Periodic Event Scheduling with Flexible Infrastructure Assignment

Enrico Bortoletto
Zuse Institute Berlin, Germany

Rolf Nelson van Lieshout
Eindhoven University of Technology, The Netherlands

Berenike Masing
Zuse Institute Berlin, Germany

Niels Lindner
Zuse Institute Berlin, Germany

Abstract

We present novel extensions of the Periodic Event Scheduling Problem (PESP) that integrate the assignment of activities to infrastructure elements. An application of this is railway timetabling, as station and platform capacities are limited and need to be taken into account. We show that an assignment of activities to platforms can always be made periodic, and that it can be beneficial to allow larger periods for the assignment than for the timetable. We present mixed-integer programming formulations for the general problem, as well as for the practically relevant case when multiple platforms can be considered equivalent, for which we present a bipartite matching approach. We finally test and compare these models on real-world instances.

2012 ACM Subject Classification Mathematics of computing → Integer programming; Applied computing → Transportation; Mathematics of computing → Combinatorial optimization; Mathematics of computing → Matchings and factors

Keywords and phrases Periodic Event Scheduling, Periodic Timetabling, Railway Timetabling, Matchings, Infrastructure Assignments, Platform Assignments, Station Capacities

Digital Object Identifier 10.4230/OASIcs.ATMOS.2024.4

Funding *Enrico Bortoletto*: Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – The Berlin Mathematics Research Center MATH+ (EXC-2046/1, project ID: 390685689).
Berenike Masing: Funded within the Research Campus MODAL, funded by the German Federal Ministry of Education and Research (BMBF) (fund number 05M20ZBM).

Acknowledgements We thank DB InfraGO AG for providing data for the S-Bahn Berlin network.

1 Introduction

Out of the many interacting pieces of public transportation services, a key determinant in the puzzle is the timetable. This is an omnipresent yet flexible concern in the operators' minds [3], as well as one of the data of most interest to passengers [14]. These are but two of the reasons for which timetabling and in particular periodic timetabling has received substantial attention in the past. The modeling framework of the Periodic Event Scheduling Problem (PESP) was initially formulated by Serafini and Ukovich [16], quickly found to have rich and interesting underlying structures [1, 9, 12, 13], and studied ever since, also in integration with various other problems, such as [5, 10, 11, 15, 17].

Of special interest in this work is the question of solving PESP while making sure that the produced timetable accounts for various infrastructural constraints of high practical interest in railway operations, involving safety of operations and physical occupation of

© Enrico Bortoletto, Rolf Nelson van Lieshout, Berenike Masing, and Niels Lindner;
licensed under Creative Commons License CC-BY 4.0

tracks. In particular, we focus on what is called *track occupation problem* in [11], which entails ensuring that no two vehicles are ever scheduled to occupy the same point in time and space. This is of particular interest, for example, when planning dwelling activities of trains at the same platform. The need to respect a given association of the activities to be scheduled to infrastructure elements led to *Infrastructure-Aware* PESP (IPESP), which can be formulated as a mixed-integer linear program, by using well-known PESP constraints as foundation [2, 11].

We present two main contributions: At first, we generalize IPESP to *Infrastructure-Aware* PESP *with Assignment* (IPESPA), by integrating the assignment of activities to infrastructure elements into the periodic timetabling problem. We show that such an assignment can always be made periodic, without impact on the timetable, and highlight how it can be advantageous for the period of such an infrastructure assignment to be larger than the one of the timetable, with an elucidatory example. We then also present a mixed-integer programming formulation for IPESPA, that generalizes the so-called Q0-constraints of [11].

The setting above is for the general case, in which we allow any map of activities to sets of infrastructure elements. In a second step, we then restrict our inquiry to a more common and practical use-case, where multiple infrastructure elements can be considered as equivalent. We then assume that activities can only be assigned to one element, but this element is allowed to have a capacity larger than one. For example, the two sides of a platform are oftentimes equivalent options to choose, and we might consider such a platform as an element with capacity two. The main achievement is a mixed-integer programming formulation for this *Infrastructure-Aware* PESP *with Capacities* (IPESPC), a model much more compact than the IPESPA model, based on matchings in certain auxiliary bipartite graphs. Finally, this allows us also to derive two new alternative mixed-integer programming formulations for standard IPESP, i.e., IPESPC with unit capacities, beyond those of [2, 11].

