the same train, we have to respect the minimum dwell time constraints, which leads to the edges

$$\begin{aligned}\bar{E}^{\mathrm{DT}}_{a_1a_2} := \{\{(a_1, j_1, d_1), (a_2, j_2, d_2)\} \mid j_1 \in J_{a_1}, j_2 \in J_{a_2}, d_1 \in D_{a_1}, d_2 \in D_{a_2},\\ j_2 < j_1 + d_1 + c_{a_1a_2}\}.\end{aligned}$$

For legs $(a_1, a_2) \in F$ in turnaround relation, we have the edges

$$\begin{aligned}\bar{E}^{\mathrm{TA}}_{a_1a_2} := \{\{(a_1, j_1, d_1), (a_2, j_2, d_2)\} \mid j_1 \in J^{a_1}, j_2 \in J^{a_2}, d_1 \in D^{a_1}, d_2 \in D^{a_2},\\ j_2 < j_1 + d_1 + f^{a_1a_2}\}.\end{aligned}$$

Last, but not least, the two types of connection constraints for two corresponding legs (a_1, a_2) lead to the edges

$$\begin{aligned}\bar{E}^{\mathrm{CC1}}_{a_1a_2} := \{\{(a_1, j_1, d_1), (a_2, j_2, d_2)\} \mid j_1 \in J_{a_1}, j_2 \in J_{a_2}, d_1 \in D_{a_1}, d_2 \in D_{a_2},\\ (j_2 < j_1 + d_1 + \underline{h}_{a_1a_2}) \vee (j_2 > j_1 + d_1 + \bar{h}_{a_1a_2})\}\end{aligned}$$

if $(a_1, a_2) \in H^1$ and

$$\bar{E}^{\text{CC2}}_{a_1 a_2} := \{\{(a_1, j_1, d_1), (a_2, j_2, d_2)\} \mid j_1 \in J_{a_1}, j_2 \in J_{a_2}, d_1 \in D_{a_1}, d_2 \in D_{a_2}, (j_2 < j_1 + \underline{h}_{a_1 a_2}) \vee (j_2 > j_1 + \bar{h}_{a_1 a_2})\}$$

if $(a_1, a_2) \in H^2$.

Combining all the above types of edges yields the edge set $\bar{E}$ of the complement graph $\bar{G}$:

$$\bar{E} := \bigcup_{a \in A} \bar{E}^{\text{MC}}_a \cup \bigcup_{(a_1,a_2) \in S} \bar{E}^{\text{HW}}_{a_1 a_2} \cup \bigcup_{(a_1,a_2) \in W} \bar{E}^{\text{SHW}}_{a_1 a_2} \cup \bigcup_{(a_1,a_2) \in C} \bar{E}^{\text{DT}}_{a_1 a_2} \cup \bigcup_{(a_1,a_2) \in F} \bar{E}^{\text{TA}}_{a_1 a_2} \cup \bigcup_{(a_1,a_2) \in H^1} \bar{E}^{\text{CC1}}_{a_1 a_2} \cup \bigcup_{(a_1,a_2) \in H^2} \bar{E}^{\text{CC2}}_{a_1 a_2}.$$

The graph $G = (V, E)$ now represents the compatibilities between the possible departure configurations of the legs in the timetable. Each subset $V_a \subseteq V$ represents the available departure configurations of leg $a \in A$ to choose from. By the definition of $\bar{E}^{\text{MC}}_a$, each of these subsets is a stable set in G. Accordingly, the required partition $\mathcal{V}$ of V is given by

$$\mathcal{V} := \{V_a \mid a \in A\}.$$

This yields the (CPMC)-instance $\mathcal{I} = (G, \mathcal{V})$, whose feasible set corresponding to model (1) we denote by $X(\mathcal{I})$. The decision variable $x_{ajd} \in \{0, 1\}$ then models the choice of a node $(a, j, d) \in V$ and consequently the choice of the departure configuration $(j, d) \in J_a \times D_a$ for the leg a. Furthermore, the multiple-choice constraints (1a) require the choice of exactly one departure configuration per leg, and the edges E represent the compatibilities between the departure configurations, which are enforced via constraints (1b). Altogether, model (1) is a complete model for the set of feasible timetable adaptations.

B.2 Reference Optimization Models

In combination with model (1) for the set of feasible timetables, the optimization problems resulting from the possible choices of objective functions introduced in Section 4 can all easily be stated as mixed-integer linear programs (MIPs) by introducing auxiliary variables, if needed. A reference optimization model for each of them is briefly presented in the following.

