A Stochastic Optimization Approach to Energy-Efficient Underground Timetabling under Uncertain Dwell and Running Times

Patrick Gemander [1], Andreas Bärmann [2], Alexander Martin[2]

Abstract

We consider a problem from the context of energy-efficient underground railway timetabling, in which an existing timetable draft is improved by slightly changing departure and running times. In practice, synchronization between accelerating and braking trains to utilize regenerative braking plays a major role for the energy-efficiency of a timetable. Since deviations from a planned timetable may lead to unnecessarily high energy consumption during actual operation, we include operational uncertainties in our model to create a timetable that remains energy-efficient, even if deviations from the nominal timetable occur. To solve the problem we use a scenario expansion model in conjunction with a Benders decomposition approach. As an alternative to solving the Benders subproblems we present a heuristic sparse cut that can be computed efficiently. The resulting sparse-cut heuristic produces high-quality solutions on a set of real-world instances stemming from the Nürnberg underground system, outperforming the integrated mixed-integer programming approach as well as the basic Benders approach. Additionally, we evaluate two static recovery strategies—shortening dwell times as well as shortening dwell and running times—to determine the cost and benefit of handling delays using a simple static rule. In our experiments, we are able to reduce the energy consumption by up to 9.4 % and confirm that delay recovery via shortening dwell times is an energy-efficient and effective way to increase punctuality at low cost in terms of energy.

Keywords: Railway Timetabling, Energy Consumption, Integer Programming, Stochastic Optimization, Clique Problem, Multiple-Choice Constraints

Mathematics Subject Classification (MSC2020): 90-10, 90B20, 90B36, 90C11, 90C15, 90C27

[1] *Fraunhofer-Institut für Integrierte Schaltungen IIS, Gruppe Optimization, Nordostpark 93, 90411 Nürnberg, Germany,* patrick.gemander@fau.de

[2] *Friedrich-Alexander-Universität Erlangen-Nürnberg, Department of Data Science / Department Mathematik, Cauerstraße 11, 91058 Erlangen, Germany,* {andreas.baermann, alexander.martin}@fau.de

1 Introduction

In recent years, energy-efficient railway timetabling has become a highly relevant topic in the research community due to rising energy prices and growing concerns about climate change. Underground rail systems in particular pose a rapid, efficient, and safe backbone to public urban passenger transport. This energy-efficient mode of transportation can be further improved by energy-efficient driving strategies as well as energy-aware timetabling. In an underground rail system without energy storage, the energy from regenerative braking is lost, if at the time there is no nearby train to utilize it. Instead of installing an energy storage at additional cost for procurement and maintenance, the energy-efficiency of underground operation can be increased by improving an existing timetable draft via small adjustments to the departure times and running times of each train. This can positively impact the total energy consumption in two ways. First, longer running times typically result in lower energy consumption. Second, a braking train can feed back energy into the power network if there is another train that can use this energy at the same time; this is called *recuperation*. Therefore, synchronization of braking and accelerating trains, which is mainly achieved by proper selection of departure times, increases the energy efficiency of a timetable. However, if deviations between the planned timetable and its actual execution are not accounted for, they may impair the synchronization and therefore lower the energy efficiency of the timetable. Consider the example in Figure 1, showing a comparison of the energy consumption between a planned underground timetable (left) that has been optimized without accounting for operational deviations and the corresponding actually executed timetable (right). We can observe that the deviations significantly impact the total energy consumption as well as the 15-minute peak-load averages, which are relevant cost factors for a train-operating company. Instead of creating a timetable that is energy-efficient under the assumption that all departure and running times are executed as planned, we now include uncertainties in the model to create a timetable that remains energy-efficient when typical deviations occur during operation.

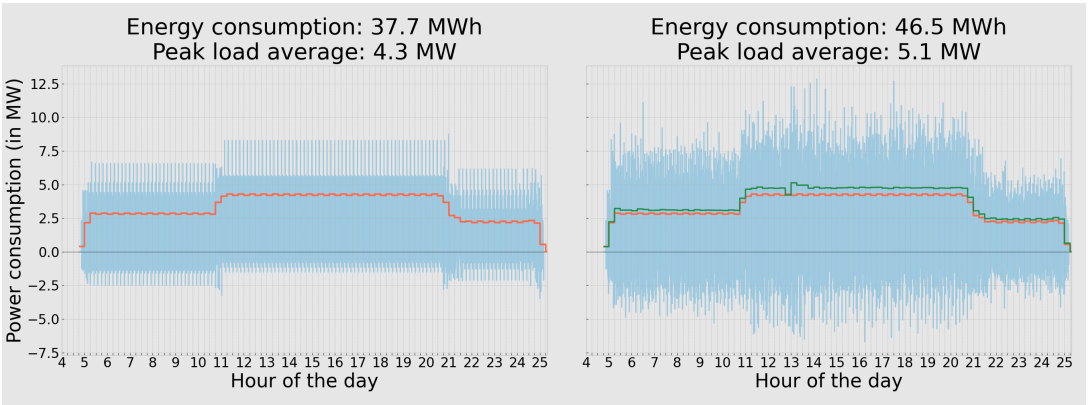

Figure 1: Comparison of a planned (left) and the corresponding executed (right) underground timetable. The energy consumption in each second is shown in blue, the 15-minute average energy consumption for the planned timetable in red, and for the executed timetable in green.

Related Literature General approaches to energy-saving train operation include energy-efficient driving strategies, operational measures, and energy-efficient timetabling. For surveys on energy-efficient driving strategies we refer to Feng et al., 2013, and more recently, Scheepmaker et al., 2017 along with Yin, Tang, Yang, Xun et al., 2017. Examples of operational measures are speed limits, as done by Hasegawa et al., 2014 and Kimura and Miyatake, 2014 or voltage control, as done by Raghunathan et al., 2014.

The two main tools for energy-efficient timetabling are adjustments of departure times, also called *dwell-time control*, and adjustments of the speed profiles of the trains, also called *running-time control*. Sansó and Girard, 1997 use dwell-time control for instantaneous peak power reduction, which plays a key role in the stability of the power network. The idea is to desynchronize trains while respecting operational timetabling constraints, such as specific dwell times for passenger exchange in stations or safety distances between trains. To the best of our knowledge, they introduce the first mixed-integer programming (MIP) model that uses similar timetable adjustments as our approach. Also related to peak power management via dwell-time control are the works by K.-M. Kim, Oh et al., 2010 and by K.-M. Kim, K.-T. Kim et al., 2011, who both use a much simpler model without operational timetabling constraints, but integrate the efficient use of recuperated energy, as well as the work by Chen et al., 2005 utilizing a genetic algorithm. Fournier et al., 2012 propose a hybrid genetic/linear programming algorithm to reduce total energy consumption by subtly modifying dwell times.

Running-time control without dwell-time control is used by e.g. Albrecht, 2010 to directly reduce the energy consumption, and many approaches combine dwell- and running-time control. For example, Peña-Alcaraz et al., 2012 use a power-flow model to determine a power-saving factor for each synchronization event between two trains, which is subsequently used in a timetabling model to maximize the duration of the synchronization events. Gong et al., 2014 also use a two-step approach that first optimizes the timetable using a genetic algorithm and subsequently handles disturbances in an energy-efficient manner. In contrast, Su et al., 2013 propose an integrated model to simultaneously optimize the timetable and speed profiles, but disregard recuperation. The dynamic approach by Li and Lo, 2014 includes a forecast of passenger demand when adjusting headway and cycle times. Similarly, Yin, Tang, Yang, Gao et al., 2016 consider uncertain passenger demand, but focus on real-time rescheduling. Zhou et al., 2017 combine trajectory optimization and timetabling using a shortest-path formulation in a discretized space-time-velocity graph. Similarly, P. Wang and Goverde, 2019 consider the same combination using the trajectory optimization to redistribute running times between different line segments of multiple train routes.

The base model introduced by Sansó and Girard, 1997 has been used successfully by Bärmann, Martin et al., 2017 and Bärmann, Gellermann et al., 2018 in the context of peak power reduction for nation-scale railway timetable adjustments via dwell-time control. Bärmann, Martin et al., 2021 use the same timetabling model with alternate, more complex objective functions and handle the additional complexity using a Benders decomposition approach. In the context of underground timetabling, Bärmann, Gemander et al., 2020 add running-time control to the model by allowing a set of discrete running times for each train. Y. Wang et al., 2021 also consider discrete sets of dwell and running times, but choose a different model and also incorporate rolling stock circulation planning in their mixed-integer nonlinear programming model. In the instance library *EETTlib*, Bärmann et al., 2022 summarize their timetabling model together with various objective functions and, most importantly, provide a timetabling instance set as basis for future work. Similar to the approach of Restel et al., 2021, who distribute time reserves to handle high energy-consumption caused by disruptions, we consider the impact of operational uncertainties in combination with simple static delay recovery strategies in this paper.

For an overview of robustness at different stages in the railway planning sequence we refer to the survey by Lusby et al., 2018. The focus of our work is the timetabling stage, where the lines, line frequencies, and train routes are fixed. For this stage, the survey by Cacchiani and Toth, 2012 highlights, among others, stochastic programming as a promising approach to robustness. In particular, they cite the work by Kroon et al., 2008 who also use a stochastic optimization model with sample average approximation to solve their timetabling problem.

In short, this work connects three key ideas from the literature. First, our current version of the timetabling model, which was originally proposed by Sansó and Girard, 1997. Second, the scenario-based approach to stochastic programming as used by Kroon et al., 2008. Third,

Kliewer and Suhl, 2011 have shown that both sophisticated real-time rescheduling as well as simple rule-based recovery strategies have their advantages and we therefore consider two simple rule-based recovery strategies in this work. For completeness sake, we would like to mention the overview on recovery models and real-time railway rescheduling by Cacchiani, Huisman et al., 2014.

Contribution We advance the approach of adjusting departure and running times in an existing timetable draft to improve energy-efficiency by including operational uncertainties in the optimization problem. For each departure of a certain train from a certain station at a certain time, we derive from real-world data the deviations between the nominal dwell and running times in the planned timetable and the actual dwell and running times that occur during operation. A scenario is a set consisting of one dwell and one running time deviation for each departure in a given timetable. In our *integrated stochastic MIP model*, we link the nominal timetable to a set of scenarios and compute, much in the spirit of sample average approximation (see Kleywegt et al., 2002), a nominal timetable that minimizes the energy consumption over all scenarios. In addition to the case where no further action is taken upon occurrence of a delay, we also consider variants of the model to *study the impact of two delay recovery strategies*. The first strategy minimizes delays by shortening dwell times, while the second one minimizes delays by shortening dwell and running times. We show that the problem can be solved using a *Benders decomposition approach*, where we find a feasible nominal timetable in the master problem and calculate its energy consumption by solving a dual subproblem for each scenario. Additionally, we introduce a *heuristic approach* where instead of solving the large dual subproblem in the Benders decomposition approach, we efficiently compute a sparse Benders-like cut. Moreover, we offer some *additional improvements* to the sparse cut itself and the heuristic approach as a whole. In a series of computational experiments on real-world instances, we first show empirically that the sparse-cut heuristic outperforms both the integrated MIP model and the Benders decomposition approach. Second, we evaluate the benefits of the additional improvements. Finally, we compare the three recovery strategies 'no recovery', 'dwell-time shortening', and 'dwell- and running-time shortening' to assess the trade-off between punctuality and energy efficiency.

Structure of the Paper In Section 2, we describe the real-world timetabling problem. Then we present our corresponding stochastic optimization MIP model, including the variants handling the alternate recovery strategies, in Section 3. The Benders decomposition approach and the closely related sparse-cut heuristic are laid out in Section 4, where we also summarize our improvements to the heuristic approach. Finally, in Section 5, we present our computational results and end with our conclusions in Section 6.

2 Problem Description

We consider the problem to increase the energy efficiency of a given timetable draft by slightly adjusting departure and running times. Let $T := \{1, \dots, \overline{T}\}$ be the planning horizon discretized in seconds. A *train* is a vehicle serving an underground line from the first to the last station and the finite set of trains to be scheduled is denoted by $I := \{1, 2, \dots\}$. A *leg* is a single trip of a given train from one station to the next and for a train $i \in I$ the finite set of its legs is denoted by $J_i := \{1, 2, \dots\}$. We assume w.l.o.g. that the legs in J_i are indexed such that $j \in J_i$ is the j-th leg travelled by train i. To keep notation concise, we may refer to leg $j \in J_i$ of train $i \in I$ simply as 'leg (i, j)' and also denote by J the set of all tuples (i, j) that represent a leg. The set of possible *departure configurations* (d, r) of leg (i, j) is denoted by C_{ij}, where $d \in T$ is the planned departure time and $r \in \mathbb{N}$ the planned running time.

