Determining All Integer Vertices of the PESP Polytope by Flipping Arcs

Niels Lindner
Zuse Institute Berlin, Germany
lindner@zib.de

Christian Liebchen
Technical University of Applied Sciences Wildau, Germany
liebchen@th-wildau.de

Abstract

We investigate polyhedral aspects of the Periodic Event Scheduling Problem (PESP), the mathematical basis for periodic timetabling problems in public transport. Flipping the orientation of arcs, we obtain a new class of valid inequalities, the *flip inequalities*, comprising both the known cycle and change-cycle inequalities. For a point of the LP relaxation, a violated flip inequality can be found in pseudo-polynomial time, and even in linear time for a spanning tree solution. Our main result is that the integer vertices of the polytope described by the flip inequalities are exactly the vertices of the PESP polytope, i.e., the convex hull of all feasible periodic slacks with corresponding modulo parameters. Moreover, we show that this flip polytope equals the PESP polytope in some special cases. On the computational side, we devise several heuristic approaches concerning the separation of cutting planes from flip inequalities. We finally present better dual bounds for the smallest and largest instance of the benchmarking library PESPlib.

2012 ACM Subject Classification Mathematics of computing → Combinatorial optimization; Applied computing → Transportation; Theory of computation → Network optimization

Keywords and phrases Periodic Event Scheduling Problem, Periodic Timetabling, Mixed Integer Programming

Digital Object Identifier 10.4230/OASIcs.ATMOS.2020.5

Funding *Niels Lindner*: Funded by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – The Berlin Mathematics Research Center MATH+ (EXC-2046/1, project ID: 390685689).

1 Introduction

Whenever certain processes to be planned shall repeat after a fixed amount of time, periodic plans (or cyclic plans) are sought. Such periodically repeating processes appear in particular in timetables for many public transportation networks, including railway systems, in Europe [4], where period times of 10 minutes or one hour can be observed regularly. One further example is the planning of traffic light signals in street networks. These often follow a periodic pattern, where the period time sometimes is 60 or 90 seconds [8, 27].

In a sense, a better understanding of mathematical models for periodic networks potentially could reduce emissions of the traffic and transportation sector: First, better timetables for public transport that require less transfer or waiting times make public transport more attractive and could thus reduce car traffic. Second, the better systems of traffic lights in networks are coordinated, the less red light stops – and thus less emissions from accelerating and decelerating – are necessary.

Since the work by Serafini and Ukovich [26], planning for periodic networks is mainly done with the *periodic event scheduling problem* (PESP) as graph-based mathematical model. This has attracted much research, presumably also because it turns out to be somehow

20th Symposium on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2020).
Editors: Dennis Huisman and Christos D. Zaroliagis; Article No. 5; pp. 5:1–5:18
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

challenging: One relatively small, but also relatively difficult PESP-based instance has been included into the MIPLIB 2003 [1]. In a more recent collection dedicated exclusively to PESP instances, PESPlib, since 2012 for none of the 20 instances any solution could be proven to be optimal [6].

In order to come up with provably optimal solutions, the well-known branch-and-bound procedure (including its variants such as branch-and-cut) is the only technique that can be applied practically to this purpose. This procedure is based on primal feasible solutions on the one hand, and dual bounds – in the case of a minimization problem, lower bounds – on the other hand. In Fig. 1, we provide an evolution of the values of primal feasible solutions and lower bounds over time, which is typical when solving PESP instances: The dual bounds stay much longer significantly far away from the actual optimal solution value than the primal solutions. Similar observations can be found in [15]. This behavior is also mirrored by the facts that the LP relaxation of a PESP instance always has a trivial solution, and that PESP generalizes the notoriously hard graph vertex coloring problem [23], including certain results concerning inapproximability [12], and parameterized complexity [19].

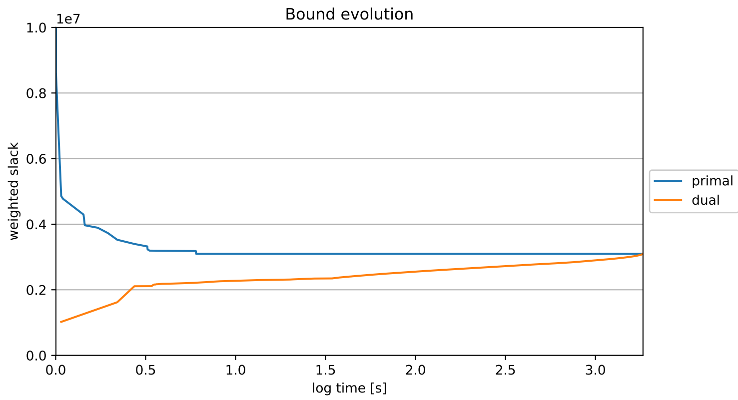

Figure 1 Typical bound evolution when solving a PESP instance by MIP methods (here: CPLEX 12.10 [9] with default settings). The time axis is logarithmic. On this instance, the primal bound stops moving after 10 seconds (x-axis value 1.0, i.e., $10^{1.0} = 10$ seconds), but proving optimality takes 30 minutes.

Hence, in order to really solve PESP instances, much better dual bounds are necessary. From the early years of the active work with PESP, some well-known classes of valid inequalities have been identified: the so-called *cycle inequalities* due to Odijk [23] as well as the so-called *change-cycle inequalities* by Nachtigall [21, 22]. Both are defined for oriented cycles of the graph. In the absence of backward arcs in the oriented cycles, these two classes of valid inequalities coincide [11].

In the sequel, there have been a few contributions regarding the generation of better lower bounds during the branch-and-bound solution process for PESP instances. The *node-disjoint chain inequalities* by Nachtigall [22] consider several internally vertex-disjoint paths between a pair of vertices, and are facet-defining in some cases. T. Lindner [20] investigates *chain cutting planes*, also based on multiple paths between a pair of vertices, and *flow inequalities*. Liebchen and Swarat [17] inspect the second Chvátal closure and propose what they denote *multi-circuit cuts*, which can be defined for structures different from simple oriented circuits. Lindner and Liebchen [18] apply the concept of graph separators to PESP instances. Initially motivated by generating better primal solutions, on some instances it turned out that also better dual solutions could be obtained.

In this paper, we revisit in particular the change-cycle inequalities. In the case of a simple oriented circuit having k arcs, we do not just consider the two initial versions of them, when traversing the circuit either in forward or backward direction. Rather, we consider 2^k different configurations for its arcs, by simply flipping them independently from each other in their initial or in their opposed orientation. All of them turn out to provide valid inequalities. These *flip inequalities* are of course exponentially many, both because of the number of circuits in a graph and because of the proposed arc flip operation.

Nevertheless, considering all these exemplars of the flip inequalities and adding them to the LP relaxation P_{LP} of the integer PESP polytope P_{IP} yields a new polytope P_{flip}. We prove that any vertex of P_{IP} turns out to be a vertex of P_{flip}, too, hereby illustrating the sharpness of these flipped change-cycle inequalities. Yet, it turns out that P_{flip} ends up with further fractional vertices. For example, for an infeasible PESP instance that has been considered in [17], we find that $P_{\text{flip}} \neq \emptyset$, whereas of course $P_{\text{IP}} = \emptyset$. In contrast, for the special case that any arc is contained in at most one cycle, it turns out that $P_{\text{IP}} = P_{\text{flip}}$.

Given a point of the LP relaxation P_{LP}, a violated flip inequality can be separated in pseudo-polynomial time as a consequence of the results of [2]. However, this method is computationally too challenging on large instances, and this is why we examine several heuristic separation strategies for flip inequalities. These turn out to be fruitful, and we compute better dual bounds for the smallest and largest PESPlib railway timetabling instances.

The paper is organized as follows: After formally describing PESP and reviewing the two common mixed integer programming formulations, we define the PESP polytope and recall the cycle and change-cycle inequalities in Section 2. Section 3 is devoted to the flip inequalities and their polyhedral investigation, including our main results and several examples. Our approach to separate flip inequalities in practice is illustrated in Section 4, before we conclude the paper in Section 5.