We compare our new formulations on three realistic instances, both in the case of unit and larger capacities, and demonstrate their computational feasibility and practical benefit.

Section 2 recalls the Periodic Event Scheduling Problem and its infrastructure-aware extension IPESP. Section 3 introduces IPESPA, discusses the theory of general infrastructure assignments, and presents a mixed-integer programming formulation. In Section 4, we restrict to IPESPC and present a matching-based MIP model, along with the resulting new formulations for IPESP. We evaluate the computational power of our models in Section 5, before concluding the paper in Section 6.

The present work is a direct consequence of the fruitful connections and conversations that were had during ATMOS 2023 [4], and we thereby thank the organizing committee for fostering the transport optimization community.

2 Periodic Event Scheduling and Infrastructure Awareness

The Periodic Event Scheduling Problem (PESP) is the standard model to compute and optimize timetables for public transport. It is formulated as follows.

▶ **Definition 1** ([16]). *Consider a directed graph G with vertex set $V(G)$ and arc set $A(G)$, together with $T \in \mathbb{N}$, vectors $\ell, u \in \mathbb{R}^{A(G)}$, and $w \in \mathbb{R}_{\geq 0}^{A(G)}$. The* Periodic Event Scheduling Problem (PESP) *is to find vectors $\pi \in \mathbb{R}^{V(G)}$ and $x \in \mathbb{R}^{A(G)}$ such that*

a) $\pi_j - \pi_i \equiv x_a \bmod T$ *for all $a = (i,j) \in A(G)$,*

b) $\ell \leq x \leq u$,

c) $w^\top x$ *is minimum,*

or to decide that no such π and x exist.

In public transportation practice the directed graph G is oftentimes a so called *event-activity network*. There nodes $V(G)$ are the events, typically arrival or departure events, whereas arcs $A(G)$ are the activities, typically driving from a departure to an arrival, or dwelling from an arrival to a departure. The number $T \in \mathbb{N}$ is the *period time*, and determines after how long each event should repeat. Then the vectors π and x sought by PESP are called *periodic timetable* and *periodic tension*, respectively, where the former represents T-periodic timestamps denoting at which point of each period each event should occur, and the latter instead denotes the duration of the activities in-between events. PESP instances are commonly denoted as (G, T, ℓ, u, w).

Note that the simultaneous use of timetable and tension variables is primarily for ease of expression, since one can always be recovered from the other. In fact, given a periodic timetable π, a corresponding tension is quickly found by setting $x_a := [\pi_j - \pi_i - \ell_a]_T + \ell_a$ for every $a = (i,j) \in A(G)$, where $[\cdot]_T$ denotes the modulo T operator with values in $[0, T)$. Likewise, given a periodic tension, a corresponding timetable is quickly found by a connected graph traversal [7, Theorem 9.8].

For in-depth analysis of the many properties of PESP, we refer to the literature, starting with [8]. Multiple mixed-integer program formulations for PESP are known [7]. For simplicity, here we choose the *standard formulation* [16], which models PESP by linearizing the modulo constraints by use of auxiliary integer variables p_{ij}, called periodic offsets:

$$\min \sum_{(i,j)\in A(G)} w_{ij} x_{ij} \tag{1a}$$

$$\text{s.t.} \quad \pi_j - \pi_i + T p_{ij} = x_{ij} \quad \forall (i,j) \in A(G), \tag{1b}$$