Optimization Model for the Railway Instances The optimization problem to minimize peak power consumption for the railway instances can be posed as a mixed-integer program by introducing continuous variables $w^t \in \mathbb{R}$ for the power consumption in each instant $t \in T$, in addition to a variable $z \in \mathbb{R}$ for linearizing the objective function. The problem can then be stated as follows:

$$\begin{aligned}
\min \quad & z && && (2a)\\
\text{s.t.} \quad & \sum_{t \in I} w^t && \leq 900z && (\forall I \in T_{15}) \quad (2b)\\
& \sum_{(a,j,d) \in V} \bar{p}^t_{ajd} x_{ajd} && \leq w^t && (\forall t \in T) \quad (2c)\\
& w^t && \geq 0 && (\forall t \in T) \quad (2d)\\
& w^t && \leq U && (\forall t \in T) \quad (2e)\\
& x && \in X(\mathcal{I}). && \quad (2f)
\end{aligned}$$

The objective function (2a) minimizes the maximum average power consumption over the planning horizon. It is computed via constraints (2b), (2c) and (2d), where (2b) averages the total power consumption in each $t \in T$ encoded in (2c) while (2d) incorporates the maximum with zero to model that recuperated energy can only be used if there is a possible receiving train at the same time. Constraint (2e) enforces that the total power consumption is never higher than the prescribed maximum instantaneous consumption. Finally, constraint (2f) requires the choice of a feasible timetable adaptation. In the following, we will refer to the above problem as Train Timetabling with respect to Maximum Average Power consumption with limited Instantaneous power consumption (TTMAPI).

Optimization Model for the Underground Instances The optimization problem for the underground instances can be modelled as follows, again using continuous linearization variables for the objective function:

$$
\begin{aligned}
\min \quad & \sum_{b \in B} \sum_{t \in T} w^{bt} && && && \text{(3a)}\\
\text{s.t.} \quad & \sum_{\substack{(a,j,d) \in X:\\ a \in A_b}} \bar{p}^t_{ajd} x_{ajd} && \leq && w^{bt} \qquad (\forall b \in B)(\forall t \in T) && \text{(3b)}\\
& w^{bt} && \geq && 0 \qquad (\forall b \in B)(\forall t \in T) && \text{(3c)}\\
& x && \in && X(\mathcal{I}). && \text{(3d)}
\end{aligned}
$$

Variables $w^{bt} \in \mathbb{R}$ model the total power consumption occurring in subnetwork $b \in B$ in time step $t \in T$. The objective function (3a) then models the total power consumption in the complete underground system. The individual power consumptions from each subnetwork and time step are computed via constraints (3b) and (3c), similar as in model (2). Constraint (3d) again ensures the choice of a feasible timetable adaptation.

Overall, the presented timetabling models (2) and (3) amount to optimizing a convex piecewise linear objective function over the CPMC-polytope. Efficient methods to solve the underlying two NP-hard problems have been proposed in [BGMS18, BMS21, BGMM21] and [BGM20] respectively.

Optimization Models for Further Possible Objective Functions Minimizing the gross power consumption of a TOC can easily be modelled by replacing $\bar{p}^t_{ajd}$ by $\max\{\bar{p}^t_{ajd}, 0\}$ in model (2). Note that if constraint (2e), which is of higher interest to the infrastructure manager (IM) than the TOC, is removed at the same time, this allows for a drastic simplification of model (2) with only a fraction of the necessary constraints and variables, as outlined in [BMS17, BMS21].

The power fluctuation induced by a railway timetable can be minimized via model (2), where the objective function needs to be replaced by

$$\sum_{t \in T} \|w^t - m\|_p,$$

with an auxiliary variable m that represents a mean power value according to some p-norm. From this family of optimization problems, two cases lead to MIPs, namely $p = 1$ and $p = \infty$, where in the latter case the problem the objective function may be simplified to

$$\max_{t \in T} w^t - \min_{t \in T} w^t,$$

i.e. the auxiliary variable m may be dropped. With $p = 1$, we minimize the summed deviations of the power consumption from its median value over the planning horizon, while for $p = \infty$ the maximum difference between the maximal and the minimal power consumption occurring at any point in time is minimized.