2 Polyhedral Basics of the Periodic Event Scheduling Problem

2.1 The Periodic Event Scheduling Problem

The *Periodic Event Scheduling Problem* (PESP) dates back to Serafini and Ukovich [26], and shows certain similarities to models that were already considered by Rüger [24]. We will use the following formalization: A PESP instance is given by a (G, T, ℓ, u, w), where

- $G = (V, A)$ is a directed graph, called *event-activity network*, whose vertices are called *events* and whose arcs are called *activities*,
- $T \in \mathbb{N}$ is a *period time*,
- $\ell \in \mathbb{Z}^A_{\geq 0}$ is a vector of *lower bounds* such that $0 \leq \ell < T$,
- $u \in \mathbb{Z}^A_{\geq 0}$ is a vector of *upper bounds*, $0 \leq u - \ell < T$, and
- $w \in \mathbb{R}^A_{\geq 0}$ is a vector of *weights*.

In this paper, we restrict ourselves to integer bounds ℓ and u. This is a common planning assumption, in particular time input values are often scaled and/or rounded accordingly. Furthermore, we assume that G is weakly connected. A vector $\pi \in [0, T)^V$ is a *periodic timetable* if there exists a *periodic tension* $x \in \mathbb{R}^A$ such that

$$\ell \leq x \leq u \quad \text{and} \quad \forall\, a = (i, j) \in A : \quad \pi_j - \pi_i \equiv x_a \bmod T.$$

A periodic timetable π assigns times modulo T to each event in G, and fixes the duration of each activity $a = (i, j) \in A$ to $\pi_j - \pi_i$ modulo T. The actual duration of a is then chosen to lie in the interval $[\ell_a, u_a]$. Since $0 \leq u_a - \ell_a < T$ for all $a \in A$, the periodic tension x is unique for a given timetable π, and can be computed by setting

$$x_a := [\pi_j - \pi_i - \ell_a]_T + \ell_a \quad \text{for all } a = (i,j) \in A,$$

where $[\cdot]_T$ denotes the modulo T operator with values in $[0,T)$. We further define the *periodic slack* as $y := x - \ell \in [0,T)^A$.

In a public transport context, an event i is usually modeling either the arrival or the departure of a directed traffic line at some station, e.g., the departure of the trains from Berlin to Munich in the city of Erfurt. An arc $a = (i,j)$ models the time duration from event i to event j. If i and j are two subsequent departure and arrival events of the same directed line, then $a = (i,j)$ models the trip duration from the station of event i to the station of event j. In turn, if i and j are the arrival and departure events of the same directed line within the same station, then $a = (i,j)$ models the dwell duration within this station. To illustrate many other commercial and operational types of constraints, we refer to [13]. If in an hourly service (i.e., $T = 60$), for a dwell arc $a = (i,j)$ we require that $\ell_a = 3$ and $u_a = 7$, then of course $\pi_i = 29$ and $\pi_j = 33$ constitute a periodic timetable. The periodic tension of a is $x_a = 4 \in [3,7]$, and the periodic slack is $y_a = 1$. However, notice that $\pi_i = 58$ and $\pi_j = 3$ constitute a periodic timetable, too, because $x_a = [3 - 58 - 3]_{60} + 3 = 2 + 3 = 5$.

▶ **Definition 1.** *Given* (G,T,ℓ,u,w) *as above, the* Periodic Event Scheduling Problem (PESP) *is to find a periodic timetable* π *with periodic slack* y *such that* $\sum_{a \in A} w_a y_a$ *is minimum or to decide that no periodic timetable exists.*

2.2 Mixed Integer Programming Formulations

Let (G,T,ℓ,u,w) be a PESP instance, $G = (V,A)$. It follows immediately from the definitions of periodic timetables, tensions and slacks that PESP can be written as:

$$\begin{aligned}
\text{Minimize} \quad & \sum_{a \in A} w_a y_a & \\
\text{s.t.} \quad & \pi_j - \pi_i = y_a + \ell_a - T p_a, & a = (i,j) \in A,\\
& 0 \le \pi_i < T, & v \in V,\\
& 0 \le y_a \le u_a - \ell_a, & a \in A,\\
& p_a \in \mathbb{Z}, & a \in A.
\end{aligned}$$

The variables p_a resolve the modulo T constraints. If $D \in \{-1,0,1\}^{V \times A}$ denotes the incidence matrix of G, and D^t is its transpose, then the PESP constraints can be summarized as $D^t\pi - y = \ell - Tp$. Since the matrix $(D^t \mid -I)$ is totally unimodular, it follows that if the problem is feasible, then there is an optimal integral periodic timetable with an optimal integral periodic slack.

Another formulation is obtained by *cycle bases* of G: An *oriented cycle* in G is a vector $\gamma \in \{-1,0,1\}^A$ with $D\gamma = 0$. Such a γ corresponds to an undirected, possibly non-simple cycle in G on the arcs a with $\gamma_a \neq 0$, where arcs with $\gamma_a = 1$ are traversed forward, i.e., following the direction given by a, and arcs with $\gamma_a = -1$ are traversed backward. We will sometimes decompose $\gamma = \gamma^+ - \gamma^-$ into its positive and negative part, and we denote by $|\gamma|$ the length of the cycle, i.e., the number of $a \in A$ with $\gamma_a \neq 0$. If D is seen as a linear map of $\mathbb{Z}$-modules, the kernel of D is called the *cycle space* of G, and its rank is the *cyclomatic number* μ. An *integral cycle basis* of G is a collection $B = \{\gamma_1, \dots, \gamma_\mu\}$ of oriented cycles generating the cycle space of G as a $\mathbb{Z}$-module. The matrix Γ with $\gamma_1, \dots, \gamma_\mu$ as rows is called a *cycle matrix* and the kernel of Γ equals the image of D^t over $\mathbb{Z}$ [14]. This results in the following cycle-based mixed-integer programming formulation for PESP:

$$\begin{aligned} \text{Minimize} \quad & \sum_{a \in A} w_a y_a \\ \text{s.t.} \quad & \Gamma(y+\ell) = Tz, \\ & 0 \leq y \leq u - \ell, \\ & y \in \mathbb{Z}^A, \\ & z \in \mathbb{Z}^B. \end{aligned} \tag{$\star$}$$

By the above discussion on total unimodularity, it is no restriction to assume that y is integral. An important subclass of integral cycle bases is given by *(strictly) fundamental cycle bases*: A *spanning tree* S on G is a spanning tree on the graph that results from undirecting G. The μ fundamental cycles of S give rise to simple oriented cycles in G, and these form an integral cycle basis [16].

2.3 Periodic Timetabling Polytopes

We will base our polytopal investigations on the cycle-based integer programming formulation ($\star$) for PESP. Let (G, T, ℓ, u, w) be a PESP instance. Fix a cycle matrix Γ of an integral cycle basis B. Let further $n := |V|$, $m := |A|$, and denote by $\mu = m - n + 1$ the cyclomatic number of G.

▶ **Definition 2.** *Define*

$$\begin{aligned} P_{LP} &:= \{(y,z) \in \mathbb{R}^A \times \mathbb{R}^B \mid \Gamma(y+\ell) = Tz, 0 \leq y \leq u - \ell\}, \\ P_{IP} &:= \operatorname{conv}(P_{LP} \cap (\mathbb{Z}^A \times \mathbb{Z}^B)). \end{aligned}$$

That is, P_{IP} is the convex hull of the set of feasible solutions to the integer program ($\star$), and P_{LP} is the set of feasible solutions to the linear programming relaxation of ($\star$).

Since our further investigations will regularily touch on vertices, recall the following basic theorem on the structure of polytopes.

▶ **Theorem 3** ([25, Theorem 5.7])**.** *Let $P = \{x \mid Ax \leq b\}$ be a polyhedron in $\mathbb{R}^r$ and let $x^* \in P$. Then x^* is a vertex of P, if and only if the submatrix A_{x^*} of the inequalities from $Ax \leq b$ that are satisfied by x^* with equality has rank r.*

▶ **Lemma 4.** *The vertices of P_{LP} are given by*

$$\left\{ \left(y, \frac{\Gamma(y+\ell)}{T} \right) \in \mathbb{R}^A \times \mathbb{R}^B \,\middle|\, \forall a \in A: \, y_a \in \{0, u_a - \ell_a\} \right\}.$$

A proof of Lemma 4 is given in the appendix. In particular, P_{LP} has 2^m vertices. Since the weights w are non-negative by definition, we also conclude that $(y^*, z^*) = (0, \Gamma\ell/T)$ is an optimal solution to the the LP relaxation of ($\star$).