A feasible timetable then consists of one chosen departure configuration $(d, r) \in C_{ij}$ for each leg $(i, j) \in J$, such that the following operational constraints are satisfied. *Headway time constraints* ensure sufficient buffer time between consecutive legs on the same track in the same direction. *Single-track headway time constraints* ensure sufficient buffer time between consecutive legs on the same track in opposing direction. Typically, this only happens if two stations are connected by a singular track instead of one track per direction. *Dwell time constraints* ensure sufficient time between arrival and departure at a station for passenger boarding and alighting. *Terminal turnaround constraints* ensure sufficient buffer time between arrival at a terminal station and departure from the same station in the opposite direction. Finally, *connection constraints* ensure that the connection times for passengers changing lines remain in a certain time window when adjusting the timetable.

Our formulation uses a binary variable x_{ijdr} for each leg $(i, j) \in J$ and departure configuration $(d, r) \in C_{ij}$, indicating whether (d, r) is the planned departure configuration of leg (i, j). In a feasible solution, for each leg $(i, j) \in J$ exactly one of the variables $x_{ijdr}, (d, r) \in C_{ij}$ is set to one such that the constraints listed above are satisfied. To keep the notation concise, we simply denote by X the set of binary solution vectors $x \in \{0, 1\}^{\sum_{(i,j)\in J} |C_{ij}|}$ representing feasible timetables. Our MIP formulation of $x \in X$, stated in detail in Appendix A, is based on the graph model by Bärmann et al., 2022, Appendix B together with the *stable set formulation* laid out in Bärmann, Gemander et al., 2020, Section 3.

Let $x \in X$ be a feasible timetable. In the nominal case, i.e. all departure times and running times are executed exactly as planned, its total energy consumption can be calculated as follows. Denote by $p_{ijdrt} \in \mathbb{R}$ the energy consumption at time $t \in T$ of leg $(i, j) \in J$ executing departure configuration $(d, r) \in C_{ij}$. We then introduce continuous variables $z_t, t \in T$ representing the total energy consumption of all trains at time t. In our model, recuperation is only considered within a time step and braking energy that cannot be used by another train in this time step is lost. Therefore, for each $t \in T$, the total energy consumption z_t of all trains is non-negative. The optimization model for the nominal case can then be stated as follows:

$$\min_{x,z} \sum_{t \in T} z_t \tag{1a}$$

$$\text{s.t.} \sum_{(i,j)\in J, (d,r)\in C_{ij}} p_{ijdrt} x_{ijdr} \leq z_t, \quad t \in T, \tag{1b}$$

$$z_t \geq 0, \quad t \in T, \tag{1c}$$

$$x \in X. \tag{1d}$$

The trains in an underground system are powered via different power subnetworks, depending on their location in the system. To model the subnetworks, one can simply add a second index to the variables z_t and state the Constraint (1b) for each combination of time and subnetwork. Since a train can draw and recuperate power from exactly one subnetwork at a time, depending on the train's location, no further reformulation is required. We omit this additional index in favour of readability.

As showcased by the comparison in Figure 1, the deviations between the nominal and actual dwell and running times have a major impact on energy consumption. To address this problem, we derive a set of scenarios from real-world operational data and minimize the expected energy consumption over a set of scenarios S. In our approach, a scenario s is a set of deviations from nominal dwell and running times. We denote by $\delta_{sij} \in \mathbb{Z}$ the deviation between nominal and actual dwell time before departure of leg $(i, j) \in J$ in scenario s. The deviation between nominal and actual running time on leg $(i, j) \in J$ in scenario s is denoted by $\rho_{sij} \in \mathbb{Z}$. Based on the nominal timetable and a given scenario s with deviations $(\delta_{sij}, \rho_{sij}), (i, j) \in J$, we can then derive the actual timetable.

For example, consider the first leg $(i, 1)$ of some train $i \in I$ with nominal departure configuration $(0, 50) \in C_{i1}$ and a scenario s with $(\delta_{si1}, \rho_{si1}) = (5, 3)$. Then the actual departure config-

uration of this leg becomes $(5, 53)$, which results in a delay of eight seconds at time of arrival. Let the second leg $(i, 2)$ of the same train have nominal departure configuration $(70, 50) \in C_{i2}$ and let $(\delta_{si2}, \rho_{si2}) = (-1, 2)$. Then the nominal dwell time before leg $(i, 2)$ amounts to 20 s and the actual departure configuration of this leg becomes $(77, 52)$, as it departs 19 s after leg $(i, 1)$ arrives at time step 58. The leg $(i, 2)$ then arrives with a total delay of 9 s, which in turn carries over to the next leg $(i, 3)$.

We assume the first leg $(i, 1)$ of each train $i \in I$ does not carry over any preexisting deviations from earlier legs representing the same vehicle serving the same line in the opposite direction before turnaround. All other legs are shifted based on the scenario and the deviation at time of arrival of the previous leg. This distinction is based on the observation that turnaround times typically offer enough buffer time to recover from delays.

When determining running times, a special case occurs if the deviation ρ_{sij} is negative, which means that train i on leg j runs faster than planned. For example, let $\rho_{sij} = -1$ for some leg $(i, j) \in J$ in some scenario s and let $C_{ij} = \{(0, 55), (0, 60)\}$ be the plannable nominal departure configurations. Further, let the minimal running time of this leg be 55 s, i.e. the train physically cannot reach the next station any faster. If the departure configuration $(0, 55)$ is chosen as the nominal departure configuration, then the train cannot reach the next station within 54 s and the running time remains 55 s. It is also conceivable to deal with this issue by removing any departure configuration that leads to an invalid actual departure or running time. However, this approach is overly restrictive as it may remove departure configurations that are perfectly valid in most scenarios and only lead to slight modelling inaccuracies in very rare cases.

To summarize, the actual departure configuration (d', r') of some leg (i, j) in scenario s is given by

$$d' = d + k + \delta_{sij}, \tag{2}$$

$$r' = \max\{\underline{r}_{ij}, r + \rho_{sij}\}, \tag{3}$$

where $d \in T$ and $r \in \mathbb{Z}$ are the nominal departure and running time of the leg, $k \in \mathbb{Z}$ the preexisting deviation at time of arrival of the previous leg, and $\underline{r}_{ij}$ the minimal running time on leg (i, j). If the leg (i, j) is the first one after turnaround, then we simply have $k = 0$. Based on the actual departure and running times, the energy consumption of the actual timetable can be calculated in the same way as for the nominal timetable.

In principle, the deviations may depend on the nominal dwell and running times. In this work, we assume the deviations to be independent of the chosen nominal dwell and running times for three reasons. First, the adjustments to the dwell and running times are relatively small and it can therefore be expected that the distribution of deviations remains an accurate approximation for altered dwell and running times. Second, our model respects the minimal dwell and running times that are being used by our industry partner VAG to ensure smooth operation, i.e. , the feasible timetables do not lead to additional deviations that are of 'structural nature'. In other words, it is not possible to, e.g. , shorten the dwell time below a critical threshold that would incur regular extension of train stops. Finally, our model explicitly includes a kind of dependency between minimal running time and deviation (see Equation 3) and for minimal dwell times, the deviations are too small to lead to issues.

Table 1 summarizes the notation from this and the next section. In the next section, we present an integrated model to find a nominal timetable that minimizes the expected energy consumption for a given set of scenarios S.

3 Stochastic Optimization MIP Model

We first describe in Section 3.1 our integrated stochastic optimization MIP model to compute a nominal timetable with minimal expected total energy consumption for the scenario set S.

Table 1: Overview of the notation from Section 2 (top) and Section 3 (bottom)

Symbol	Meaning
T	Planning horizon
I	Set of trains
J_i	Set of legs of train $i \in I$
J	Set of all tuples $(i,j), i \in I, j \in J_i$ representing a leg in the timetable
(d,r)	Departure configuration consisting of departure time d and running time r
C_{ij}	Set of plannable (nominal) departure configurations (d,r) of leg $(i,j) \in J$
p_{ijdrt}	Energy consumption of leg $(i,j) \in J$ executing departure configuration $(d,r) \in C_{ij}$ at time $t \in T$
x_{ijdr}	Binary variable representing the nominal departure configuration $(d,r) \in C_{ij}$ of leg $(i,j) \in J$
X	Set of binary vectors $x \in \{0,1\}^{\sum_{(i,j)\in J}\lvert C_{ij}\rvert}$ representing feasible timetables (see Appendix A for details)
z_t	Continuous variable representing the combined energy consumption of all trains at time $t \in T$
S	Set of scenarios s
δ_{sij}	Deviation from nominal dwell time before leg (i,j) in scenario s
ρ_{sij}	Deviation from nominal running time of leg (i,j) in scenario s
$\underline{r}_{ij}$	Minimal running time of leg $(i,j) \in J$
k	Deviation between nominally planned and actual arrival time
K_{sij}	Set of possible deviations for leg $(i,j) \in J$ in scenario $s \in S$
C_{sij}	Set of possible actual departure configurations of leg $(i,j) \in J$ in scenario $s \in S$
z_{st}	Continuous variable representing the combined energy consumption of all trains at time $t \in T$ in scenario $s \in S$
y_{sijdr}	Binary variable representing executed departure configuration $(d,r) \in C_{ij}$ of leg $(i,j) \in J$ in scenario $s \in S$
κ_{sijk}	Binary variable representing leg $(i,j) \in J$ arriving with deviation $k \in \mathbb{Z}$ in scenario $s \in S$
$\underline{h}_{ij}$	Minimal dwell time after leg $(i,j) \in J$

Afterwards, we present two model variants in Section 3.2 that represent two recovery strategies to handle scenario-induced delays. The first one recovers delays by shortening dwell times, whereas the second one shortens both dwell and running times.

3.1 Stochastic Optimization Model without Recovery

We model the problem as a two-stage stochastic linear program (see e.g. Kall and Wallace, 1994). The first-stage problem is Model (1) without its objective function, i.e. we remove all variables $z_t, t \in T$ and the corresponding Constraints (1b). The second-stage subproblem uses for each scenario $s \in S$ binary variables y_{sijdr} to model the actually executed timetable in scenario s and continuous variables z_{ts} to model the corresponding energy consumption. The variables z_{ts} are linked to the variables y_{sijdr} in the same way as variables z_t to x_{ijdr} in Constraints (1b).

It remains to link the nominal timetable represented by the x-variables to the actually executed timetables represented by the y-variables. First, we introduce a set of auxiliary binary variables κ_{sijk} to represent whether leg (i,j) arrives with deviation $k \in \mathbb{Z}$ from the nominal timetable in scenario s. Then, for each leg $(i,j) \in J$ and each nominal departure configuration $(d,r) \in C_{ij}$, we add for each possible actual departure configuration (d',r') the constraint

$$x_{ijdr} + y_{sijd'r'} - 1 \leq \kappa_{sijk}, \tag{4}$$

where

$$k := d' + r' - d - r. \tag{5}$$

The constraints ensure that if departure configuration (d,r) is chosen for the nominal timetable and departure configuration (d',r') occurs during actual operation in scenario s, then variable κ_{sijk} is set to one, representing a deviation of k seconds from the nominal timetable. We denote by K_{sij} the (finite) set of possible deviations for leg (i,j) in scenario s and to keep notation concise, we use $K_{si0} := \{0\}$ as set of deviations before the first leg $(i,1)$ of each train $i \in I$.

Let $\underline{r}_{ij}$ be the minimal running time for leg (i,j). The linking between nominal and actual departure configuration of some leg $(i,j) \in J$ in scenario s with deviation $k \in K_{sij-1}$ at previous arrival can then—using Equalities (2) and (3) to determine d' and r'—be stated as

$$x_{ijdr} + \kappa_{sij-1k} - 1 \leq y_{sijd'r'}. \tag{6}$$

The constraint ensures that if departure configuration (d, r) is chosen for the nominal timetable and the leg arrives with deviation k during actual operation in scenario s, then variable $y_{sijd'r'}$ is set to one, representing that departure configuration (d', r') occurs in scenario s. Analogously to arrival deviations K_{sij}, we denote by C_{sij} the (finite) set of possible actual departure configurations for leg $(i, j) \in J$ in scenario $s \in S$.