$$0 \leq \pi_i < T \quad \forall i \in V(G), \tag{1c}$$

$$\ell_{ij} \leq x_{ij} \leq u_{ij} \quad \forall (i,j) \in A(G), \tag{1d}$$

$$p_{ij} \in \mathbb{Z} \quad \forall (i,j) \in A(G). \tag{1e}$$

Now we continue with the basic extension of PESP with infrastructure awareness, as per [2]. First of all we define an *infrastructure map* $\eta\colon \mathcal{A} \to E$, mapping certain arcs $\mathcal{A} \subseteq A(G)$ to a set of *infrastructure elements* E. This map encodes an assignment of activities to infrastructure, implying where those activities will physically take place within the network. We denote $\mathcal{A}_e := \eta^{-1}(e)$, for $e \in E$. For each infrastructure element we also have a minimum headway time $h \in \mathbb{R}^E_{\geq 0}$, indicating how long this element needs to be unoccupied between uses of different vehicles. As in [2], we assume that for each $e \in E$ either $h_e > 0$ or $\ell_a > 0$ for all $a \in \mathcal{A}_e$, to avoid pathological cases.

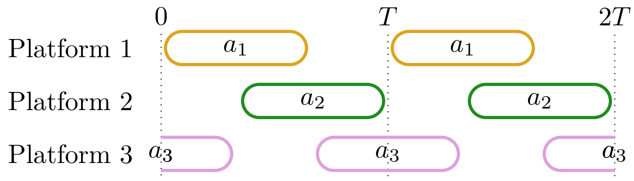
Considering then arcs $a_1 = (i_1, j_1)$ and $a_2 = (i_2, j_2)$ in $\mathcal{A}$, and such that $a_1 \neq a_2$ and $\eta(a_1) = \eta(a_2)$, we say that a_1 does not *h-overlap* a_2 if it holds that

$$[\pi_{i_2} - \pi_{i_1}]_T \geq x_{a_1} + h_e. \tag{2}$$

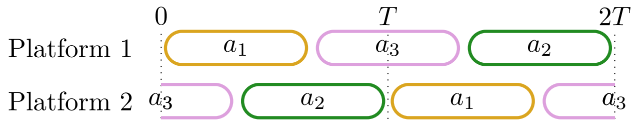
The constraint (2) is called Q0-constraint in [11], but there is also another possible equivalent formulation, namely the Q4-constraint ("butterfly constraint"). These entail, for each pair of arcs a_1 and a_2 as above, the addition of auxiliary arcs (j_1, i_2) and (j_2, i_1) with lower bound h_e and upper bound $T - h_e$, and then imposing total tension exactly equal to T along the 4-cycle $q(a_1, a_2) := (i_1, j_1, i_2, j_2, i_1)$. That is,

$$\sum_{a \in q(a_1,a_2)} x_a = T. \tag{3}$$

Then (3) holds for $q(a_1, a_2)$ if and only if a_1 and a_2 do not h-overlap each other [11].



(a) T-periodic platform assignment with 3 platforms.



(b) $2T$-periodic platform assignment with 2 platforms.

Figure 1 Two platform assignments of the same three activities.

We say that a set $S \subseteq \mathcal{A}$ is *h-conflict-free* if no arc in S h-overlaps another, and furthermore $x_a + h_e \leq T$ for all $e \in E$ and all $a \in S \cap \mathcal{A}_e$.

▶ **Definition 2** ([2]). *Let* (G, T, ℓ, u, w) *be a* PESP *instance, let* $\eta\colon \mathcal{A} \to E$ *be an infrastructure map, and let* $h \in \mathbb{R}^E_{\geq 0}$. *The* Infrastructure-Aware PESP (IPESP) *is to find a solution* (π, x) *to* PESP *on* (G, T, ℓ, u, w) *such that* $\mathcal{A}$ *is* h*-conflict-free and the solution is optimal, or to decide that no such solution exists.*

The PESP mixed-integer program (1), together with either all necessary Q0-constraints (2) or Q4-constraints (3), solves Infrastructure-Aware PESP.

This extended form of PESP implicitly assumes two rather strict properties. Firstly, the map η is, by definition, mapping each arc in $\mathcal{A}$ to a single $e \in E$, thereby presuming that such an infrastructure assignment has already been fixed. This implies that every activity must repeat every period always on the same infrastructure. However common, this need not be the case, and as we will see in the next section, it shall not.