▶ **Definition 5.** *A point $(y^*, z^*) \in P_{LP}$ is called a* spanning tree solution *if there is a spanning tree S of G such that $y^*_a = 0$ or $y^*_a = u_a - \ell_a$ holds for all arcs a in S.*

▶ **Theorem 6** (see also [22, Theorem 6.1])**.** *Let (y^*, z^*) be a vertex of P_{LP} or P_{IP}. Then (y^*, z^*) is a spanning tree solution.*

Proof. See appendix. ◀

Note that Theorem 6 does not give a sufficient criterion for being a vertex of P_{IP}: Not every choice of $y_a \in \{0, u_a - \ell_a\}$ along arcs a of some spanning tree yields a vertex of P_{IP}, e.g., if there is no periodic timetable, then $P_{\mathrm{IP}} = \emptyset$, see also Example 19.

2.4 Known Inequalities

Both polyhedra P_{LP} and P_{IP} are polytopes, as the bounds on y imply bounds on z. For P_{IP}, this observation leads to the *cycle inequalities*:

▶ **Lemma 7** (Cycle inequalities, [23]). *Let γ be an oriented cycle and $(y, z) \in P_{IP}$. Then*

$$\left\lceil \frac{\gamma_+^t \ell - \gamma_-^t u}{T} \right\rceil \leq \frac{\gamma^t (y + \ell)}{T} \leq \left\lfloor \frac{\gamma_+^t u - \gamma_-^t \ell}{T} \right\rfloor.$$

Another type of inequalities is the following:

▶ **Lemma 8** (Change-cycle inequalities, [21]). *Let γ be an oriented cycle and $(y, z) \in P_{IP}$. Set $\alpha := [-\gamma^t \ell]_T$. Then*

$$(T - \alpha)\gamma_+^t y + \alpha \gamma_-^t y \geq \alpha(T - \alpha).$$

The change-cycle inequalities are facet-defining for $\alpha > 0$ [22, Lemma 6.4].

Moreover, as mentioned in Section 1, more types of inequalities have been discovered. We will return to the multi-circuit cuts of [17] in Example 19. In the next section, we present a new and easy to describe class of inequalities that applies to each oriented cycle and generalizes both cycle and change-cycle inequalities.

3 Flipping Arcs

3.1 Flip Inequalities

Consider an arc $a = ij \in A$. By *flipping* a, we mean the following: Replace a by an arc $\overline{a} = ji$ in opposite direction, and set $\ell_{\overline{a}} := [-u_a]_T$, $u_{\overline{a}} := [-u_a]_T + u_a - \ell_a$.

▶ **Lemma 9.** *A vector $y \in \mathbb{R}^A$ is a feasible periodic slack for the original PESP instance if and only if y' defined by $y'_{\overline{a}} := u_a - \ell_a - y_a$ and agreeing with y for all other arcs in $A \setminus \{a\}$ is a feasible periodic slack for the PESP instance in which the arc a is just flipped.*

Proof. It is clear that $0 \leq y \leq u - \ell$ implies $0 \leq y' \leq u - \ell$ and vice versa. Let π be a periodic timetable for the original PESP instance. Then, from $\pi_j - \pi_i \equiv y_a + \ell_a \bmod T$,

$$\pi_i - \pi_j \equiv -(y_a + \ell_a) \equiv y'_{\overline{a}} - u_a \equiv y'_{\overline{a}} + [-u_a]_T \mod T,$$

so that π is also a feasible periodic timetable for the flipped instance, and conversely. ◀

Applying the change-cycle inequality (Lemma 8) on the PESP instance obtained by flipping a subset of arcs, and re-interpreting it in the initial instance yields the *flip inequalities*:

▶ **Corollary 10.** *Let $F \subseteq A$ and let γ be an oriented cycle. Then the* flip inequality

$$\begin{aligned}(T - \alpha_F) \sum_{\substack{a \in A \setminus F: \\ \gamma_a = 1}} y_a + \alpha_F \sum_{\substack{a \in A \setminus F: \\ \gamma_a = -1}} y_a \\ + \alpha_F \sum_{\substack{a \in F: \\ \gamma_a = 1}} (u_a - \ell_a - y_a) + (T - \alpha_F) \sum_{\substack{a \in F: \\ \gamma_a = -1}} (u_a - \ell_a - y_a) \quad \geq \quad \alpha_F (T - \alpha_F)\end{aligned}$$

is valid for all $(y, z) \in P_{IP}$*, where*

$$\alpha_F := \left[- \sum_{a \in A \setminus F} \gamma_a \ell_a - \sum_{a \in F} \gamma_a u_a \right]_T .$$

Flipping all arcs in F we obtain an oriented cycle γ_F, and the flip inequality for γ is the change-cycle inequality for γ_F in the flipped instance. Figure 2 illustrates this flip operation showing which bounds of which arcs are considered in the respective flip inequality.

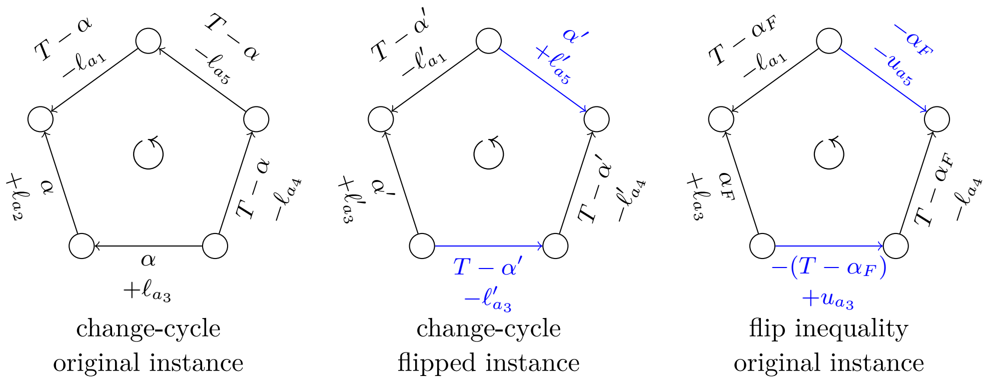

Figure 2 Left: Coefficient of y_{a_i} (top) and contribution to α (bottom) in the original change-cycle inequality for the depicted oriented cycle. Middle: Coefficient of y'_{a_i} and contribution to α' in the original change-cycle inequality in the instance obtained by flipping $F = \{a_3, a_5\}$, with adjusted lower bounds ℓ' and upper bounds u'. Right: Coefficient of y_{a_i} and contribution to α_F in the flip inequality for $F = \{a_3, a_5\}$ on the original instance. In particular, you can see that lower bounds ℓ and upper bounds u can enter the computation of α with arbitrary signs.

Notice that given an oriented cycle γ, initially there had been defined *one* change-cycle inequality making exclusively use of all the lower bounds of its arcs. Much similarly, considering the upper bounds of its arcs had been considered, too [10]. In contrast, Corollary 10 provides us with not less than up to $2^{|\gamma|}$ valid inequalities.

It is easy to see that the (lower bound) cycle inequality and the change-cycle inequality are equivalent for an oriented cycle with no backward arcs [11]. In general, we have:

▶ **Lemma 11.** *Let γ be an oriented cycle. Then the cycle inequalities for γ are equivalent to the flip inequalities when flipping all backward resp. forward arcs in γ.*

Lemma 11 is proved in the appendix. The flip inequalities hence contain both cycle and change-cycle inequalities as special cases. The inequalities with $\alpha_F > 0$ are facet-defining for P_{IP} by the same proof [22, Lemma 6.4] that works for change-cycle inequalities.

3.2 The Flip Polytope

▶ **Definition 12.** *The* flip polytope *is defined as*

$$P_{\mathit{flip}} := \{(y, z) \in P_{LP} \mid y \textit{ satisfies the flip inequality for all } F \subseteq A \textit{ and oriented cycles } \gamma\}.$$

By Corollary 10, we clearly have $P_{\mathrm{IP}} \subseteq P_{\mathrm{flip}} \subseteq P_{\mathrm{LP}}$.