Due to limited operational data, we are restricted to a very low number of scenarios in our application and the resulting timetables may be too strongly adjusted to statistical outliers. The following example illustrates the issue.

Example 3.1. *Consider a system consisting of two trains with one leg per train, i.e.* $J = \{(1,1),(2,1)\}$, *and a single scenario* $S = \{1\}$ *given by the deviations*

$$(\delta_{1,1,1}, \rho_{1,1,1}) = (0, 1),$$
$$(\delta_{1,2,1}, \rho_{1,2,1}) = (0, -1).$$

Further, let $C_{1,1} = \{(0,2),(0,3)\}$, $C_{2,1} = \{(5,2),(5,3)\}$, *and let the energy consumption for train* i, *departure time* d, *running time* r, *and time step* t *be given by*

$$p_{i,1,d,r,t} = \begin{cases} 5, & \text{if } t = d \text{ and } r = 1 \\ r, & \text{if } t = d \text{ and } r \in \{2,3,4\} \\ 0, & \text{otherwise.} \end{cases}$$

Clearly, an optimal solution will choose nominal running time two for the first train and three for the second train, resulting in respective actual running times three and two. The total energy consumption then amounts to $p_{1,1,0,3,0} + p_{2,1,5,2,5} = 3 + 2 = 5$. *If the single scenario is representative of the occurring deviations, then this is exactly the desired result. However, the trains may simply move one second faster or slower with equal probability and a second scenario could be given by*

$$(\delta_{2,1,1}, \rho_{2,1,1}) = (0, -1),$$
$$(\delta_{2,2,1}, \rho_{2,2,1}) = (0, 1).$$

In this second scenario, the chosen running times lead to actual running times one and four, which in turn result in a total energy consumption of $p_{1,1,0,1,0} + p_{2,1,5,4,5} = 5 + 4 = 9$. *Assuming equal probability of the two scenarios, we get an expected total consumption of* 7.

Consider choosing running time three for both trains, which yields actual running times four and two in the first scenario and two and four in the second scenario. In this case, either scenario leads to a total consumption of only 6. *This illustrates that the first scenario may not suffice to compute an optimal solution for typical deviation scenarios.*

Our industry partner VAG requires another type of constraint, which conveniently resolves this issue. More precisely, adjusting each individual departure configuration leads to an irregular timetable, which is undesirable from an operational point of view as well as the passenger's perspective. Instead, we synchronize the way in which different trains serving the same direction of the same line can be changed. Suppose the original timetable assigns equal dwell and running times to two such trains, e.g. both trains use a 50 s running time between stations one and two, then stop for 20 s at station two, then use a 60 s running time between stations two and three, and so forth, then we say these two trains use the same *journey profile*.

Let $i, i' \in I$ be two trains using the same journey profile and $(i, j), (i', j') \in J$ two legs of these two trains that connect the same stations. Denote by $\tilde{d}_{ij}, \tilde{d}_{i'j'}, \tilde{r}_{ij}, \tilde{r}_{i'j'}$ the respective departure and running times in the original timetable. The following constraints synchronize these two legs:

$$x_{ijdr} = x_{i'j'd'r'}, \quad (d, r) \in C_{i,j}, (d', r') \in C_{i',j'} : (d - \tilde{d}_{ij} = d' - \tilde{d}_{i'j'}) \wedge (r - \tilde{r}_{ij} = r' - \tilde{r}_{i'j'}), \qquad (7)$$

For example, let (i,j) and (i',j') be two such legs with $C_{ij} = \{(0,60),(5,60),(0,62)\}$, $C_{i'j'} = \{(300,60),(305,60),(300,62)\}$, and let $(0,60)$ and $(300,60)$ be the originally planned departure configurations. Then the constraints

$$\begin{aligned} x_{i,j,0,60} &= x_{i',j',300,60}, \\ x_{i,j,5,60} &= x_{i',j',305,60}, \\ x_{i,j,0,62} &= x_{i',j',300,62} \end{aligned}$$

ensure that the trains i and i' use the same journey profiles after optimization. We would like to note that this synchronization also maintains periodicity, since the departures maintain a five-minute difference after optimization.

To show a quick example how this can play out during optimization, adding the constraints $x_{1,1,0,2} = x_{2,1,5,2}$ and $x_{1,1,0,3} = x_{2,1,5,3}$ synchronizes the legs $(1,1)$ and $(2,1)$ in Example 3.1. This forces the solution to choose the same 'departure configuration change' for the deviations of both legs. As a result, choosing a running time of three for both trains becomes the optimal solution even if only the first scenario is considered.

In a sense, the synchronization constraints multiply the number of scenarios by the number of trains using the same journey profile, since each journey profile must be adjusted to the deviations of all trains using this journey profile. In the case of our application, this essentially turns a single scenario into 60 to 180 scenarios, depending on how many trains use a journey profile. Moreover, the additional variable identifications simplify the model by the same 'factor'. On the downside, the additional constraints may cut off otherwise feasible solutions.

The full stochastic optimization model can then be stated as follows:

$$\begin{aligned} \min_{x,y,z,\kappa} \quad & \sum_{s\in S, t\in T} z_{st} \\ \text{s.t.} \quad & \sum_{i\in I, j\in J_i, (d,r)\in C_{sij}} p_{ijdrt} y_{sijdr} \leq z_{st}, \quad s\in S, t\in T, && \text{(8a)} \\ & x\text{-}\kappa\text{-linking constraints (6)}, && \text{(8b)} \\ & x\text{-}y\text{-linking constraints (4)}, && \text{(8c)} \\ & 0 \leq z_{st}, \quad s\in S, t\in T, && \text{(8d)} \\ & \sum_{(d,r)\in C_{sij}} y_{sijdr} = 1, \quad s\in S, (i,j)\in J, && \text{(8e)} \\ & \sum_{k\in K_{sij}} \kappa_{sijk} = 1, \quad s\in S, (i,j)\in J, && \text{(8f)} \\ & \kappa_{s,i,0,0} = 1, \quad s\in S, i\in I, && \text{(8g)} \\ & \text{synchronization constraints (7)}, && \text{(8h)} \\ & y_{sijdr} \in \{0,1\}, \quad s\in S, (i,j)\in J, (d,r)\in C_{sij}, && \text{(8i)} \\ & \kappa_{sijk} \in \{0,1\}, \quad s\in S, (i,j)\in J, k\in K_{sij}. && \text{(8j)} \\ & x\in X. && \text{(8k)} \end{aligned}$$

Before moving on to the Benders decomposition approach, we want to present how to model two recovery strategies that can be used to counteract delays.

3.2 Alternate Recovery Strategies

Another important aspect of railway timetabling is the service quality as seen from the passenger's perspective, especially punctuality. One approach to improving punctuality is to use spare dwell times or both spare dwell and running times to recover from delays. Interestingly, both of these strategies allow for a model where the κ-variables are no longer required. This is due

to the fact that the actual departure configuration of a leg no longer depends on the deviation between planned and actual arrival time of the previous leg, which had to be represented using the κ-variables. Instead, they depend only on the nominal departure and running time of the leg itself and the actual departure configuration of the previous leg, which are already represented by their respective x- and y-variables. We model the two delay recovery strategies as follows.

Dwell Time Recovery If the dwell time between arrival of some leg $(i, j-1)$ and departure on the next leg (i, j) is longer than the minimum dwell time $\underline{h}_{ij-1}$ in this station, then the next departure can be advanced to make up for a delay. Let d be the planned departure time of some leg $(i, j) \in C_{ij}, j \geq 2$ and let $(d'', r'') \in C_{sij-1}$ be the actual departure configuration of the previous leg $(i, j-1)$. If there is no delay or the delay can be recovered by departing as planned, then the planned departure time remains unchanged and the actual departure time in scenario s is $d' = d + \delta_{sij}$. Otherwise, the departure is scheduled at time $d'' + r'' + \underline{h}_{ij-1}$ to satisfy the minimal dwell time $\underline{h}_{ij-1}$ after arriving at time $d'' + r''$ and recover as much of the delay as possible. Then the actual departure time in scenario s is $d' = d'' + r'' + \underline{h}_{ij-1} + \delta_{sij}$. Hence, the actual departure time d' of the leg (i, j) can be expressed by

$$d' = \max\{d, d'' + r'' + \underline{h}_{ij-1}\} + \delta_{sij}, \tag{9}$$

and the corresponding variables can then be linked by

$$x_{ijdr} + y_{sij-1d''r''} - 1 \leq y_{sijd'r'}, \tag{10}$$

where the running time r' is determined as in (3). To summarize, this recovery strategy can be modelled by using Constraints (10) and (9) instead of (4), (6) and (2). Finally, in the case of a first leg $(i, 1) \in J$, Constraint (10) simply reduces to

$$x_{i1dr} \leq y_{si1d'r'}, \tag{11}$$

with $d' = d + \delta_{si1}$ and r' again determined by (3).

Full Recovery This recovery strategy utilizes dwell and running times to recover from delays. We assume the new departure configuration is determined upon arrival of a train. As in the previous recovery strategy, we use Constraints (10) and (9), instead of (4), (6) and (2) to recover as much delay as possible from shortening the dwell time. Equation (9) yields the actual departure time d' and we can state the expected rescheduled departure time as $d' - \delta_{sij}$, which in turn results in an expected departure delay of $d' - \delta_{sij} - d$. To recover the expected departure delay by shortening the running time, replace (3) with

$$r' = \max\{\underline{r}_{ij}, r - (d' - d - \delta_{sij}) + \rho_{sij}\}. \tag{12}$$

In the case of a first leg $(i, 1)$, the expected delay is zero and we simply use (11).

It is theoretically possible to use the actual delay upon departure $d' - d$ to determine the running time of leg (i, j). This would reflect a planning decision at the time of departure of leg (i, j) instead of rescheduling at the time of arrival of leg $(i, j-1)$. However, this would force the planning system to instantly make decisions, which is not yet a widely achieved standard. Before embedding Model (8) in a Benders decomposition approach we close this section with a remark on the nature of the second stage problem.

Remark 3.2. *Within the framework presented by Kliewer and Suhl, 2011, who distinguish between rule-based recovery strategies and optimization-based recovery strategies, the three recovery strategies presented in this work are of a simple rule-based nature. As a result, the second stage problem can be interpreted as an 'adjustment' problem rather than an actual 'optimization' problem. However, the chosen framework of two-stage stochastic linear programs allows extension and application of the ideas presented in this paper to more advanced (optimization-based) recovery strategies, which, albeit exceeding the scope of this work, poses interesting possibilities for future research.*

4 Benders Decomposition Approach

To deal with the size and complexity of the model, we present a Benders decomposition approach in Section 4.1. Since the subproblems to compute the energy consumption in all scenarios remain relatively large, we then introduce a sparse-cut heuristic in Section 4.2 as well as a parameterized version of the heuristic in Section 4.3. Finally, in Section 4.4, we offer some additional improvements to the heuristic solution approach.

4.1 Basic Method

Given a nominal timetable and a scenario, we can efficiently compute the actual timetable and its energy consumption. Therefore, we keep only the nominal timetabling constraints including the synchronization constraints (7) in the master problem. A new nonnegative continuous objective variable $\theta \geq 0$ represents the total energy consumption over all scenarios according to the currently available Benders cuts. The initial master problem where no Benders cuts are available can then be stated as

$$\min_{x,\theta} \quad \theta \tag{13a}$$
$$\text{s.t.} \quad \theta \geq 0, \tag{13b}$$
$$\text{synchronization constraints (7)}, \tag{13c}$$
$$x \in X. \tag{13d}$$

For a solution $\widetilde{x} \in X$, a Benders cut of the form $a^\top x \leq \theta$ can be generated to link the timetable $\widetilde{x}$ to its expected energy consumption over all scenarios $s \in S$ as follows.