3 General Infrastructure Awareness with Flexible Infrastructure Maps

Let us begin by considering the following illustrative example, motivating our work.

▶ **Example 3.** Consider an IPESP situation where we have $T = 30$ minutes, and three dwelling activities $a_k = (i_k, j_k)$ for $k \in \{1, 2, 3\}$. Suppose further that there are three platforms e_1, e_2, e_3 with $\eta(a_k) = e_k$ for all $k \in \{1, 2, 3\}$, without headway requirements. Let $\pi_{i_1} = 0$, $\pi_{i_2} = 10$, $\pi_{i_3} = 20$, and $\pi_{j_1} = 20$, $\pi_{j_2} = 0$, $\pi_{j_3} = 10$, meaning that each dwelling activity is scheduled for 20 min. In Figure 1a, we see how this would play out. Crucially, this leaves each platform unoccupied for 10 minutes per period of 30 minutes.

Suppose we were now to allow all activities to be assigned to either platform, and possibly to different platforms at different periodic repetitions. Then, the configuration of Figure 1b would be possible. With the same timetable as before, only two platforms are required now. This enables a more efficient use of the existing infrastructure.

Moreover, IPESP always assumes that no infrastructure element can be occupied for longer than T. This might however be practically necessary, e.g., due to regulations on minimum turnaround times.

3.1 Problem Definition and Periodizability of Assignments

We now present the tools to do timetabling while ensuring efficient use of the underlying infrastructure. Consider the flexible infrastructure map $\eta\colon \mathcal{A} \to \mathcal{E} \subseteq 2^E$, and so having the option to choose where to have activities occur. Given a periodic timetable π and tension x on a PESP instance (G, T, ℓ, u, w), we call

$$\mathcal{I} \coloneqq \left\{ I_a^{(k)} \;\middle|\; a \in \mathcal{A}, k \in \mathbb{Z} \right\} \tag{4}$$

the *realisation* of (π, x), where $I_a^{(k)} \coloneqq [\pi_i + kT, \pi_i + kT + x_{ij}]$ for $k \in \mathbb{Z}$ and $a = (i, j) \in \mathcal{A}$. Now, an *infrastructure assignment* is any map $\nu\colon \mathcal{I} \to E$, and we say it is *valid* if we have that $\nu\left(I_a^{(k)}\right) \in \eta(a)$ for all $I_a^{(k)} \in \mathcal{I}$. Furthermore, for a given infrastructure assignment ν and an infrastructure element $e \in E$ we define

$$\mathcal{I}_e \coloneqq \left\{ [\pi_i + kT, \pi_i + kT + x_{ij} + h_e) \;\middle|\; \forall I_a^{(k)} \in \mathcal{I}\colon \nu\left(I_a^{(k)}\right) = e \right\}, \tag{5}$$

and we say ν is *h-conflict-free* if the intervals in $\mathcal{I}_e$ are pairwise disjoint for every $e \in E$.

It now comes natural to formulate the following.

▶ **Definition 4.** *Let (G, T, ℓ, u, w) be a* PESP *instance and $\eta\colon \mathcal{A} \to \mathcal{E} \subseteq 2^E$ an infrastructure map, and let $h \in \mathbb{R}^E_{\geq 0}$. The* Infrastructure-Aware PESP with Assignment (IPESPA) *is to find a solution (π, x) to* PESP *on (G, T, ℓ, u, w), together with a valid and h-conflict-free infrastructure assignment ν, such that the solution is optimal, or to decide that no such solution exists.*