▶ **Theorem 13.** *Let $(y, z) \in P_{LP} \setminus P_{IP}$ be a fractional spanning tree solution. Then $(y, z) \notin P_{\mathit{flip}}$, and any such (y, z) is separated from P_{flip} by at least one of 2μ flip inequalities.*

Proof. Let $(y, z) \in P_{\mathrm{LP}}$, and let $S \subseteq A$ be the set of arcs of a spanning tree such that for all $a \in S$ holds $y_a \in \{0, u_a - \ell_a\}$. We will show that if y satisfies a particular set of 2μ flip inequalities, then (y, z) already turns out to be integer.

Consider any co-tree arc, say $a' \notin S$, with fundamental cycle γ, assuming w.l.o.g. that $\gamma_{a'} = 1$. Suppose that (y, z) satisfies the flip inequalities for γ and the subsets $F_1 := \{a \in S \mid y_a = u_a - \ell_a\}$, $F_2 := F_1 \cup \{a'\}$. Then, because of zero slack of the resulting y-variables (occasionally flipped), the contribution of the arcs in S is 0 in both flip inequalities, so that only

$$(T - \alpha_{F_1}) y_{a'} \geq \alpha_{F_1}(T - \alpha_{F_1}) \quad \text{and} \quad \alpha_{F_2}(u_{a'} - \ell_{a'} - y_{a'}) \geq \alpha_{F_2}(T - \alpha_{F_2})$$

remain. Recall from Lemma 8 that in general $0 \leq \alpha < T$, and now suppose that $\alpha_{F_2} > 0$. Then, together with $\alpha_{F_1} < T$,

$$0 \leq \alpha_{F_1} \leq y_{a'} \leq u_{a'} - \ell_{a'} - T + \alpha_{F_2} < T.$$

By the definition of F_1 and F_2, $u_{a'} - \ell_{a'} - T + \alpha_{F_2} \equiv \alpha_{F_1} \mod T$ and $y_{a'} \in [0, T)$, and we conclude that $y_{a'} = \alpha_{F_1}$. By definition of α_{F_1},

$$\gamma^t(y + \ell) = \sum_{\substack{a \in S \\ y_a = 0}} \gamma_a \ell_a + \sum_{\substack{a \in S \\ y_a = u_a - \ell_a}} \gamma_a u_a + y_{a'} + \ell_{a'} \equiv 0 \mod T.$$

In the case $\alpha_{F_2} = 0$, it holds that $y_{a'} \geq \alpha_{F_1} = u_{a'} - \ell_{a'}$, so that again $y_{a'} = \alpha_{F_1}$ and $\gamma^t(y + \ell) \equiv 0 \mod T$.

We conclude that for all μ fundamental cycles γ of S, $\gamma^t(y + \ell)$ is an integer multiple of T. As these cycles form an integral cycle basis, we find that z is integer, and hence $(y, z) \in P_{\mathrm{IP}}$. ◀

Theorem 13 provides a linear-time separation procedure for spanning tree solutions. In general, there is a pseudo-polynomial time separation algorithm:

▶ **Theorem 14.** *There is an $O(T^2 n^2 m)$ algorithm that given $(y, z) \in P_{LP}$ finds a flip inequality violated by (y, z) or decides that none exists.*

Proof. Construct a network G' as follows: Remove each arc $a = ij \in A$ and insert instead four arcs $is_a, s_a t_a, t_a s_a, t_a j$, where s_a, t_a are new vertices. The arc $s_a t_a$ receives the bounds of a, while $t_a s_a$ obtains the bounds of the flipped arc $\overline{a}$ as in the beginning of this section. The other two arcs have lower and upper bound 0.

In this network G', any oriented cycle γ' either consists only of $s_a t_a$ and $t_a s_a$ for some $a \in A$, or it uses at most one of $s_a t_a$ and $t_a s_a$. In the latter case, γ' corresponds to a pair (γ, F), where γ is an oriented cycle in G and F consists of the arcs in G where γ' uses $t_a s_a$. Moreover, the change-cycle inequality for γ' in G' is equivalent to the flip inequality for γ in G w.r.t. F. On the other hand, the change-cycle inequality for a cycle γ' on the arcs $s_a t_a$ and $t_a s_a$ is satisfied for any y: Assuming that both arcs are forward, we have that

$$(T - \alpha) y_a + (T - \alpha)(u_a - \ell_a - y_a) = (T - \alpha)(u_a - \ell_a) = \alpha(T - \alpha), \quad \text{as } \alpha = u_a - \ell_a.$$

The analogous result holds when both arcs in γ' are backward. As a consequence, we can find violated flip inequalities in G by separating change-cycle inequalities in G', which can be done in $O(T^2 n^2 m)$ time [2, Theorem 10]. ◀

The complexity of the separation problem remains open, a few partial NP-completeness results are known for cycle and change-cycle inequalities [2].

We present now an astonishing result on the relation between P_{flip} and P_{IP}:

▶ **Theorem 15.** *The vertices of P_{IP} are precisely the integer vertices of P_{flip}.*

Proof. It is clear that any integer vertex of P_{flip} is a vertex of P_{IP}. Now, let (y^*, z^*) be a vertex of P_{IP}. In view of Theorem 3, we need to identify $m + \mu$ linearly independent defining inequalities of P_{flip} that are satisfied with equality for (y^*, z^*), and we are doing so in three sets:

- Tree arcs ($n - 1$ inequalities):
  By Theorem 6, (y^*, z^*) is a spanning tree solution, denote the set of arcs of the spanning tree by S. It follows that (y^*, z^*) satisfies $n - 1$ linearly independent inequalities of the form $y_a \geq 0$ or $y_a \leq u_a - \ell_a$ for $a \in S$ with equality.
- Co-tree arcs ($m - (n - 1)$ inequalities):
  By the proof of Theorem 13, for each co-tree arc $a \notin S$, we have $y^*_a = \alpha_{F_1}$, and this is a flip inequality satisfied with equality. There are of course $\mu = m - (n - 1)$ co-tree arcs. Observe that these are linearly independent with the ones identified for the tree arcs.
- Cycle periodicity constraints (μ inequalities):
  Finally, we obtain μ equality constraints from $\Gamma y^* - T z^* = \Gamma \ell$, one constraint for each z-variable.

In total, we have found $(n - 1) + \mu + \mu = m + \mu$ linearly independent defining inequalities of P_{flip} that are satisfied with equality. Hence (y^*, z^*) is a vertex of P_{flip}. ◀

The following theorem, whose proof can be found in the appendix, states that the flip inequalities (together with the slack bounds) fully describe P_{IP} on PESP instances with $\mu \leq 1$, and also on networks with higher cyclomatic number provided that the arc set of any two distinct cycles is disjoint.

▶ **Theorem 16.** *Suppose that each arc $a \in A$ is contained in at most one (undirected) cycle. Then $P_{flip} = P_{IP}$.*

3.3 Examples of Flip Polytopes

▶ **Example 17** (Integral flip polytope)**.** The PESP instance depicted in Figure 3 has cyclomatic number $\mu = 1$. In particular, Theorem 16 implies $P_{\text{flip}} = P_{\text{IP}}$. The polytope is drawn in Figure 4.

▶ **Example 18** (Non-integral flip polytope)**.** It is possible that P_{flip} contains fractional vertices when an arc belongs to more than one cycle. In the instance from Figure 5 with cyclomatic number 2, a computation with `polymake` [5] revealed that P_{flip} has 24 vertices from P_{IP}, but also 39 fractional vertices. For example, $(y_{12}, y_{23}, y_{31}, y_{34}, y_{42}, z_1, z_2) = (7.7, 2.1, 4.2, 6.5, 0.4, 1.7, 1.1)$ is such a vertex.

▶ **Example 19** (An infeasible PESP instance)**.** The PESP instance on the wheel graph in Figure 6 is infeasible. Adding the cycle and change-cycle inequalities to P_{LP} results in a non-empty polytope. The second Chvátal closure $P^{(2)}_{\text{LP}}$ of P_{LP} is empty, and the emptyness is certified by two *multi-circuit cuts* [17].