Let $\widetilde{x} \in X$ be a feasible timetable. The remaining subproblem to compute the true objective value of $\widetilde{x}$ of Model (8) can be stated as

$$\min_{y,z,\kappa} \quad \sum_{s\in S, t\in T} z_{st} \tag{14a}$$
$$\text{s.t.} \quad \sum_{i\in I, j\in J_i, (d,r)\in C_{sij}} p_{ijdrt} y_{sijdr} - z_{st} \leq 0, \quad s \in S, t \in T, \tag{14b}$$
$$x\text{-}\kappa\text{-linking constraints (6)}, \tag{14c}$$
$$x\text{-}y\text{-linking constraints (4)}, \tag{14d}$$
$$0 \leq z_{st}, \quad s \in S, t \in T, \tag{14e}$$
$$\sum_{(d',r')\in C_{sij}} y_{sijd'r'} = 1, \quad s \in S, (i,j) \in J, \tag{14f}$$
$$\sum_{k\in K_{sij}} \kappa_{sijk} = 1, \quad s \in S, (i,j) \in J, \tag{14g}$$
$$\kappa_{s,i,0,0} = 1, \quad s \in S, i \in I, \tag{14h}$$
$$y_{sijdr} \in \{0,1\}, \quad s \in S, (i,j) \in J, (d,r) \in C_{sij}, \tag{14i}$$
$$\kappa_{sijk} \in \{0,1\}, \quad s \in S, (i,j) \in J, k \in K_{sij}. \tag{14j}$$

Although the subproblem contains binary variables, we can show that solving its linear programming relaxation yields a unique solution that fulfils the integrality constraints.

Theorem 4.1. *Let $(\widetilde{x}, \widetilde{\theta})$ be a feasible solution to the master problem (13). Then there exists a unique optimal solution $\widetilde{y}, \widetilde{\kappa}, \widetilde{z}$ to the Subproblem (14) and the values of $\widetilde{y}$ and $\widetilde{\kappa}$ are integral.*

Proof. Let $(\widetilde{x}, \widetilde{\theta})$ be a feasible solution to the master problem, $s \in S$ and $i \in I$. We prove the theorem by induction over the set $\{j : (i,j) \in J_i\}$. As base case, consider the first leg $(i,1) \in J$ of train i. By construction of the problem we have $K_{s,i,0} = \{0\}$ and $\kappa_{s,i,0,0} = 1$.

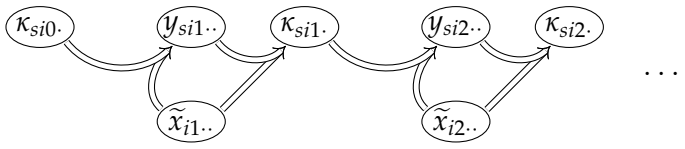

Figure 2: Implications between variable sets in Subproblem (14). For clarity in the diagram we denote by, e.g. , $y_{si0\cdot\cdot}$ the set of variables y_{si0dr} with $(d,r) \in C_{si0}$.

Assume there exist $j \in J_i$ and $k \in K_{sij-1}$ with $\kappa_{sij-1k} = 1$ and $\kappa_{sij-1k'} = 0, k' \in K_{sij-1} \setminus \{k\}$, which already holds for the base case. Since $\widetilde{x} \in X$ is a feasible timetable, there exists exactly one departure configuration $(\tilde{d}, \tilde{r}) \in C_{ij}$ with $\widetilde{x}_{ij\tilde{d}\tilde{r}} = 1$. The actual departure configuration (d', r') of leg (i,j) is determined by k, (2) and (3). The corresponding Inequality (6) then implies $y_{sijd'r'} = 1$ via

$$\underbrace{\widetilde{x}_{i,j,\tilde{d},\tilde{r}} + \kappa_{sij-1k} - 1}_{=1} \leq y_{sijd'r'}.$$

Let $k' = d' + r' - \tilde{d} - \tilde{r}$ be the deviation between nominal and actual arrival time of leg (i,j) in scenario s. The corresponding Inequality (4) then implies $\kappa_{sijk'} = 1$ via

$$\underbrace{\widetilde{x}_{ij\tilde{d}\tilde{r}} + y_{sijd'r'} - 1}_{=1} \leq \kappa_{sijk'}.$$

Additionally, we have $y_{sijdr} = 0$ for $(d,r) \in C_{sij}, (d,r) \neq (d',r')$ and $\kappa_{sijk''} = 0$ for $k'' \in K_{sij}, k'' \neq k'$ due to the respective multiple-choice constraints (14f) and (14g). Which allows us to repeat this inductive step if $j+1 \in \{j : (i,j) \in J_i\}$.

To illustrate how a solution is constructed, Figure 2 shows these implications for the first two legs of some train $i \in I$ in a given scenario $s \in S$. First, we have $K_{si0} = \{0\}$ and hence, the deviation variable κ_{si00} together with the nominal departure configuration variables $\widetilde{x}_{i1dr}, (d,r) \in C_{i1}$ implies the values of the actual departure configuration variables $y_{si1dr}, (d,r) \in C_{si1}$ via the corresponding Inequalities (6). Subsequently, the actual departure configuration variables $y_{si1dr}, (d,r) \in C_{si1}$ together with the nominal departure configuration variables $\widetilde{x}_{i1dr}, (d,r) \in C_{i1}$ imply the values of the deviation variables $\kappa_{si1k}, k \in K_{si1}$ via the corresponding Inequalities (4). Clearly, this can be repeated for the leg $(i,2)$ as well as all remaining legs of train i.

Altogether, we can for all $s \in S$ and all $(i,j) \in J$ uniquely determine the values of all variables $y_{sijdr}, (d,r) \in C_{sij}$ and all $\kappa_{sijk}, k \in K_{sij}$, which produces an integral vectors $\widetilde{y}, \widetilde{\kappa}$. This in turn yields a unique optimal solution $(\widetilde{y}, \widetilde{\kappa}, \widetilde{z})$ since each variable z_{st} will be set to $\max\{0, \sum_{(i,j)\in J,(d',r')\in C_{sij}} p_{ijd'r't} y_{sijd'r'}\}$, which completes the proof. □

By Theorem 4.1 we can therefore consider the subproblem as a linear program (LP) and apply the standard Benders decomposition (see e.g. Kall and Wallace, 1994, Section 3.2.2). Let α, β, β', γ, γ', γ'', π, π' be the respective vectors of dual variables corresponding to the energy consumption Constraints (14b), the linking constraints (14c) and (14d), the multiple-choice constraints (14f),(14g), (14h), and the upper bounds from (14i) and (14j). Given an optimal solution $(\widetilde{\alpha}, \widetilde{\beta}, \widetilde{\beta}', \widetilde{\gamma}, \widetilde{\gamma}', \widetilde{\gamma}'', \widetilde{\pi}, \widetilde{\pi}')$, we add the following cut to the master problem (13)

$$\sum_{sijdrk} \widetilde{\beta}_{sijdrk}(1 - x_{ijdr}) + \sum_{sijdrd'r'} \widetilde{\beta}'_{sijdrd'r'}(1 - x_{ijdr}) + \sum_{sij}(\widetilde{\gamma}_{sij} + \widetilde{\gamma}'_{sij}) + \sum_{si} \widetilde{\gamma}''_{si} + \sum_{sijdr} \widetilde{\pi} + \sum_{sijk} \widetilde{\pi}' \leq \theta. \tag{15}$$

Note that Theorem 4.1 also implies the subproblem to be feasible for all $\widetilde{x} \in X$ and we can therefore solve model (8) to global optimality via Algorithm 1.

Algorithm 1: LP-based Benders decomposition

```
Initialize the master problem as min_{θ∈ℝ₀⁺, x∈X} θ
do
    Calculate an optimal solution (θ̃, x̃) to the current master problem
    Calculate an optimal solution (α̃, β̃, β̃′, γ̃, γ̃′, γ̃″, π̃, π̃′) to the dual subproblem
      (parameterized in x̃)
    Add Constraint (15) to the current master problem
while (θ̃, x̃) violates (15)
return (θ̃, x̃)
```

We want to note that the analogues of Theorem 4.1 for the two model variants from Section 3.2 can be proven in the same way. As a result, Algorithm 1 can easily be adjusted to solve the two model variants via Benders decomposition.

4.2 Sparse-Cut Heuristic

Our experiments in Section 5.2 show empirically that solving the subproblem as LP is too expensive in terms of runtime and memory requirements. To deal with this issue, we replace the cut given by Inequality (15) with a cut that, given a solution $(\widetilde{x}, \widetilde{\theta})$ to the master problem (13), has two key properties. First, it can be computed efficiently and second, it represents the correct expected energy consumption θ of the timetable $\widetilde{x}$ it is computed from. However, in comparison to the original Benders cut given by Equality (15), this cut no longer corresponds to an optimal dual solution to the subproblem. As a result there may exist special cases where this cut overestimates the energy consumption of different timetables $x \in X \setminus \{\widetilde{x}\}$, which can result in cutting off feasible solutions. Although the resulting solution approach is therefore a heuristic, it produces satisfying results in our application.

Let $\widetilde{x}$ be a solution to the current master problem. By the arguments in the proof of Theorem 4.1 (see also Figure 2) we can uniquely determine the solution vector $\widetilde{y}$ of Subproblem (14) that represents the actual departure configurations corresponding to the chosen nominal departure configurations. For each $s \in S$ we then define the set of related planned and actual departure configurations as

$$\widetilde{R}_s := \big\{(i,j,d,r,d',r') : (i,j) \in J, (d,r) \in C_{ij}, (d',r') \in C_{sij} \text{ with } \widetilde{x}_{ijdr} = 1 \text{ and } \widetilde{y}_{sijd'r'} = 1\big\}.$$

By collecting from (14a) and (14b) the terms corresponding to the elements in $\widetilde{R}_s$, the total energy consumption over all scenarios can be expressed as

$$\sum_{s \in S, t \in T} \max\{0, \sum_{(i,j,d,r,d',r') \in \widetilde{R}_s} p_{ijd'r't} \widetilde{x}_{ijdr}\}. \tag{16}$$

Let

$$v_{st} := \begin{cases} 1, & \text{if } \sum_{(i,j,d,r,d',r') \in \widetilde{R}_s} p_{ijd'r't} \widetilde{x}_{ijdr} > 0, \\ 0, & \text{otherwise} \end{cases} \tag{17}$$

be the indicator of the total energy consumption of all legs in scenario s and time step t being positive. Using v_{st}, we can reformulate (16) to

$$\sum_{s \in S, t \in T} \sum_{(i,j,d,r,d',r') \in \widetilde{R}_s} v_{st} p_{ijd'r't} \widetilde{x}_{ijdr}. \tag{18}$$

The sparse cut $\widetilde{a}^\top x \leq \theta$ is then given by

$$\widetilde{a}_{ijdr} := \begin{cases} \sum_{s\in S, t\in T} v_{st} p_{ijd'r't}, & \text{if } \widetilde{x}_{ijdr} = 1 \\ 0, & \text{otherwise.} \end{cases} \tag{19}$$

By construction, the cut has the following property.

Lemma 4.2. *Let $\widetilde{x} \in X$ be a feasible timetable and $\widetilde{a}^\top x \leq \theta$ the sparse cut given by* (19)*. Then $\widetilde{a}^\top \widetilde{x}$ is equal to the optimal objective value of Subproblem* (14)*.*

In other words, the sparse cut is valid for the timetable $\widetilde{x}$ it is derived from. However, Example 4.3 shows that the sparse cut may overestimate the total expected energy consumption of other solutions $x \in X$.

Example 4.3. *Consider a system with three trains and one leg per train, i.e. $J = \{(1,1),(2,1),(3,1)\}$, and departure configurations $C_{11} = \{(0,2)\}, C_{21} = \{(2,2)\}, C_{31} = \{(0,2),(1,2)\}$. The energy consumption for $(i,j) \in J$ and $(d,r) \in C_{ij}$ is given by*

$$p_{ijdrt} = \begin{cases} 1, & \text{if } t = d \\ -1, & \text{if } t = d+1 \\ 0, & \text{otherwise.} \end{cases}$$

Consider the timetable represented by $\widetilde{x}_{1102} = \widetilde{x}_{2122} = \widetilde{x}_{3102} = 1$. Its energy consumption per time step t can be stated as

	$t=0$	$t=1$	$t=2$	$t=3$
p_{1102t}	*1*	*-1*	*0*	*0*
p_{2122t}	*0*	*0*	*1*	*-1*
p_{3102t}	*1*	*-1*	*0*	*0*
$\sum$	*2*	*-2*	*1*	*-1,*

resulting in a total energy consumption $\max\{0,2\} + \max\{0,-2\} + \max\{0,1\} + \max\{0,-1\} = 3$*. The corresponding sparse cut, defined by* (19)*, is*

$$x_{1102} + x_{2122} + x_{3102} \leq \theta.$$

The energy consumption per time step t of the timetable represented by $x_{1102} = x_{2122} = x_{3112} = 1$ can be stated as

	$t=0$	$t=1$	$t=2$	$t=3$
p_{1102t}	*1*	*-1*	*0*	*0*
p_{2122t}	*0*	*0*	*1*	*-1*
p_{3112t}	*0*	*1*	*-1*	*0*
$\sum$	*1*	*0*	*0*	*-1,*

resulting in a total energy consumption of 1*. The cut derived from $\widetilde{x}$, however, states* $2 \leq \theta$*.*

Next, we present a way to alleviate or even completely negate the overestimation.