C Details on the Instances

We took great care to compile the extensive real-world data we obtained from our industry partners into representative problem instances for energy-efficient railway timetabling. In the following, we describe how precisely the data was compiled, which assumptions have been taken and which particular properties these instances possess.

C.1 Railway Instances

The data for the German railway network was provided by Deutsche Bahn AG. As no data on the physical railway network was available in our joint project, we took the assumption that the trains travel on tracks directly connecting the stations they service consecutively. This construction leads to the artefact that regional trains and long-distance trains travel on separate sets of tracks in many cases, as the former have many intermediate stops between large stations which the latter do not have. However, we consider our instances as sufficiently accurate to assess the potential of the overall approach. Note that this assumption only pertains to our interpretation of the available data, not to the optimization models presented in Section B. The latter can handle the case of mixed track use if we stipulate that the optimization leave the assignment of trains to tracks untouched; for practical computations this requires, of course, to have an ordered list of all trains passing a certain track during the planning horizon. In Figure 4, we show the connections between the stations established in the timetable.

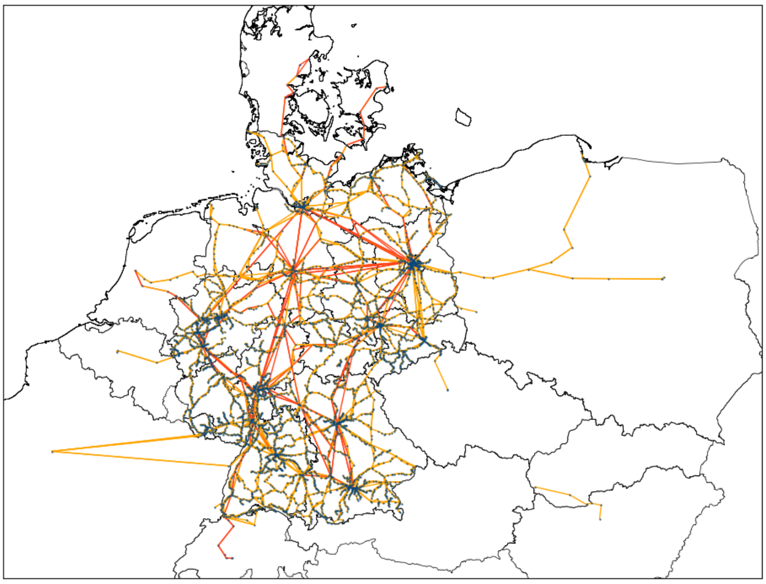

Figure 4: The connections between the railway stations established in the timetable – regional traffic in *light orange* and long-distance traffic in *dark orange*, train stations in *blue*

The Power Profiles In order to obtain data for the power consumptions of the trains, we created representative power profiles for each leg via a simulation depending on the type of train, the travel time and the heights of start and end station. To this end, we grouped the trains into four classes, which we name in brief after their main representative train type. These classes are the long-distance trains Intercity-Express (ICE), the two regional train types Regional-Express (RE) and Regionalbahn (RB) as well as the urban trains S-Bahn (S). The two classes RB and RE differ in that the former stops at every station, while the latter only stops at somewhat bigger stations. The power profiles we generated take into account typical train characteristics according to this classification and incorporate the necessary acceleration power (based on simplified velocity profiles), the power required to overcome the downhill force (based on the average slope from the start to the end of each trip) as well as the power required to overcome the rolling and the air friction. Note that absent any data on the locomotives used on the individual lines, we have assumed that all trains use electric traction and that they are able to recuperate energy. In this sense, the results we present in Section 6 represent a best-case scenario. A typical velocity and power consumption profile for an ICE train as we have used it for our computations is shown in Figure 5.