Due to Theorem 15, the flip polytope can be used to detect that no integer points exist as well: An instance is infeasible if and only if P_{flip} has no integer vertices. We use `polymake` to compute the vertices of P_{flip} on the wheel instance. It turns out that P_{flip} is zero-dimensional with a single fractional vertex with slack $\frac{1}{2}$ on all spokes and 2 on all arcs of the outer cycle. However, $P_{\text{flip}} \neq \emptyset$, so the flip inequalities differ from the multi-circuit cuts. Recall from [17] that the change-cycle inequalities can have Chvátal rank $\geq \frac{T}{2}$, so does the superclass of flip

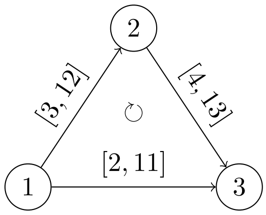

Figure 3 PESP instance from Example 17, arcs a are labeled with $[\ell_a, u_a]$, $T = 10$.

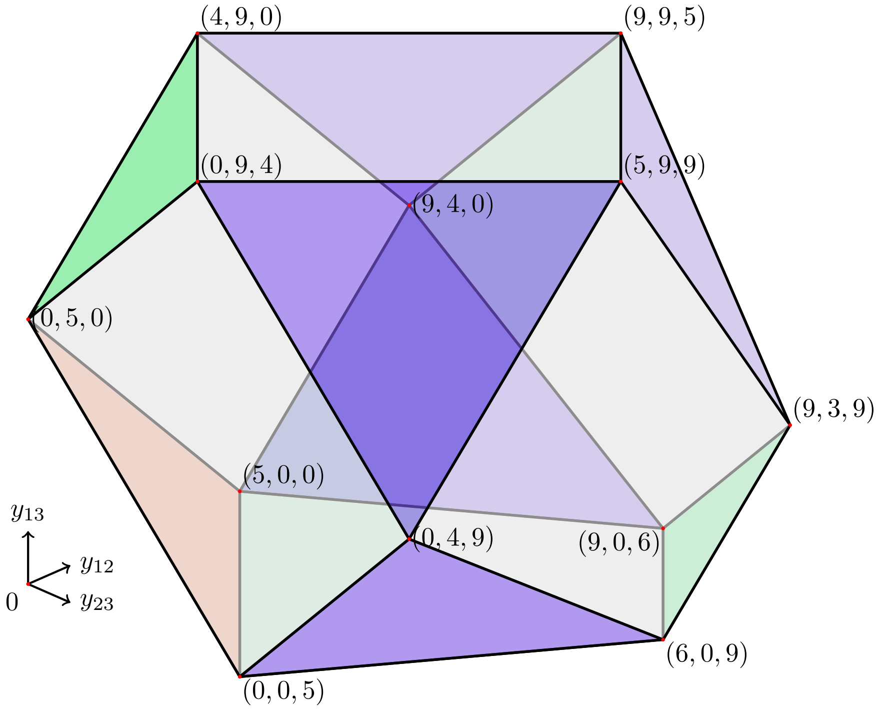

Figure 4 The polytope $P_{\text{IP}} = P_{\text{flip}}$ for the instance in Figure 3 has 12 vertices, 24 edges, 14 facets, and is combinatorially equivalent to a cuboctahedron. The vertices are labeled with (y_{12}, y_{13}, y_{23}). Variable bounds are drawn in light grey, the cycle inequalities $y_{12} - y_{13} + y_{23} \geq -5$ ($z \geq 0$, containing the vertices $(0,5,0)$, $(0,9,4)$ and $(4,9,0)$) and $y_{12} - y_{13} + y_{23} \leq 15$ ($z \leq 2$, containing the vertices $(6,0,9)$, $(9,0,6)$, and $(9,3,9)$) are green, the unflipped change-cycle inequality $5y_{12} + 5y_{13} + 5y_{23} \geq 25$ is red, and the remaining five flip inequalities are drawn blue. The light green hexagon in the center is the hyperplane for $z = 1$ (containing the vertices $(0,0,5)$, $(0,4,9)$, $(5,0,0)$, $(5,9,9)$, $(9,4,0)$ and $(9,9,5)$). We used `polymake` [5] for computations and visualization.

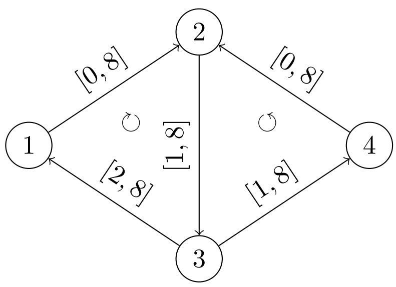

Figure 5 PESP instance from Example 18, arcs a are labeled with $[\ell_a, u_a]$, $T = 10$.

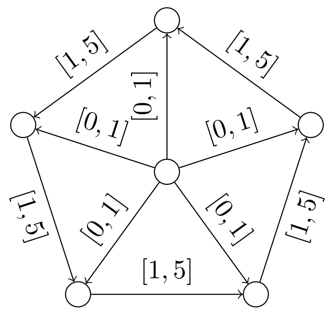

Figure 6 Wheel instance from Example 19, arcs a are labeled with $[\ell_a, u_a]$, $T = 6$.

inequalities. Hence flipping arcs, as proposed in this paper, in a sense can be considered to be kind of complimentary to exploiting the concept of Chvátal closures for the first such closures.

4 Separating Flips in Practice

The flip polytope reveals the vertices of P_{IP} by Theorem 15. Moreover, the flip inequalities are facet-defining, and can be separated in pseudo-polynomial time by Theorem 14. Hence this type of inequalities is a reasonable target for cutting plane approaches. Since the general separation algorithm from [2] is a Bellman-Ford-type dynamic program, which is practically infeasible due to both time and space consumption, one has to come up with different separation strategies.

A successful heuristic strategy for separating cycle and change-cycle inequalities in a branch-and-cut context is to build a minimum spanning tree S w.r.t. the slack of the current LP relaxation and to add violated inequalities for the fundamental cycles of S. This approach [3] produced the current best known dual bounds for all 20 instances of the PESP benchmarking library PESPlib [6]. We will call this the *standard* approach.

Building on top of the standard approach on a minimum slack spanning tree S, we consider the following heuristic separation algorithms:

- *all-flip*: For each fundamental cycle γ of S, add all violated cycle and change-cycle inequalities, as well as all violated flip inequalities obtained by flipping a single arc of γ.
- *max-flip*: For each fundamental cycle γ of S, add all violated cycle and change-cycle inequalities, and the maximally violated flip inequality among all single-arc flips of γ.
- *all-flip-hybrid* resp. *max-flip-hybrid*: As all-flip resp. max-flip, but consider flips only if less than a fixed number of violated (change-)cycle inequalities have been found in the standard approach.
- *all-flip-hybrid-small* resp. *max-flip-hybrid-small*: We precompute all cycles of length $\leq k$, and also all up to 2^k flip inequalities for each of these cycles. The strategy is then as in all-flip-hybrid resp. max-flip-hybrid, but with another round that adds violated flip inequalities from the precomputed pool whenever all-/max-flip-hybrid does not produce sufficiently many cuts. In this round, the all-version adds all violated flip inequalities, whereas the max-version adds only the maximally violated flip inequality for each small cycle.

Conceptually, the standard approach adds the least cuts, and all-flip the most. Less cuts mean smaller LPs, which is beneficial concerning running time and memory. On the other hand, more cuts should offer better quality. Our goal is to analyze this trade-off. We always include the separation of the cycle and change-cycle inequalities, as they belong to the class of the flip inequalities, and it suffices to validate only one cycle inequality and one change-cycle inequality per cycle.

Table 1 Overview of our set of instances. The -0.6 suffix indicates that free arcs whose weight sums up to 60% of the total free weight have been deleted [7].

Instance	Hardness	n	m	μ
R1L1-0.6	easy	125	225	101
R4L4-0.6	medium	506	960	455
R1L1	hard	3 664	6 385	2 722
R4L4	extreme	8 384	17 754	9 371

We compare these strategies on four instances derived from the PESPlib set, see Table 1. The separation strategies have been implemented in the concurrent PESP solver from [3] using CPLEX 12.10 [9] as underlying MIP solver. We choose the cycle-based MIP formulation ($\star$) and compute a minimum weight cycle basis in the sense of [14] in order to keep the branch-and-bound tree small (except for R4L4, where our implementation runs out of memory). With the current PESPlib incumbent (i.e., the solution with the smallest objective value according to [6] as of June 2020) as initial solution, we let the PESP solver run for 12 hours on up to 6 internal CPLEX threads with best bound MIP emphasis. The computations were carried out on an Intel Xeon E3-1245 v5 CPU running at 3.5 GHz with 32 GB RAM.