4.3 Sparse Cut with Additional Slack

Let $\widetilde{x}$ be a solution to the master problem and $\lambda > 0$. Consider the adjusted cut

$$\widehat{a}^\top x - \lambda|J| \leq \theta, \tag{20}$$

where

$$\widehat{a}_{ijdr} \coloneqq \begin{cases} \widetilde{a}_{ijdr} + \lambda, & \text{if } \widetilde{x}_{ijdr} = 1 \\ 0, & \text{otherwise.} \end{cases}$$

A first observation is that for $\widetilde{x}$ this cut is equal to the original one given by (19).

Lemma 4.4. *Let $\widetilde{x} \in X$ be a feasible timetable and $\widehat{a}^\top x \leq \theta$ the sparse cut given by* (20)*. Then $\widehat{a}^\top \widetilde{x}$ is equal to the optimal objective value of Subproblem* (14)*.*

Proof. By construction of $\widehat{a}$ we have

$$\begin{aligned} \widehat{a}^\top \widetilde{x} - \lambda|J| &= \sum_{(i,j)\in J,(d,r)\in C_{ij}\colon \widetilde{x}_{ijdr}=1} \widehat{a}_{ijdr} - \lambda|J| \\ &= \sum_{(i,j)\in J,(d,r)\in C_{ij}\colon \widetilde{x}_{ijdr}=1} (\widetilde{a}_{ijdr} + \lambda) - \lambda|J| \\ &= \sum_{(i,j)\in J,(d,r)\in C_{ij}\colon \widetilde{x}_{ijdr}=1} \widetilde{a}_{ijdr} \\ &= \widetilde{a}^\top \widetilde{x}. \end{aligned}$$

The result then follows directly from Lemma 4.2. □

On the other hand, if $x \in X$ represents some timetable other than $\widetilde{x}$, then the left-hand side reduces by λ times the number of legs with differing departure configurations in $\widetilde{x}$ and x. More precisely, we get

$$\widehat{a}^\top x = \widetilde{a}^\top x - \lambda \sum_{(i,j)\in J,(d,r)\in C_{ij}} \max\{0, \widetilde{x}_{ijdr} - x_{ijdr}\} \leq \theta.$$

In Example 4.3, this adjusted cut plays out as follows.

Example 4.5. *Consider again the situation from Example 4.3. If we use the adjusted cut* (20) *with $\lambda = 1$, then we get*

$$2x_{1102} + 2x_{2122} + 2x_{3102} - 3 \leq \theta.$$

For the timetable represented by the solution $x_{1102} = x_{2122} = x_{3112} = 1$, this cut no longer overestimates the energy consumption since we have $2x_{1102} + 2x_{2122} + 2x_{3102} - 3 = 1$.

We would like to remark that a large enough value for λ yields an exact method in the spirit of Benders decomposition. For instance, if we assume no recuperation and for each leg in each scenario we use the departure configuration with the highest total energy consumption, then we get the following upper bound on the objective value of (14):

$$\overline{\lambda} \coloneqq \sum_{s\in S,(i,j)\in J} \max\Big\{ \sum_{t\in T} \max\{0, p_{ijdrt}\} : (d,r) \in C_{sij} \Big\}.$$

Let $\widetilde{x} \in X$ and consider the corresponding cut $\widehat{a}^\top x - \overline{\lambda}|J| \leq \theta$. Then for all $x \in X, x \neq \widetilde{x}$ we have

$$\widehat{a}^\top x - \overline{\lambda}|J| \leq 0,$$

which implies the cut no longer overestimates the energy consumption for any $x \in X, x \neq \widetilde{x}$. At the same time, however, this cut is extremely local in the sense that it is redundant for all $x \in X, x \neq \widetilde{x}$.

In our experiments in the next section, we will use three variants of this relaxation. The first one is simply (20), where we use a fixed value for λ in all iterations. The second variant changes each coefficient by a relative amount, i.e. we use

$$\widehat{a}_{ijdr} := \begin{cases} (1+\lambda)\widetilde{a}_{ijdr}, & \text{if } \widetilde{x}_{ijdr} = 1, \\ 0, & \text{otherwise,} \end{cases} \tag{21}$$

and the corresponding cut becomes

$$\widehat{a}^\top x - \lambda \sum_{(i,j) \in C_{ij} \colon \widetilde{x}_{ijdr}=1} \widetilde{a}_{ijdr} \leq \theta. \tag{22}$$

The third variant is a hybrid of the first two, using (22) with

$$\widehat{a}_{ijdr} := \begin{cases} \widetilde{a}_{ijdr} + \frac{\lambda}{|J|} \sum_{(i,j) \in C_{ij} \colon \widetilde{x}_{ijdr}=1} \widetilde{a}_{ijdr}, & \text{if } \widetilde{x}_{ijdr} = 1, \\ 0, & \text{otherwise.} \end{cases} \tag{23}$$

Consequently, all coefficients of the same cut are adjusted by the same absolute amount, but cuts with a higher energy consumption receive a stronger adjustment.

Finally, the sparse cut (and its variants) can also be parameterized to yield a more aggressive heuristic by choosing $\lambda < 0$. In this case, the cut yields stronger lower bounds on the objective variable θ, resulting in a lower number of iterations at the cost of a higher probability of overestimating the energy consumption of alternate timetables. Interestingly, the computational experiments in Section 5 show that this is indeed an effective approach.

Before presenting three additional improvements to the sparse-cut-based solution approach, we would like to offer some insight on alternate energy-related objective functions.

Remark 4.6. *In this work, we chose to present our results using the total energy consumption as objective since this is the largest cost factor for our industry partner. However, depending on the train operating company's goals, minimizing, e.g. , the peak load or the peak average load, may be alternate relevant objectives. For instance, Bärmann, Martin et al., 2021 consider different variants of the peak load average. These alternate objective functions only require linear constraints with continuous variables and therefore the Benders decomposition approach to handle uncertainties can also be applied with these objective functions. Moreover, the idea to 'only sum up the energy consumption of timestamps with positive total consumption' is also applicable to these objectives, but a detailed analysis would exceed the scope of this paper and is therefore left as an interesting prospect for future research. Finally, if the full pricing scheme is known it is also possible to formulate both objectives in terms of a single-objective framework.*

4.4 Further Improvements to the Solution Approach

We consider three further improvements to the sparse-cut heuristic. First, it is possible to guide the solving process by adding a partial or approximate objective function in the master problem instead of relying entirely on the cuts derived from the subproblem. Second, we will see that one of the possible objective functions for the master problem can in fact be used as a global underestimator for the energy consumption of all timetables in all scenarios. Third, we evaluate restarting the solving process after a certain number of cuts are derived from the subproblems.

Master Problem with Objective: We consider three candidates as objective functions for the master problem. Namely, the original objective function from Model 1 ('ActualRecup') and

two other objective functions that are strictly linear in the x variables representing the nominal timetable.

In the most simple case, we do not remove the original objective function consisting of (1a) and (1b) from Model (1) and adjust it to

$$\min \sum_{t \in T} z_t + \theta. \tag{24}$$

This can be seen as explicitly adding the scenario where all dwell and running times are executed as planned to the master problem. However, this approach results in longer solving times for the master problem, due to minimizing a piecewise-linear convex objective over the polytope representing X.

The intuition behind the next two objectives is as follows. Assuming a recuperation rate of 100 %, which we call the (idealized) *net energy consumption* (*'NetCon'*, leads to a high-quality approximation of the true energy consumption if the recuperation rate is high. Conversely, assuming no recuperation at all, which we call the *gross energy consumption* (*'GrossCon'*), leads to a high-quality approximation of the true energy consumption if the recuperation rate is low. We would like to note that either variant also works well if the net (or gross) energy consumption correlates with the actual energy consumption. To illustrate the difference between the three variants we reuse from Example 4.3 the timetable represented by $x_{1102} = x_{2122} = x_{3112} = 1$:

	$t = 0$	$t = 1$	$t = 2$	$t = 3$
p_{1102t}	1	-1	0	0
p_{2122t}	0	0	1	-1
p_{3112t}	0	1	-1	0
'ActualRecup'	1	0	0	0
'GrossCon'	1	1	1	0
'NetCon'	1	0	0	-1

In the case of net energy consumption we set

$$p_{ijdr} \coloneqq \sum_{t \in T} p_{ijdrt}, \quad (i,j) \in J, (d,r) \in C_{ij}, \tag{25}$$

and in the case of gross energy consumption we set

$$p_{ijdr} \coloneqq \sum_{t \in T} \max\{0, p_{ijdrt}\}, \quad (i,j) \in J, (d,r) \in C_{ij}. \tag{26}$$

The objectives can then be stated as

$$\min \sum_{(i,j) \in J, (d,r) \in C_{ij}} p_{ijdr} x_{ijdr} + \theta. \tag{27}$$

The explicitly added objectives can be expected to improve the gaps found in the solving process, since θ is bounded from below only by zero as well as the rather local sparse cuts. In contrast, the explicitly added objectives yield much stronger dual bounds. For instance, in the case of gross energy consumption, we can state a simple yet valid lower bound by

$$\sum_{(i,j) \in J} \min_{(d,r) \in C_{ij}} p_{ijdr},$$

i.e. we choose for each leg the cheapest available departure configuration. Moreover, there is a hierarchy between the three objectives, since the energy consumption with full recuperation (expressed by (27) with (25)) is a lower bound on the energy consumption with actual recuperation (expressed by (24)), which in turn is a lower bound on the energy consumption without recuperation (expressed by (27) with (26)).

Net Energy Consumption Cut: The net energy consumption of a leg, which is independent of all other trains, underestimates the true energy consumption, since there will usually be at least some unused recuperation energy. If we apply this idea to the scenario-based approach, we can derive an inequality that states a nontrivial lower bound of the energy consumption for scenarios and timetables. More precisely, consider the cut

$$\sum_{(i,j)\in J,(d,r)\in C_{ij}} \underline{p}_{ijdr} x_{ijdr} \leq \theta, \tag{28}$$

where

$$\underline{p}_{ijdr} := \sum_{\substack{s\in S,t\in T \\ r'=\max\{\underline{r}_{ij},r+\rho_{sij}\}}} p_{ijdr't}. \tag{29}$$

Since we assume 100% recuperation rate in this cut, the impact of the scenario on departure times can be disregarded entirely and it suffices to sum up the energy consumption corresponding to the adjusted running time over all scenarios. This cut can also be used with the dwell time recovery strategy. For the full recovery strategy, which also affects running times, we use in each scenario the running time with the lowest net energy consumption, i.e. in this case we define

$$\underline{p}_{ijdr} := \sum_{s\in S} \min\Big\{ \sum_{t\in T} p_{ijdr't} : r' \in \{\underline{r}_{ij},\dots,\max\{\underline{r}_{ij}, r+\rho_{sij}\}\}\Big\}. \tag{30}$$

Restarts: As shown by e.g. Achterberg, 2007 it is sometimes useful to restart the solving process for difficult problems. In our approach we run the initial method until a threshold number of generated sparse cuts is hit. Then we restart the entire solving process and use the final master problem from the original solving process as master problem in the first iteration of the restarted solving process. The restart procedure is not repeated if the iteration threshold is hit a second time.

5 Computational experiments

After laying out our experimental setup in Section 5.1, we present results of the following three experiments. First, we benchmark our heuristic solution approach based on the sparse cuts from Section 4 against the integrated MIP model from Section 3 and against the standard Benders decomposition approach stated as Algorithm 1. Second, we evaluate the improvements presented in Sections 4.3 and 4.4. Finally, we compare the base model to the alternate recovery strategies (Section 3.2) in terms of punctuality and energy efficiency.