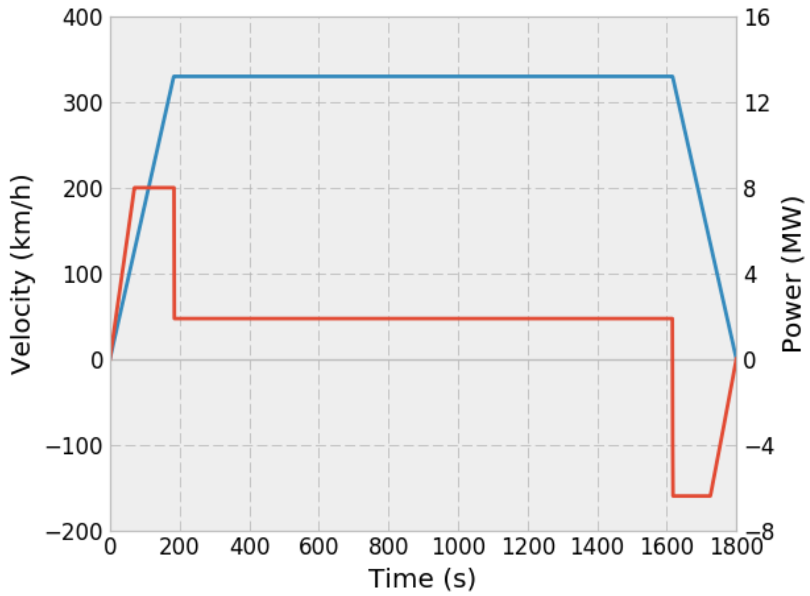

Figure 5: The velocity profile (*blue*) and the power consumption profile (*red*) for an ICE train running on an even track (i.e. start and end station are on the same height) and a journey time of 30 minutes

Constraints To ensure that the adjusted timetable is valid, we include the following constraints.

The *minimum headway times* were taken from [Pac16, Table 5.4] and rounded up to full minutes. They are displayed in Table 8, subsuming class S under RB (S-Bahn and Regionalbahn respectively, see below).

	ICE	RE	RB
ICE	4	3	3
RE	7	4	4
RB	9	7	4

Table 8: Minimum headway times for each pair of train types in minutes. The rows refer to the train departing earlier, while the columns refer to the following train.

The *minimum dwell time* for each train in each of its stations was chosen as the time between arrival and departure in the current timetable.

Finally, for a given train arriving at a station, we consider all trains departing between 5 and 15 minutes after this arrival in the current timetable as *connecting trains*. For such a pairing, we require that the current transfer time be changed by at most $\pm$ 3 minutes and stay between 5 and 15 minutes.

All these assumptions for feasible timetable adjustments and constraints are explicitly coded into the departure configurations available for each leg in a given instance.

C.2 Underground Instances

The topologies of the railway network and the power supply network in our instances were provided by VAG Verkehrs-Aktiengesellschaft. They are based on the actual underground network of Nürnberg, Germany, which consists of three underground lines supplied by a power network consisting of four subnets. Driverless operation of the lines U2 and U3 has been in place since 2008. Due to the distinction between manually and automatically operated underground lines, we created three instances for each timetable: *line U1*, *lines U2 and U3*, and *all lines*.

The Power Profiles In order to obtain the power consumption profiles of the trains on each leg, we parsed the tachograph data of several trains over multiple months in order to obtain sufficiently many samples per trip. The measurements include current speed, distance travelled and the acceleration control signal. We used these together with typical train characteristics like weight, electrical power of the engines, current flow, voltage and recuperation efficiency in order to estimate the power consumption profiles of the trains.

For a small subset of trains it was possible to directly measure the power consumptions (and recuperations) at the engines via a device specifically installed for our project. A comparison to our approximated power consumption profiles showed that in almost all cases, our approximation is very close to reality. An example comparison is shown in Figure 6. We can clearly