Table 2 Summary of computational results.

Instance	PESPlib dual bound	new dual bound	best strategy	optimality gap
R1L1-0.6	–	5 681 843	standard	0.0 %
R4L4-0.6	–	5 245 781	max-flip-hybrid	34.4 %
R1L1	19 878 200	20 230 655	max-flip-hybrid	33.5 %
R4L4	15 840 600	17 961 400	standard	53.2 %

The results are presented in Figures 7 and 8 (in the appendix), and summarized in Table 2. As the two -0.6 instances are not part of the PESPlib, there are no official dual bounds available. On the easiest instance R1L1-0.6, all approaches prove optimality, the standard approach being the fastest. The potentially higher quality of the flip inequality methods is outweighed by the speed of the standard approach. The picture for R4L4-0.6 is different: The standard approach performs best only within the first 5 minutes, and after roughly one and a half hours, it is outperformed by all other methods. The winner here is max-flip-hybrid, the difference to all-flip and all-flip-hybrid-small (here with cycles of length ≤ 8) being minor.

On the instances R1L1 and R4L4, all-flip requires too much memory and terminates rather early. For the small cycles, we set a length bound of 4. The winner on R1L1 is again max-flip-hybrid with all-flip-hybrid as runner-up. For the last two hours, the standard approach eventually becomes dominated by all other approaches. On R4L4, standard produces the best bounds within the time limit of 12 hours. However, all algorithms except max-flip ran out of memory. Here, the solver finds plenty of cuts and does not leave the root node for the whole running time, explaining the minor differences between the strategies. We want to remark that, although we use the same method, our dual bound is better than the PESPlib bound, which is due to the slightly longer computation time compared to [3] and to some improvements to the code.

There seems to be a point in time when the speed-quality trade-off shifts from the standard approach towards a flipping strategy such as max-flip-hybrid. We do not reach this point on R1L1-0.6, as the instance is solved to optimality before, and also not on R4L4, as the instance is too large to show a significant difference within 12 hours. However, the positive role of the flip inequalities becomes clearly visible on R4L4-0.6 and R1L1, leading to significantly better bounds. As it seems to us that the instances are similarly structured, we expect that the flip inequality approach is able to compute better dual bounds at least for the smaller PESPlib instances.

5 Conclusion

We generalized the change-cycle inequalities [21] for a PESP instance by considering them in a modified instance that emerges from the initial one simply by flipping some arcs to the opposite direction. We call the resulting set of valid inequalities the *flip inequalities*. These turn out to contain not only of course the change-cycle inequalities, but also the cycle inequalities [23].

From a theoretical point of view, the set of flip inequalities provides a much better understanding of the integer PESP polytope P_{IP}. To assess their power, add *only* the flip inequalities to the LP relaxation P_{LP} of P_{IP} to get another polytope P_{flip}. Then, the integer vertices of this particular polytope P_{flip} turn out to be already precisely the (integer) vertices of P_{IP}. In some special cases, e.g., when the cyclomatic number is one, P_{flip} equals P_{IP}.

From a practical point of view, some first positive effects on dual bounds during branch-and-cut-processes show up in our first computational experiments. But we suppose that there might exist better separation strategies that take even more benefit out of the flip inequalities. Yet, this might not turn out to be trivial, due to the huge number of flip inequalities, both because of the potentially exponential number of cycles in a graph, and the exponentially many possibilities to perform all flips for each of these cycles.

References

1 Tobias Achterberg, Thorsten Koch, and Alexander Martin. MIPLIB 2003. *Oper. Res. Lett.*, 34(4):361–372, 2006. doi:10.1016/j.orl.2005.07.009.

2 Ralf Borndörfer, Heide Hoppmann, Marika Karbstein, and Niels Lindner. Separation of cycle inequalities in periodic timetabling. *Discrete Optimization*, 35:100552, 2020. doi:10.1016/j.disopt.2019.100552.

3 Ralf Borndörfer, Niels Lindner, and Sarah Roth. A concurrent approach to the Periodic Event Scheduling Problem. *Journal of Rail Transport Planning & Management*, pages 100–175, 2019. doi:10.1016/j.jrtpm.2019.100175.

4 Gabrio Caimi, Leo G. Kroon, and Christian Liebchen. Models for railway timetable optimization: Applicability and applications in practice. *J. Rail Transp. Plan. Manag.*, 6(4):285–312, 2017. doi:10.1016/j.jrtpm.2016.11.002.

5 Ewgenij Gawrilow and Michael Joswig. polymake: a framework for analyzing convex polytopes. In *Polytopes—combinatorics and computation (Oberwolfach, 1997)*, volume 29 of *DMV Sem.*, pages 43–73. Birkhäuser, Basel, 2000.

6 Marc Goerigk. PESPlib – A benchmark library for periodic event scheduling, 2012. URL: http://num.math.uni-goettingen.de/~m.goerigk/pesplib/.

7 Marc Goerigk and Christian Liebchen. An improved algorithm for the periodic timetabling problem. In *ATMOS*, volume 59 of *OASICS*, pages 12:1–12:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.

8 Refael Hassin. A flow algorithm for network synchronization. *Operations Research*, 44:570–579, 1996.

9 IBM. IBM CPLEX Optimizer 12.10 user's manual, 2019. URL: https://www.ibm.com/analytics/cplex-optimizer.

10 C. Liebchen. *Periodic Timetable Optimization in Public Transport.* Dissertation.de – Verlag im Internet, 2006.

11 Christian Liebchen. Optimierungsverfahren zur Erstellung von Taktfahrplänen. Diploma Thesis, Technische Universität Berlin, 1998. In German.

12 Christian Liebchen. A cut-based heuristic to produce almost feasible periodic railway timetables. In Sotiris E. Nikoletseas, editor, *Experimental and Efficient Algorithms, 4th International-*

Workshop, WEA 2005, Santorini Island, Greece, May 10-13, 2005, Proceedings, volume 3503 of *Lecture Notes in Computer Science*, pages 354–366. Springer, 2005. `doi:10.1007/11427186_31`.

13 Christian Liebchen and Rolf H. Möhring. The modeling power of the periodic event scheduling problem: Railway timetables - and beyond. In Frank Geraets, Leo G. Kroon, Anita Schöbel, Dorothea Wagner, and Christos D. Zaroliagis, editors, *Algorithmic Methods for Railway Optimization, International Dagstuhl Workshop, Dagstuhl Castle, Germany, June 20-25, 2004, 4th International Workshop, ATMOS 2004, Bergen, Norway, September 16-17, 2004, Revised Selected Papers*, volume 4359 of *Lecture Notes in Computer Science*, pages 3–40. Springer, 2004. `doi:10.1007/978-3-540-74247-0_1`.

14 Christian Liebchen and Leon Peeters. Integral cycle bases for cyclic timetabling. *Discrete Optimization*, 6(1):98–109, 2009. `doi:10.1016/j.disopt.2008.09.003`.

15 Christian Liebchen, Mark Proksch, and Frank Wagner. Performance of algorithms for periodic timetable optimization. In Mark Hickman, Pitu Mirchandani, and Stefan Voß, editors, *Computer-aided Systems in Public Transport*, volume 600 of *Lecture Notes in Economics and Mathematical Systems*, pages 151–180. Springer, 2008. `doi:10.1007/978-3-540-73312-6_8`.

16 Christian Liebchen and Romeo Rizzi. Classes of cycle bases. *Discret. Appl. Math.*, 155(3):337–355, 2007. `doi:10.1016/j.dam.2006.06.007`.

17 Christian Liebchen and Elmar Swarat. The second Chvátal closure can yield better railway timetables. In Matteo Fischetti and Peter Widmayer, editors, *ATMOS 2008 - 8th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems, Karlsruhe, Germany, September 18, 2008*, volume 9 of *OASICS*. Internationales Begegnungs- und Forschungszentrum fuer Informatik (IBFI), Schloss Dagstuhl, Germany, 2008. URL: `http://drops.dagstuhl.de/opus/volltexte/2008/1580`.