5.1 Experimental Setup

Data and Instances The underlying data sets for this computational study have been provided by our partner VAG. Our instances are based on the 2022 Sunday timetable of the Nürnberg underground system, consisting of the manually driven line U1 and the automated lines U2 and U3, as well as real-world power profiles for the energy consumption of the trains. For full details on the instance generation, we refer to the EETTlib by Bärmann et al., 2022, where instances have been created in the same manner for the 2020 Sunday timetable.

In addition to the timetable and energy data, we have been provided with the operational data on seven Sundays of January and February 2022, from which we extracted the seven scenarios used in our experiments. We optimize the energy consumption of the automated underground lines U2 and U3 based on five of the seven scenarios. All evaluations of energy consumption are based on the remaining two scenarios, which simulates the energy consumption in 'unknown' scenarios.

Table 2: Metadata on instances in the first test set. For each instance, we state the start and end of the time horizon it represents as well as the number of trains, legs and departure configurations in the instance. Since it is fixed, line U1 is excluded from these counts.

		Number of		
Instance	Time Horizon	Trains	Legs	DepConfs
15_5_15min	12–12:15	21	124	2 912
15_5_30min	12–12:30	30	252	5 901
15_5_01hr	12–13	48	504	11 802
15_5_02hr	12–14	84	1 008	23 604
15_5_04hr	12–16	156	2 016	47 208
15_5_06hr	12–18	228	3 024	70 812
15_5_08hr	12–20	300	4 032	94 416
15_5_10hr	11–21	369	5 017	117 488
15_5_12hr	10–22	419	5 717	134 085

Table 3: Model sizes after Gurobi presolve for Model 8 in the format 'no recovery'/'dwell recovery'/'full recovery' and the master problem (13). Numbers are abbreviated as 'k' for 'thousands' and 'M' for 'millions'.

	Number of Variables		Number of Constraints		Number of Nonzeros	
Instance	MIP	Master	MIP	Master	MIP	Master
15_5_15min	51k/42k/36k	1 819	154k/512k/394k	523	2.6M/3.6M/2.8M	11k
15_5_30min	116k/87k/73k	1 816	440k/1.2M/891k	522	6.3M/8.0M/6.0M	11k
15_5_01hr	244k/184k/147k	1 816	999k/2.7M/2.0M	522	13.8M/17.5M/13.1M	11k
15_5_02hr	504k/367k/292k	1 816	2.2M/5.4M/4.0M	522	29.0M/35.2M/26.3M	11k
15_5_04hr	1.0M/737k/582k	1 816	4.5M/11.0M/8.1M	522	59.3M/70.0M/51.8M	11k
15_5_06hr	1.5M/1.1M/874k	1 816	6.8M/16.5M/12.2M	522	89.4M/104.4M/77.4M	11k
15_5_08hr	2.1M/1.5M/1.2M	1 816	9.1M/22.0M/16.3M	522	119.6M/138.3M/102.7M	11k
15_5_10hr	2.6M/1.8M/1.5M	2 256	11.0M/26.3M/19.5M	656	148.0M/168.4M/125.4M	13k
15_5_12hr	3.0M/2.1M/1.7M	3 059	12.4M/30.5M/22.5M	884	169.5M/194.2M/144.4M	18k

Finally, the underground line U1 is driven manually and also runs on its own power network. We therefore do not optimize this line and also disregard its energy consumption, but keep the unaltered timetable of line U1 in the model to represent the connection constraints.

Test Sets Our computational experiments use two test sets. The first one consists of instances that vary in size and represent parts of the whole day. Given the size of the time frame, we choose its start such that the number of legs in the instances scales about linearly with the size of the time horizon. The resulting instances in the first test set and relevant metadata are summarized in Table 2. The departure time of each leg may be shifted by up to ± 15 s in 5 s increments and the running time of each leg can be chosen from a set of three or four running times; see Bärmann et al., 2022 for full details on the selection of available running times. As a result, for each leg we can choose from 21 or 28 combinations of departure and running time. The resulting sizes of Model (8) and the master problem (13), both after Gurobi's presolve, are stated in Table 3. The size of Model (8) scales about linearly with the number of legs in the instance (cf. Table 2). In the case of the master problem (13), which contains only the timetabling component of the problem, Gurobi is able to presolve most instances to the same size. This is possible due to the synchronization constraints (7) and the fact that the timetable is periodic between noon and 8 pm.

The second test set consists of instances that represent the entire operational day, which begins at about 4 am and ends 22 hours later at about 2 am. The timetable represented by these instances consists of 604 trains with a total of 8406 legs to be scheduled on lines U2 and U3. In these instances, we consider varying maximum departure time shifts and shift steps which are summarized in Table 4.

Table 4: Metadata on instances in the second test set. For each instance, we state the maximum departure shift, shift step and resulting number of departure configurations in the instance. Each instance consists of 604 trains with 8406 legs on lines U2 and U3.

Instance	Max. Departure Shift	Shift Step	Number of Dep. Confs.
10_2	10	2	310 662
15_3	15	3	310 662
15_5	15	5	197 694
20_4	20	4	310 662
20_5	20	5	254 178
25_5	25	5	310 662
30_5	30	5	367 146

Computational Setup The implementations of all models are done in Python 3.7.11 (Van Rossum and Drake, 2009) using the Python-API of Gurobi 9.5.1 (Gurobi Optimization, 2022).

The computations in Section 5.2 have been performed on a server with 2x Intel Xeon E5-2643 v4 CPUs ("Broadwell", 12 cores/24 threads, 3.4 GHz base frequency) and 256 GB RAM due to the large memory requirements of Model (8) and Algorithm 1. To save computational resources, the computations in Sections 5.3 and 5.4 have been performed on a server with Intel Xeon E3-1240 v6 CPUs ("Kaby Lake", 4 cores, HT disabled, 3.7 GHz base frequency) and 32 GB RAM. We stabilize the results against the impact of nondeterminism in the optimization solver by performing five runs per experiment. All computations use a Gurobi time limit of two hours and run on four threads. To add our cuts in Algorithm 1 and the sparse-cut heuristic, we additionally set GRB.LazyConstraints to 1 in these runs. As a basic improvement of these two approaches, each time our callback cuts off a solution, we pass an updated feasible solution to Gurobi. All approaches use the original timetable as starting solution. Otherwise, we use Gurobi's standard parameter settings throughout.

5.2 Comparison of the Solution Approaches

In our first test, we compare the performances of solving Model (8) (*MIP*), applying Algorithm 1 (*BND*), and the sparse-cut heuristic from Section 4.2 (*SCH*) on the first test set. Tables 5 and 6 state for each of the three recovery strategies *No Recovery* (Section 3.1), *Dwell Recovery* and *Full Recovery* (Section 3.2), Gurobi's final optimality gap and the energy savings potential, respectively. Each row represents the results on one instance, averaged over the aforementioned five runs and the best-performing approach per instance is marked in bold font. The savings potential is calculated as the relative difference between the expected energy consumption over the scenarios of the original timetable and the optimized timetable. In particular, this difference is not based on the objective value of SCH, as it may overestimate the solution's energy consumption. In the case of BND, the large subproblems could not be solved within the time limit, leading to failed runs on all but the smallest instances. The tables contain no numerical values in the corresponding entries.

We can conclude that the MIP approach is well-suited for very small instances, but does not scale with instance size since the model becomes too large. In fact, for larger instances it is not even able to improve the starting solution at all and the bounds do not improve beyond the root relaxation either. The Benders decomposition approach runs into the same problem of the subproblems becoming too large to be solved within the time limit. As a result, we observe no solution improvement for larger instances and the approach typically yields the trivial zero-bound as best bound. On the other hand, the final optimality gap and savings potential achieved with the sparse cut heuristic remains stable across the entire test set. We would also like to note that, on average, the dual bounds found by the sparse cut heuristic are slightly better than the ones found by the MIP approach. Based on these results we disregard

Table 5: Comparison of the final optimality gaps achieved with the three solution approaches.

	No Recovery			Dwell Recovery			Full Recovery		
Instance	MIP	BND	SCH	MIP	BND	SCH	MIP	BND	SCH
15_5_15min	**8.8**	60.0	57.4	**9.2**	100.0	56.4	**7.7**	100.0	54.7
15_5_30min	**12.3**	100.0	57.9	**12.3**	100.0	57.1	**11.3**	100.0	54.6
15_5_01hr	63.0	100.0	**58.1**	**13.9**	100.0	57.8	**13.4**	100.0	54.8
15_5_02hr	**24.3**	100.0	57.5	**52.3**	100.0	57.3	**29.5**	100.0	55.1
15_5_04hr	62.4	100.0	**57.5**	62.4	-	**57.6**	59.9	100.0	**54.9**
15_5_06hr	62.2	-	**57.5**	62.3	-	**57.5**	59.8	-	**55.0**
15_5_08hr	62.3	-	**57.4**	62.4	-	**57.4**	59.8	-	**54.8**
15_5_10hr	62.2	-	**57.1**	62.2	-	**57.1**	59.6	-	**54.5**
15_5_12hr	61.1	-	**56.0**	61.1	-	**56.2**	58.5	-	**53.5**

Table 6: Comparison of the energy saving achieved with the three solution approaches.

	No Recovery			Dwell Recovery			Full Recovery		
Instance	MIP	BND	SCH	MIP	BND	SCH	MIP	BND	SCH
15_5_15min	**7.9**	2.4	5.1	**8.0**	4.3	5.3	**9.8**	3.2	5.8
15_5_30min	3.1	3.8	**5.3**	2.8	3.5	**5.8**	4.0	4.6	**6.5**
15_5_01hr	0.0	3.3	**4.6**	0.0	2.8	**4.4**	0.2	3.2	**6.1**
15_5_02hr	0.0	1.5	**4.2**	0.3	1.7	**4.7**	0.0	2.4	**4.7**
15_5_04hr	0.0	3.2	**5.2**	0.0	-	**4.9**	0.0	3.2	**5.3**
15_5_06hr	0.0	-	**5.2**	0.0	-	**5.3**	0.0	-	**5.0**
15_5_08hr	0.0	-	**6.5**	0.0	-	**5.4**	0.0	-	**6.7**
15_5_10hr	0.0	-	**5.7**	0.0	-	**5.5**	0.0	-	**5.8**
15_5_12hr	0.0	-	**6.4**	0.0	-	**5.0**	0.0	-	**5.9**

solving Model (8) and Algorithm 1 during the remaining experiments.

5.3 Improvements to the Sparse-Cut Approach

Our second set of experiments focuses on the sparse-cut heuristic. We first evaluate the λ-parameterized sparse cuts from Section 4.3 and then follow up with an evaluation of the three improvements from Section 4.4.

Table 7 shows the results for using the unaltered sparse cut in the first row and different settings for λ in the sparse cut with a fixed absolute correction to each coefficient (*Abs*), a relative correction to each individual coefficient (*Rel* without *Balance*) and finally the hybrid correction (*Rel* with *Balance*). Each row represents the results averaged over five runs and the entire second test set.

We can observe empirically that, regardless of the cut variant, lower values for λ result in smaller MIP gaps. It is important to note, however, that the gap always relates to the current master problem. Decreasing values for λ lead to more aggressive sparse cuts in the master problem, i.e. , cuts with larger left-hand sides. As a result, these cuts yield stronger dual bounds in the master problem, but may overestimate the true energy consumption to a higher degree. Although the differences in energy savings potential are small to moderate, we observe empirically that the best results are achieved with an absolute $\lambda = -5$ for 'No Recovery', the unaltered heuristic cut for 'Dwell Recovery', and relative $\lambda = -0.2$ with balancing for 'Full Recovery'. We use the respective parameter settings as base runs in the next experiment.

Since adding an objective to the master problem and adding the net energy consumption cut strongly interact with each other, we run a performance evaluation on all combinations and summarize the results in Table 8. First of all, the gaps suggest that the explicitly added objectives help Gurobi to find stronger bounds. We recall from Section 4.4 that 'NetCon' is a lower bound on 'ActualRecup', which itself is a lower bound on 'GrossCon'. Moreover, the two objectives 'NetCon' and 'GrossCon' can easily be bounded by, e.g. , choosing the most energy efficient departure configuration for each individual leg. If we compare 'NetCon' to 'GrossCon',

Table 7: Performance comparison of heuristic approach with different parameter settings for the adjusted sparse cuts. The base run is marked grey and the best result in each column is highlighted in bold font.