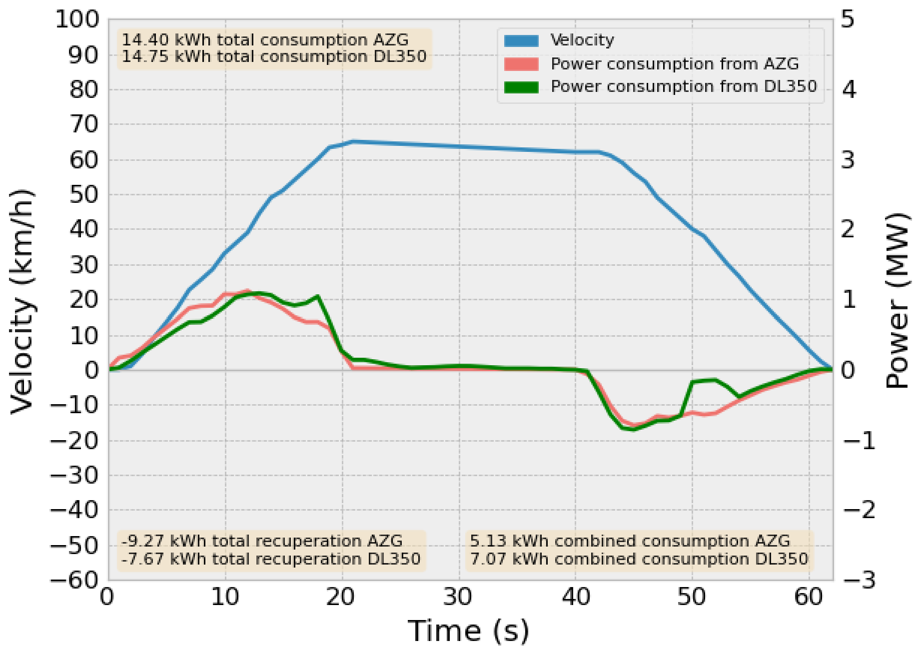

Figure 6: Example comparison for an underground profile measured in 2019 between the stations 'Opernhaus' and 'Plärrer'

see that the curves representing the estimated (AZG) and the measured (DL350) power consumptions coincide for the most part, with small deviations around the 8-, 18- and 50-second marks. The cause of the first two deviations is still unknown. However, we know that the third and largest deviation is caused by no other train being available to use the recuperated energy. In this case, the energy is not fed back into the power network (and thus not measured by the DL350 device), but instead discharged via heat dissipation. For each combination of train type, track and travel time used in the nominal timetable or its adjusted version, we looked at the set of power profiles for this combination, computed the total power consumption for each profile assuming full usage of the recuperation energy available and chose the one with the median value for the total power consumption.

Constraints In order to ensure that the adjusted timetable is realizable in practice, we include the following constraints.

The *headway time* between consecutive trains on the same track is generally set to 100 seconds. In the Nürnberg underground system, there exists exactly one pair of stations with only a single track in between. As the order of trains is fixed, it suffices to ensure for each pair of potentially conflicting trains, that the first train arrives at its destination before the second train departs.

The *dwell time constraints* are derived from minimal dwell times defined by the company planners for the end stations of each track. This is due to the fact that trains arriving at the same station may carry different passenger loads, depending on which station they come from. The minimum dwell times at each station typically range from 20 seconds to 40 seconds.

The *terminal turnaround constraints* incorporate station-specific times ranging from 112 seconds to 420 seconds. Typically, a longer time is required if the train has to actually turn around or drivers need to change, whereas shorter times are required if the train simply stays in place and then departs in the opposite direction.

In rare cases, the nominal timetable provided by the company planners did not satisfy the above constraints. We handled this by relaxing the constraint as little as possible such that the nominal timetable becomes feasible. For example, if a minimum dwell time is violated by one second, then we reduce it by that one second and add the relaxed version of the constraint to the instance.

Connection constraints at a station are required in two cases. Firstly, if the next train arrives at least 10 minutes later, and secondly, if there is no next train. If a passenger simply needs to cross to the opposing side of the platform, the connection time is set to 60 seconds, whereas changing floors requires a connection time of 180 seconds. The connection constraints in the instances were parsed from the nominal schedule according to the rules above. More precisely, for any connection candidate we checked the following criteria in the given order to determine how the connection was handled in each particular case:

1. Time between arrival of the first train and departure of the second train.
2. Time between departure of the first train and departure of the second train.
3. Time between arrival of the first train and departure of the second train minus the minimal dwell time of the first train.

In the latter case, we increased the connection time by the minimal dwell time.

We point out that the above constraints yield a complete characterization of timetable adjustments which are actually practically realizable in the Nürnberg underground system. In fact, the timetables we optimized using model (3) together with the above data were validated and tested in multiple evaluation runs of the trains in the system over a whole day. They fulfil all necessary operational timetable requirements of VAG.

Acknowledgements

We thank our collaborators at our project partners Deutsche Bahn AG and VAG for providing us with data and expertise. In particular, on the side of Deutsche Bahn AG we thank Michael

Hauss, Michael Hock, Boris Krostitz, Hanno Schülldorf, Olesya Shcheglova and Nina Teumer. On the side of VAG, we would like to express our gratitude towards Heinz Fleig, Oliver Früh, Norbert Grob, Matthias Haase, Thomas Huber, Michael Kaiser, Norbert Müller, Markus Popp and Manfred Schönleben. Further, we thank Alexander Martin for supporting our research in these two projects. Finally, we acknowledge financial support by the Bavarian Ministry of Economic Affairs, Regional Development and Energy through the Center for Analytics – Data – Applications (ADA-Center) within the framework of "BAYERN DIGITAL II".