18 Niels Lindner and Christian Liebchen. New perspectives on PESP: T-partitions and separators. In Valentina Cacchiani and Alberto Marchetti-Spaccamela, editors, *19th Symposium on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems, ATMOS 2019, September 12-13, 2019, Munich, Germany*, volume 75 of *OASICS*, pages 2:1–2:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. `doi:10.4230/OASIcs.ATMOS.2019.2`.

19 Niels Lindner and Julian Reisch. Parameterized complexity of periodic timetabling. Technical Report 20-15, Zuse Institute Berlin, 2020.

20 Thomas Lindner. *Train Schedule Optimization in Public Rail Transport*. Ph.d. thesis, Technische Universität Braunschweig, 2000.

21 Karl Nachtigall. Cutting planes for a polyhedron associated with a periodic network. Institutsbericht IB 112-96/17, Deutsche Forschungsanstalt für Luft- und Raumfahrt e.V., July 1996.

22 Karl Nachtigall. *Periodic Network Optimization and Fixed Interval Timetables*. Habilitation thesis, Universität Hildesheim, 1998.

23 Michiel A. Odijk. Construction of periodic timetables, part 1: A cutting plane algorithm. Technical Report 94-61, TU Delft, 1994.

24 Siegfried Rüger. *Transporttechnologie städtischer öffentlicher Personenverkehr*. Transpress Verlag für Verkehrswesen, Berlin, 3rd edition, 1986.

25 Alexander Schrijver. *Combinatorial Optimization – Polyhedra end Efficiency*. Algorithms and Combinatorics. Springer, 2003.

26 Paolo Serafini and Walter Ukovich. A mathematical model for periodic scheduling problems. *SIAM Journal on Discrete Mathematics*, 2(4):550–581, 1989.

27 Gregor Wünsch. *Coordination of Traffic Signals in Networks*. Ph.D. thesis, Technische Universität Berlin, 2008.

A Appendix

Proof of Lemma 4. Any feasible solution of P_{LP} satisfies the μ linear independent equations $\Gamma(y+\ell) = Tz$. Hence, by Theorem 3, a vertex must satisfy m linear independent out of the $2m$ inequalities $0 \leq y \leq u - \ell$ with equality. Conversely, let $(y^*, z^*) \in P_{\mathrm{LP}}$ with $y^*_a \in \{0, u_a - \ell_a\}$ for all $a \in A$. Define $c \in \mathbb{R}^A$ by

$$c_a := \begin{cases} 1 & \text{if } y^*_a = 0, \\ -1 & \text{if } y^*_a = u_a - \ell_a, \end{cases} \qquad a \in A.$$

Then (y^*, z^*) is the unique point in P_{LP} minimizing $c^t y$ and hence a vertex of P_{LP}. ◀

Proof of Theorem 6. The statement for P_{LP} follows from Lemma 4. Let (y^*, z^*) be a vertex of P_{IP}. Then y^* is a vertex of the polytope

$$P_{z^*} := \{y \in \mathbb{R}^A \mid \Gamma(y+\ell) = Tz^*, 0 \leq y \leq u - \ell\},$$

because otherwise, if we find a proper convex combination $y^* = \lambda y' + (1-\lambda)y''$ for some $y', y'' \in P_{z^*} \setminus \{y^*\}$ and $\lambda \in (0,1)$, then also $(y^*, z^*) = \lambda(y', z^*) + (1-\lambda)(y'', z^*)$ constitutes a proper convex combination in P_{LP}, thus preventing (y^*, z^*) from being a vertex. Being a vertex of P_{z^*} means that there are $\mu - m = n - 1$ arcs $a \in A$ for which $y^*_a = 0$ or $y^*_a = u_a - \ell_a$ is true, and the subgraph on these arcs must not contain a cycle, as the rows of Γ span the cycle space. So these $n-1$ arcs belong to a spanning tree. ◀

Proof of Lemma 11. Suppose $F = \{a \in A \mid \gamma_a = -1\}$. Then the flip inequality reads as

$$(T - \alpha_F)\gamma_+^t y + (T - \alpha_F)\gamma_-^t(u - \ell - y) \geq \alpha_F(T - \alpha_F),$$

and hence, since $\alpha_F \in [0, T)$,

$$\gamma_+^t y + \gamma_-^t(u - \ell - y) \geq \alpha_F = [-\gamma_+^t \ell + \gamma_-^t u]_T.$$

Adding $\gamma_+^t \ell - \gamma_-^t u$ on both sides, we obtain

$$\gamma^t(y+\ell) \geq \gamma_+^t \ell - \gamma_-^t u + [-\gamma_+^t \ell + \gamma_-^t u]_T = T\left\lceil \frac{\gamma_+^t \ell - \gamma_-^t u}{T} \right\rceil,$$

because of $r + [-r]_T = T\left\lceil \frac{r}{T} \right\rceil$. The other part of the cycle inequality is analogously obtained for $F = \{a \in A \mid \gamma_a = 1\}$. ◀

Proof of Theorem 16. Under the hypotheses of the theorem, we can partition the PESP instance into subinstances consisting either of exactly one cycle each (i.e., $\mu = 1$) or of arcs not contained in any cycle ($\mu = 0$). Observe that $P_{\mathrm{LP}}, P_{\mathrm{flip}}, P_{\mathrm{IP}}$ all decompose as the product of the corresponding polytopes of these subinstances. Since we clearly have $P_{\mathrm{LP}} = P_{\mathrm{IP}}$ if $\mu = 0$, we can hence assume w.l.o.g. that G is a single oriented cycle γ.

By Theorem 6, any vertex (y, z) of P_{IP} is a spanning tree solution. In our case, this means that $y_a \in \{0, u_a - \ell_a\}$ for at least $m - 1 = n - 1$ arcs $a \in A$, and z is already determined by y via $z = \frac{\gamma^t(y+\ell)}{T}$. This means that y is a point on an edge of the projection Q_{LP} of P_{LP} to the slack space, as Q_{LP} is an m-dimensional cube scaled by $u - \ell$ (Lemma 4). Of course, if y_a is at its lower or upper bound for all m edges, then y is a vertex of Q_{LP}. Note that for each cube edge, we find at most one y such that (y, z) is a vertex of P_{IP}. The projection Q_{IP} of P_{IP} to the y-space is the convex hull of all y for $(y, z) \in P_{\mathrm{IP}}$, and is hence combinatorially equivalent to a *(partially) rectified m-cube*: We obtain Q_{IP} from the m-cube Q_{LP} by cutting off each cube vertex q by the hyperplane H_q given by the convex hull of all points y on the edges incident to q stemming from vertices of P_{IP}. The resulting polytope has two types of facets: The up to 2^m hyperplanes H_q and the remainining parts of the $2m$ facets of the original m-cube. The latter are clearly given by the bounds $y_a \geq 0$ and $y_a \leq u_a - \ell_a$.

We claim that all H_q are coming from the flip inequalities. Let q be a vertex of Q_{LP} and define $F := \{a \in A \mid q_a = u_a - \ell_a\}$. Note that $q_a = 0$ for $a \in A \setminus F$. Now F gives rise to a flip inequality. Let $(y, z) \in P_{\mathrm{IP}}$ be a vertex such that y is on an edge of Q_{LP} incident to q. Let a' denote the co-tree arc of the spanning tree associated to (y, z), so that $y_a = q_a$ for all $a \in A \setminus \{a'\}$. If $\gamma_{a'} = 1$ and $a' \notin F$, then the flip inequality for F is $(T - \alpha_F)y_{a'} \geq \alpha_F(T - \alpha_F)$. As $\gamma^t(y + \ell) \equiv 0 \bmod T$ implies $y_{a'} = \alpha_F$ (compare the proof of Theorem 13), the flip inequality is hence satisfied with equality. In the case $\gamma_{a'} = 1$ and $a' \in F$, the flip inequality reads as $\alpha_F(u_{a'} - \ell_{a'} - y_{a'}) \geq \alpha_F(T - \alpha_F)$ and is satisfied with equality, as $\gamma^t(y + \ell) \equiv 0 \bmod T$ implies $u_{a'} - \ell_{a'} - y_{a'} = T - \alpha_F$. The computations for $\gamma_{a'} = -1$ are similar. We conclude that y lies on the hyperplane where the flip inequality for F is tight. In particular, H_q is induced by a flip inequality.