Parameters			No Recovery		Dwell Recovery		Full Recovery	
Abs	Rel	Balance	Gap (%)	Saving (%)	Gap (%)	Saving (%)	Gap (%)	Saving (%)
0	0.0	False	54.3	6.8	**54.6**	**5.9**	52.1	5.7
10	0.0	False	59.9	6.7	60.2	5.4	57.4	5.6
5	0.0	False	57.1	6.7	57.4	5.3	54.7	5.8
-5	0.0	False	54.4	**7.0**	54.7	5.3	52.1	5.6
-10	0.0	False	54.3	7.0	54.6	5.5	**52.0**	6.0
0	0.2	False	66.0	6.8	66.4	5.3	63.3	5.7
0	0.1	False	60.2	6.8	60.5	5.5	57.6	5.8
0	-0.1	False	54.4	6.9	54.6	5.5	52.0	6.0
0	-0.2	False	54.4	6.8	54.7	5.5	52.0	5.8
0	0.2	True	65.7	6.7	66.1	5.3	63.5	5.6
0	0.1	True	60.1	6.8	60.3	5.3	57.8	5.7
0	-0.1	True	54.3	6.6	54.6	5.5	52.1	5.8
0	-0.2	True	**54.3**	6.8	54.6	5.8	52.0	**6.0**

Table 8: Performance of adding an objective to the master problem and using the net consumption cut. The base run is marked grey and the best result in each column is highlighted in bold font.

Parameters		No Recovery		Dwell Recovery		Full Recovery	
MA Obj.	Netcon Cut	Gap (%)	Saving (%)	Gap (%)	Saving (%)	Gap (%)	Saving (%)
None	False	54.4	7.0	54.6	5.9	52.0	6.0
ActualRecup	False	47.8	4.2	47.5	5.0	45.9	4.9
GrossCon	False	39.7	8.0	40.0	6.5	38.6	6.7
NetCon	False	46.2	8.1	46.5	6.6	44.8	6.8
None	True	23.4	9.0	23.9	7.7	30.1	7.0
ActualRecup	True	21.3	5.6	20.8	6.3	26.0	6.1
GrossCon	True	**17.6**	8.8	**18.0**	7.2	**22.9**	7.0
NetCon	True	20.0	**9.3**	20.5	**7.8**	25.8	**7.8**

then we can observe that smaller values of the (easily bounded) master problem objectives indeed correspond to larger gaps. Although 'Actual' does not cleanly fit this pattern and the solution quality deteriorates for this objective, this is most likely a consequence of this objective being piecewise linear and convex instead of simply linear, which in turn results in much longer solution times for the master problem. For instance, in the case of 'No Recovery', the solving process generates, on average, only 11 sparse cuts and Gurobi explores only a single node with the actual recuperation objective in the master problem, whereas we get 1015 and 1033 sparse cuts as well as 80678 and 85529 nodes for 'NetCon' and 'GrossCon', respectively. Finally, for all three recovery strategies the net energy consumption cut has a larger positive impact than explicitly adding any of the other objectives to the master problem. However, doing both yields the best results for all three recovery models. Again, we use the best configuration—(NetCon, True)—as baseline in the next experiment.

The final performance experiment evaluates the impact of a restart mechanism. Table 9 shows the results, where we can observe a slight improvement when restarts are being used.

5.4 Comparison of Recovery Strategies

To evaluate the cost and benefit of the recovery strategies, we compare the best results for each recovery strategy consisting of the five runs on the seven instances in Table 10. On the cost side, we use the energy consumption for 'No Recovery' as baseline and state the additional energy cost for the other two recovery strategies. On the benefit side, we use two metrics to give an

Table 9: Performance of the restart mechanism. The base run is marked grey and the best result in each column is highlighted in bold font.

	No Recovery		Dwell Recovery		Full Recovery	
Restart	Gap (%)	Saving (%)	Gap (%)	Saving (%)	Gap (%)	Saving (%)
-1	20.0	9.3	20.5	7.8	**25.8**	7.8
500	**20.0**	**9.5**	20.4	8.0	25.9	7.9
1000	20.0	9.5	**20.4**	**8.0**	25.8	8.0
1500	20.1	9.3	20.4	7.9	25.9	**8.1**

Table 10: Cost and benefit of recovery strategies

		Avg. #Deps. with Delay			Avg. #Deps. by Minute	
Recovery	Additional Energy Cost (%)	≥ 10 s	≥ 20 s	≥ 30 s	Punctual	Delayed
None	-	4464.7	2327.0	882.9	6444.7	1961.3
Dwell	1.1	2112.8	521.5	233.8	7457.1	948.9
Full	1.6	1981.8	416.0	197.5	7491.0	915.0

overview of the change in punctuality. For *Avg. #Deps. with Delay* we simply count the number of legs that are delayed by a certain amount, averaged over the five runs and seven instances. For *Avg. #Deps. by Minute*, suggested by our industry partner, we count a departure as delayed if it departs at a later minute than originally planned. For example, if a specific leg is planned to depart at 11:55:43, it will be counted as delayed if it departs at least 17 s late. The idea behind this measurement is, first, that from the passenger's perspective, departures are scheduled by the minute, and second, connections to other means of public transport, as e.g. rail or bus, are also scheduled by the minute. In short, we can observe that using spare dwell times to recover from delays is a comparatively low-cost strategy to significantly reduce delays. In contrast, using spare dwell and running times to recover from delays yields very little additional benefit at an additional 0.5 % increase in energy consumption.

To summarize, the first experiment shows that the integrated MIP approach and the standard Benders decomposition approach are not practically applicable to whole-day instances due to the size of the models. However, the MIP approach remains relevant for small-scale instances, as they could arise in moving-horizon approaches (see e.g. Glomb et al., 2022) or optimization of periodic timetables. Second, on whole-day instances we can slightly improve the results by increasing the aggressiveness of the sparse cuts (see Section 4.3, $\lambda < 0$) and substantially improve solution quality by adding a linear objective to the master problem as well as adding the net energy consumption cut. In our tests, the effect of a restart mechanism can be considered slightly positive. Finally, our comparison between the recovery strategies shows that spare dwell times pose a cheap and efficient way to improve punctuality, while shortening running times provides much less improvement at a higher cost.

Before moving on to the conclusions, we would like to close the loop to our first example from Figure 1, which represents the unaltered real-world instance relevant to our industry partner VAG. Table 11 states the total energy consumption of the original timetable, the timetable optimized without considering uncertainty, and the timetable optimized under uncertainty with our new approach. The first interesting observation is that the deviations lead to a higher energy consumption (*Actual Timetable*) in comparison to all departure and running times being executed as planned (*Planned Timetable*). The optimization without uncertainty significantly underestimates the actual energy consumption, while the difference between planned and actual energy consumption is relatively small after optimization under uncertainty. This already gives our industry partner a much better view on their energy consumption. Second, we can observe that optimizing without uncertainty results in a 5.1 % lower total energy consumption in the actual timetable. Third, we see a 9.4 % reduction after optimization under uncertainty, which

Table 11: Energy savings potential for real-world instance

	Total energy consumption (MWh)	
Timetable	Planned Timetable	Actual Timetable
Original by VAG	45.1	49.0
Optimization without uncertainty	37.7	46.5
Optimization with uncertainty	44.7	44.4

shows that optimization under uncertainty indeed holds significant potential. We expect this potential to increase even further, once enough scenario data becomes available such that the synchronization constraints (7) can be relaxed or dropped entirely. The resulting additional difficulty of the model poses an interesting topic for further research.

6 Conclusions and Outlook

The main result of this work are a new model and solution approach to improve an existing timetable draft in terms of energy efficiency under uncertainty. The MIP model can, in principle, be solved directly or via a Benders decomposition approach. However, both approaches rely on solving large optimization models, and we therefore devise a sparse-cut heuristic to efficiently compute high-quality solutions. We further improve the heuristic by first introducing a variant where aggressiveness of the cut can be controlled using a parameter λ and then evaluating three smaller improvements: an objective in the master problem, an auxiliary cut, and restarts. Our computational experiments validate the effectiveness of the heuristic approach. Further, our results show that shortening dwell times is an effective approach to improve punctuality while shortening dwell and running times yields little additional benefit. There are two obviously interesting topics for future research. First, once additional data becomes available, it would be interesting to see how the results improve if the synchronization constraints are relaxed or even dropped from the model. Second, existing and new real-time optimization algorithms for delay recovery could be combined with the methods of this work to compute advanced recoverable energy-efficient timetables under uncertainty. One approach could be to simulate a nominal timetable using a real-time recovery algorithm and then deriving the sparse cut as in our approach. If the simulation turns out to be a bottleneck in terms of computation time, one could reduce computation times by simplifying or approximating the recovery behaviour. The proper simulation could then be used only as a benchmark or a hybrid approach could be produced by occasionally running the exact simulation.

References

Achterberg, Tobias (2007). 'Constraint integer programming'. Doctoral Thesis. Berlin: Technische Universität Berlin, Fakultät II - Mathematik und Naturwissenschaften. DOI: 10.14279/depositonce-1634.

Albrecht, Thomas (2010). 'Reducing power peaks and energy consumption in rail transit systems by simultaneous train running time control'. In: *WIT Transactions on State of the Art in Science and Engineering* 39.

Bärmann, Andreas, Thorsten Gellermann, Maximilian Merkert and Oskar Schneider (2018). 'Staircase compatibility and its applications in scheduling and piecewise linearization'. In: *Discrete Optim.* 29, pp. 111–132. DOI: 10.1016/j.disopt.2018.04.001.

Bärmann, Andreas, Patrick Gemander, Lukas Hager, Frederik Nöth and Oskar Schneider (2022). *EETTlib – Energy-efficient train timetabling library*. Under revision at *Networks*.

Bärmann, Andreas, Patrick Gemander and Maximilian Merkert (2020). 'The clique problem with multiple-choice constraints under a cycle-free dependency graph'. In: *Discrete Appl. Math.* 283, pp. 59–77. DOI: 10.1016/j.dam.2019.12.015.

Bärmann, Andreas, Alexander Martin and Oskar Schneider (2017). 'A comparison of performance metrics for balancing the power consumption of trains in a railway network by slight timetable adaptation'. In: *Public Transport* 9.1-2, pp. 95–113. DOI: 10.1007/s12469-017-0160-4.

— (2021). 'Efficient formulations and decomposition approaches for power peak reduction in railway traffic via timetabling'. In: *Transportation Sci.* 55.3, pp. 747–767. DOI: 10.1287/trsc.2020.1021.

Cacchiani, Valentina, Dennis Huisman, Martin Kidd, Leo Kroon, Paolo Toth, Lucas Veelenturf and Joris Wagenaar (2014). 'An overview of recovery models and algorithms for real-time railway rescheduling'. In: *Transportation Res. Part B* 63, pp. 15–37. DOI: 10.1016/j.trb.2014.01.009.

Cacchiani, Valentina and Paolo Toth (2012). 'Nominal and robust train timetabling problems'. In: *European J. Oper. Res.* 219.3, pp. 727–737. DOI: 10.1016/j.ejor.2011.11.003.

Chen, Jiann-Fuh, Ray-Lee Lin and Yow-Chyi Liu (2005). 'Optimization of an MRT train schedule: reducing maximum traction power by using genetic algorithms'. In: *IEEE Transactions on power systems* 20.3, pp. 1366–1372. DOI: 10.1109/TPWRS.2005.851939.

Feng, Xuesong, Hanxiao Zhang, Yong Ding, Zhili Liu, Hongqin Peng and Bin Xu (2013). 'A review study on traction energy saving of rail transport'. In: *Discrete Dynamics in Nature and Society* 2013. DOI: 10.1155/2013/156548.

Fournier, David, Denis Mulard and François Fages (2012). 'Energy optimization of metro timetables: A hybrid approach'. In: *The 18th International Conference on Principles and Practice of Constraint Programming*. Springer, pp. 7–12.

Glomb, Lukas, Frauke Liers and Florian Rösel (2022). 'A rolling-horizon approach for multi-period optimization'. In: *European J. Oper. Res.* 300.1, pp. 189–206. DOI: 10.1016/j.ejor.2021.07.043.