Mapping Q_{IP} back to P_{IP} using the affine transformation $y \mapsto (y, \frac{\gamma^t(y+\ell)}{T})$, we obtain $P_{\mathrm{flip}} = P_{\mathrm{IP}}$. ◀

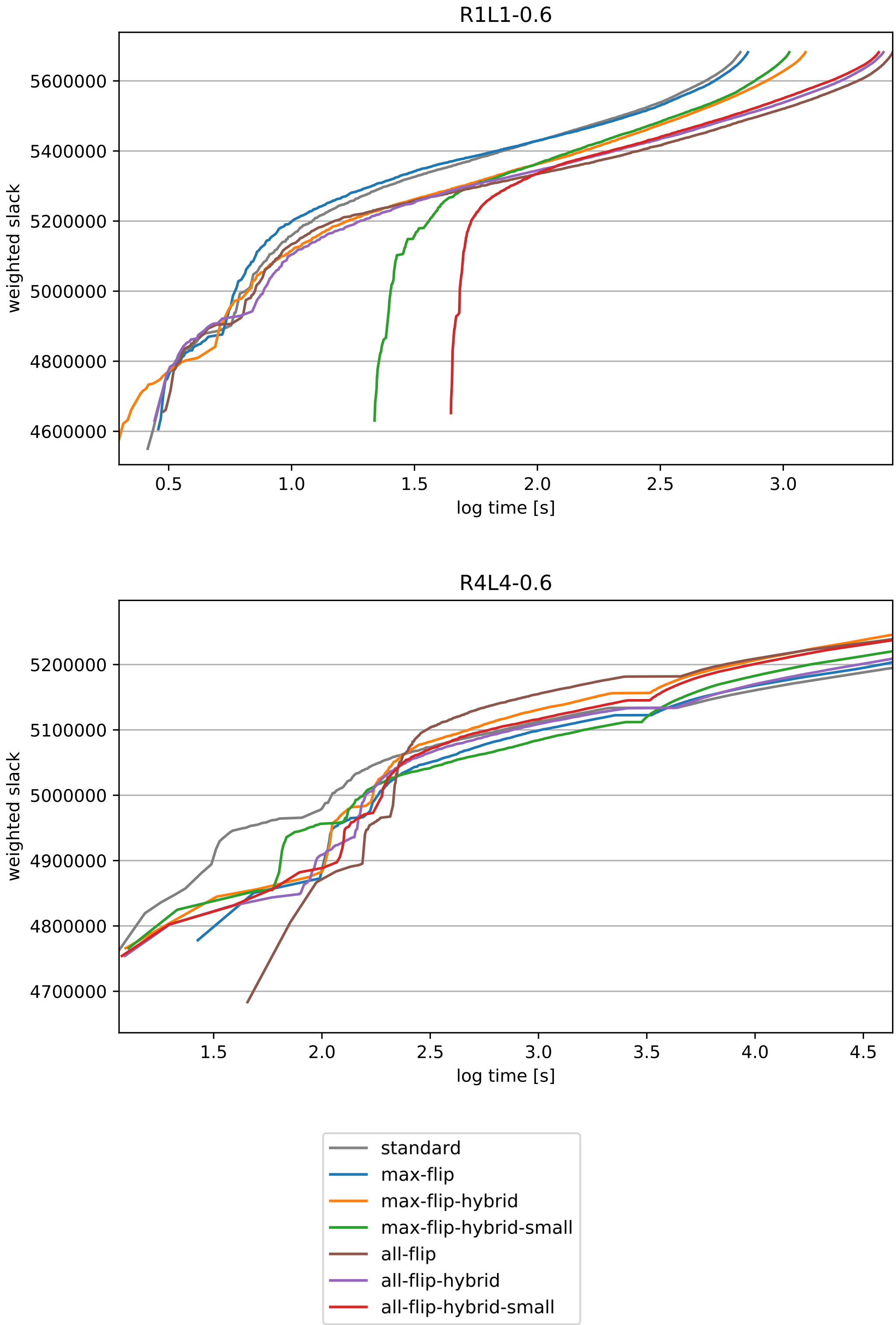

Figure 7 Dual bound evolution on R1L1-0.6 and R4L4-0.6: Dual bound vs. logarithmic time (i.e. 2.0 stands for $10^{2.0} = 100$ seconds of computation time).

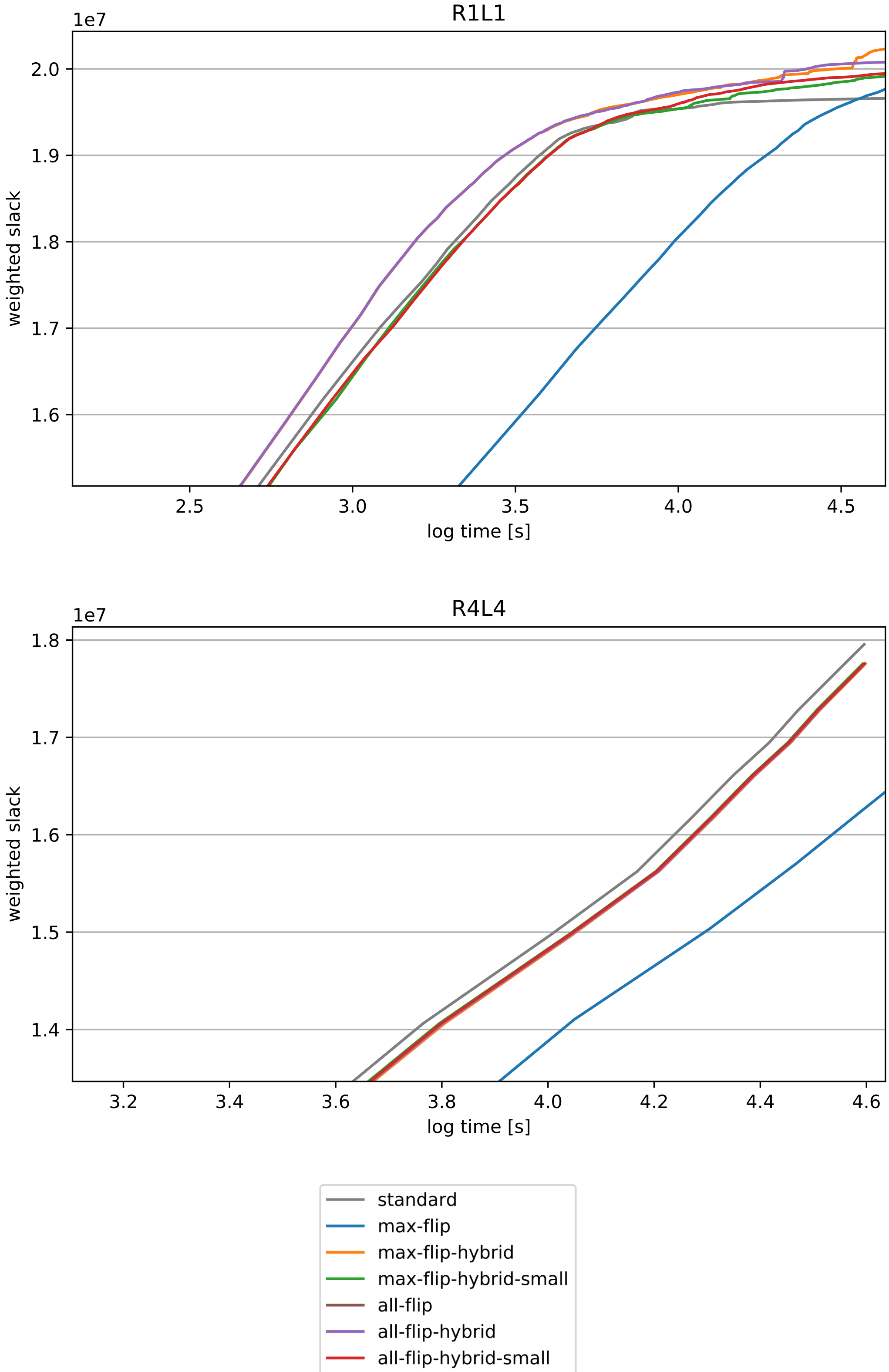

Figure 8 Dual bound evolution on R1L1 and R4L4: Dual bound vs. logarithmic time (i.e. 2.0 stands for $10^{2.0} = 100$ seconds of computation time).