Gong, Cheng, Shiwen Zhang, Feng Zhang, Jianguo Jiang and Xinheng Wang (2014). 'An integrated energy-efficient operation methodology for metro systems based on a real case of Shanghai metro line one'. In: *Energies* 7.11, pp. 7305–7329. DOI: 10.3390/en7117305.

Gurobi Optimization, LLC (2022). *Gurobi Optimizer Reference Manual*. URL: http://www.gurobi.com.

Hasegawa, Daisuke, Gemma L. Nicholson, Clive Roberts and Felix Schmid (2014). 'The impact of different maximum speeds on journey times, energy use, headway times and the number of trains required for phase one of Britain's high speed two line'. In: *WIT Transactions on the Built Environment* 135, pp. 485–496. DOI: 10.2495/CR140401.

Kall, Peter and Stein W. Wallace (1994). *Stochastic Programming*. Wiley-Interscience Series in Systems and Optimization. John Wiley & Sons, Ltd., Chichester, pp. xii+307. DOI: 10.1007/978-3-642-88272-2.

Kim, Kyung-Min, Kwang-Tae Kim and Moonseob Han (2011). 'A model and approaches for synchronized energy saving in timetable'. In: *Proceedings of the 9th World Congress Railway Research (WCRR)*. Lille, France, May 2011, pp. 1–8.

Kim, Kyung-Min, Suk-Mun Oh and Moonseob Han (2010). 'A mathematical approach for reducing the maximum traction energy: the case of Korean MRT trains'. In: *Proceedings of the International MultiConference of Engineers and Computer Scientists*. Hong Kong, 17-19 March 2010, vol. III (International Association of Engineers), pp. 2169–2173.

Kimura, N. and Masafumi Miyatake (2014). 'Strategy of speed restriction allowing extended running times to minimize energy consumption and passenger disutility'. In: *WIT Transactions on The Built Environment* 135, pp. 733–743. DOI: 10.2495/CR140611.

Kleywegt, Anton J., Alexander Shapiro and Tito Homem-de-Mello (2002). 'The sample average approximation method for stochastic discrete optimization'. In: *SIAM J. Optim.* 12.2, pp. 479–502. DOI: 10.1137/S1052623499363220.

Kliewer, Natalia and Leena Suhl (2011). 'A note on the online nature of the railway delay management problem'. In: *Networks* 57.1, pp. 28–37. DOI: 10.1002/net.20381.

Kroon, Leo, Gábor Maróti, Mathijn R. Helmrich, Michiel Vromans and Rommert Dekker (2008). 'Stochastic improvement of cyclic railway timetables'. In: *Transportation Res. Part B* 42.6, pp. 553–570. DOI: 10.1016/j.trb.2007.11.002.

Li, Xiang and Hong K. Lo (2014). 'Energy minimization in dynamic train scheduling and control for metro rail operations'. In: *Transportation Res. Part B* 70, pp. 269–284. DOI: 10.1016/j.trb.2014.09.009.

Lusby, Richard M., Jesper Larsen and Simon Bull (2018). 'A survey on robustness in railway planning'. In: *European J. Oper. Res.* 266.1, pp. 1–15. DOI: 10.1016/j.ejor.2017.07.044.

Peña-Alcaraz, Maite, Antonio Fernández, Asuncion P. Cucala, Andres Ramos and Ramon R. Pecharromán (2012). 'Optimal underground timetable design based on power flow for maximizing the use of regenerative-braking energy'. In: *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit* 226.4, pp. 397–408. DOI: 10.1177/0954409711429411.

Raghunathan, Arvind U., Toshihiro Wada, Kenji Ueda and Satoru Takahashi (2014). 'Minimizing energy consumption in railways by voltage control on substations'. In: *WIT Transactions on The Built Environment* 135, pp. 697–708. DOI: 10.2495/CR140581.

Restel, Franciszek, Łukasz Wolniewicz and Matea Mikulčić (2021). 'Method for designing robust and energy efficient railway schedules'. In: *Energies* 14.24, p. 8248. DOI: 10.3390/en14248248.

Sansó, Brunilde and Pierre Girard (1997). 'Instantaneous power peak reduction and train scheduling desynchronization in subway systems'. In: *Transportation Sci.* 31.4, pp. 312–323. DOI: 10.1287/trsc.31.4.312.

Scheepmaker, Gerben M., Rob M. P. Goverde and Leo G. Kroon (2017). 'Review of energy-efficient train control and timetabling'. In: *European J. Oper. Res.* 257.2, pp. 355–376. DOI: 10.1016/j.ejor.2016.09.044.

Su, Shuai, Xiang Li, Tao Tang and Ziyou Gao (2013). 'A subway train timetable optimization approach based on energy-efficient operation strategy'. In: *IEEE Transactions on Intelligent Transportation Systems* 14.2, pp. 883–893. DOI: 10.1109/TITS.2013.2244885.

Van Rossum, Guido and Fred L. Drake (2009). *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace.

Wang, Pengling and Rob M. P. Goverde (2019). 'Multi-train trajectory optimization for energy-efficient timetabling'. In: *European J. Oper. Res.* 272.2, pp. 621–635. DOI: 10.1016/j.ejor.2018.06.034.

Wang, Yihui, Songwei Zhu, Andrea D'Ariano, Jiateng Yin, Jianrui Miao and Lingyun Meng (2021). 'Energy-efficient timetabling and rolling stock circulation planning based on automatic train operation levels for metro lines'. In: *Transportation Research Part C: Emerging Technologies* 129, p. 103209. DOI: 10.1016/j.trc.2021.103209.

Yin, Jiateng, Tao Tang, Lixing Yang, Ziyou Gao and Bin Ran (2016). 'Energy-efficient metro train rescheduling with uncertain time-variant passenger demands: An approximate dynamic programming approach'. In: *Transportation Res. Part B* 91, pp. 178–210. DOI: 10.1016/j.trb.2016.05.009.

Yin, Jiateng, Tao Tang, Lixing Yang, Jing Xun, Yeran Huang and Ziyou Gao (2017). 'Research and development of automatic train operation for railway transportation systems: A survey'. In: *Transportation Research Part C: Emerging Technologies* 85, pp. 548–572. DOI: 10.1016/j.trc.2017.09.009.

Zhou, Leishan, Lu (Carol) Tong, Junhua Chen, Jinjin Tang and Xuesong Zhou (2017). 'Joint optimization of high-speed train timetables and speed profiles: A unified modeling approach using space-time-speed grid networks'. In: *Transportation Res. Part B* 97, pp. 157–181. DOI: 10.1016/j.trb.2017.01.002.

A Graph Model for Feasible Timetable Adjustments

We first explain the construction of the *compatibility graph* representing the feasible timetable adjustments as previously stated by Bärmann et al., 2022, Appendix B. The timetabling component of the problem can then be solved as a *clique problem with multiple-choice constraints* CPMC for which we summarize the *stable set formulation*, previously stated by Bärmann, Gemander et al., 2020, Section 3.

We reuse and extend the notation from Section 2. Let L^{HW} be the set containing all pairs of legs consecutively traversing the same track in the same direction. For each $(i_1, j_1, i_2, j_2) \in L^{HW}$ let t_{i_1,j_1,i_2,j_2} be the minimal headway time between the first and second leg. Analogously, let L^{SHW}, L^{D}, L^{TT} and L^{C} be the sets containing all pairs of legs corresponding to the single-track headway time constraints, dwell time constraints, terminal turnaround constraints and connection constraints, respectively. Further, denote by t_{i_1,j_1,i_2,j_2} the minimal dwell or terminal turnaround time between the legs (i_1, j_1) and (i_2, j_2). In the case of connection constraints, we denote by $t^{\min}_{i_1,j_1,i_2,j_2}$ the minimal connection time and by $t^{\max}_{i_1,j_1,i_2,j_2}$ the maximal connection time.

The vertex set of the compatibility graph G, representing feasible timetable adjustments, consists of one vertex for each leg and departure configuration, which is can be stated as

$$V(G) = \{(i, j, d, r) : (i, j) \in J, (d, r) \in C_{ij}\}.$$

We choose for each leg $(i, j) \in J$ exactly one departure configuration. Hence, we partition the set $V(G)$ into

$$\mathcal{V} \coloneqq \big\{\{(i, j, d, r) : (d, r) \in C_{ij}\} : (i, j) \in J\big\}.$$

The task of CPMC is to choose one vertex $v \in V_i$ for each $V_i \in \mathcal{V}$ such that the chosen vertices are compatible to each other, i.e. they form a clique in G. It remains to define the edge set $E(G)$, which is the union of the following subsets:

$$\begin{aligned}
E^{HW} \coloneqq \big\{\{(i_1, j_1, d_1, r_1), (i_2, j_2, d_2, r_2)\} : {} & (i_1, j_1, i_2, j_2) \in L^{HW}, (d_1, r_1) \in C_{i_1,j_1}, (d_2, r_2) \in C_{i_2,j_2} \\
& \text{with } d_1 + \min\{r_1 - r_2, 0\} + t_{i_1,j_1,i_2,j_2} \le d_2\big\}, \\
E^{SHW} \coloneqq \big\{\{(i_1, j_1, d_1, r_1), (i_2, j_2, d_2, r_2)\} : {} & (i_1, j_1, i_2, j_2) \in L^{SHW}, (d_1, r_1) \in C_{i_1,j_1}, (d_2, r_2) \in C_{i_2,j_2} \\
& \text{with } d_1 + r_1 \le d_2\big\}, \\
E^{D} \coloneqq \big\{\{(i_1, j_1, d_1, r_1), (i_2, j_2, d_2, r_2)\} : {} & (i_1, j_1, i_2, j_2) \in L^{D}, (d_1, r_1) \in C_{i_1,j_1}, (d_2, r_2) \in C_{i_2,j_2} \\
& \text{with } d_1 + r_1 + t_{i_1,j_1,i_2,j_2} \le j_2\big\}, \\
E^{TT} \coloneqq \big\{\{(i_1, j_1, d_1, r_1), (i_2, j_2, d_2, r_2)\} : {} & (i_1, j_1, i_2, j_2) \in L^{TT}, (d_1, r_1) \in C_{i_1,j_1}, (d_2, r_2) \in C_{i_2,j_2} \\
& \text{with } d_1 + r_1 + t_{i_1,j_1,i_2,j_2} \le j_2\big\}, \\
E^{C} \coloneqq \big\{\{(i_1, j_1, d_1, r_1), (i_2, j_2, d_2, r_2)\} : {} & (i_1, j_1, i_2, j_2) \in L^{C}, (d_1, r_1) \in C_{i_1,j_1}, (d_2, r_2) \in C_{i_2,j_2} \\
& \text{with } t^{\min}_{i_1,j_1,i_2,j_2} \le d_2 - d_1 + r_1 \le t^{\max}_{i_1,j_1,i_2,j_2}\big\}, \\
E \coloneqq \big\{\{(i_1, j_1, d_1, r_1), (i_2, j_2, d_2, r_2)\} : {} & (i_1, j_1), (i_2, j_2) \in J, (d_1, r_1) \in C_{i_1,j_1}, (d_2, r_2) \in C_{i_2,j_2} \\
& \text{with } (i_1, j_1) \neq (i_2, j_2) \\
& \text{and } (i_1, j_1, i_2, j_2) \notin L^{HW} \cup L^{SHW} \cup L^{TT} \cup L^{C}\big\}.
\end{aligned}$$

We would like to note that for ease of presentation we omit the division of E^C into two subtypes of constraints as it can be seen as a minor implementation detail and we refer the interested reader to Bärmann et al., 2022.

Given a (compatibility) graph G with partition $\mathcal{V}$ of its vertex set $V(G)$, the corresponding stable set formulation of CPMC, as stated by Bärmann, Gemander et al., 2020, is given by

$$\begin{aligned} &\text{find } x \\ &\text{s.t.} \sum_{v \in V_i} x_v = 1, \quad V_i \in \mathcal{V}, \\ &\quad \sum_{v \in S} x_v \leq 1, \quad S \text{ is a stable set in } G, \\ &\quad x_v \in \{0,1\}, \quad v \in V(G). \end{aligned}$$

Acknowledgements

This research was supported 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" (20-3410-2-9-8). Moreover, we would like to thank Johannes Thürauf for reading a former version of this manuscript and providing comments that helped improve the paper.