It is clear that were η not to be actually flexible, meaning $|\eta(a)| = 1$ for every $a \in \mathcal{A}$, then we would fall back into Definition 2 by fixing the only possible assignment $\nu(I_a^{(k)}) \coloneqq \eta(a)$ for all intervals in the realisation. Otherwise, this problem formulation allows for full flexibility in the choice of infrastructure, which can change after any periodic repetition, within the limits of η. This lack of structure and predictability may seem to be an issue of design, since the solutions could even become indescribable in finite terms. Thankfully, this will turn out not to be an issue. We say an infrastructure assignment ν is ω-periodic, for some $\omega \in \mathbb{N}$, if

$$\nu(I_a^{(k)}) = \nu(I_a^{(k)} + \omega T), \tag{6}$$

for every $I_a^{(k)} \in \mathcal{I}$. In such a case, we call ω the infrastructural period of the assignment. As it turns out, we are always able to restrict to such repeating patterns without losing any underlying PESP solution.

▶ **Theorem 5.** *Consider an instance (G, T, ℓ, u, w) of* IPESPA *with $\eta\colon \mathcal{A} \to \mathcal{E} \subseteq 2^E$, and a solution (π, x) together with a valid and h-conflict-free infrastructure assignment ν. Then, there exist $\omega \in \mathbb{N}$, with $\omega \leq |E|^{|E|}$, and a valid and h-conflict-free infrastructure assignment σ such that σ is ω-periodic.*

Proof. Let us consider $\mathcal{I}$ for the above instance and solution. All activities have a single representative interval in $\mathcal{I}$ whose lower bound is in $[pT, pT + T)$, with $p \in \mathbb{Z}$. This set of representatives, which we denote by $\mathcal{F}_p$, is finite. Even more so, considering for each $e \in E$ the one interval in $\mathcal{F}_p$ that is ν-assigned to e and with the minimum lower bound, there are at most $|E|$ such leading intervals. There are at most $|E|^{|E|}$ ways to assign these intervals, and so there exists a $p' \in \mathbb{Z}$ such that $p - |E|^{|E|} \leq p' \leq p + |E|^{|E|}$, and such that $\nu|_{\mathcal{F}_{p'}}$ mirrors $\nu|_{\mathcal{F}_p}$ on all the leading intervals of $\mathcal{F}_p$. By that, mirroring the whole of $\nu|_{\mathcal{F}_p}$ on $\mathcal{F}_{p'}$ can be done without h-conflict. Then, without loss of generality, we assume that $p < p'$, and set $\omega := p' - p$. By construction $\omega \leq |E|^{|E|}$, and we can construct a valid, h-conflict-free, and ω-periodic infrastructure assignment σ by setting

$$\sigma\left(I_a^{(k)}\right) := \nu\left(I_a^{([k-p]_\omega)}\right). \tag{7}$$

◀

The theorem ensures that IPESPA can be solved by restricting to periodic assignments, whose maximum period is bounded by the instance. Note that the bound on ω can be significantly improved if all headways are the same, i.e., $h_e = h_{e'}$ for any $e, e' \in E$. In that case, much along the lines of [6, Theorem 3.1], the bound becomes $\omega \leq |E|!$.

3.2 Pattern Functions and Conflict-Freeness

We will now construct a finitely described object from which a periodic infrastructure assignment can be derived, and conclude this chapter by showing how to use said object to formulate IPESPA as a mixed-integer program. As in Definition 4 we have a PESP instance, an infrastructure map, and a vector of headways. Choosing some maximum infrastructural period $M \in \mathbb{N}_{>0}$, we construct a *pattern function* $\mathcal{H}\colon \mathcal{A} \to \mathcal{P}$, that assigns to each arc in $\mathcal{A}$ a *pattern* in

$$\mathcal{P} := \bigcup_{i|M} E^i = \{(e_1, \dots, e_i) \mid e_1, \dots, e_i \in E \text{ and } i \text{ divides } M\}. \tag{8}$$

A pattern function is said to be *valid* if every image $\mathcal{H}(a)$ only contains elements of $\eta(a)$. The prescribed pattern is intended to be repeated ad infinitum. A corresponding infrastructure assignment $\nu_{\mathcal{H}}$ is quickly extracted from a pattern function $\mathcal{H}$, by setting

$$\nu_{\mathcal{H}}\left(I_a^{(k)}\right) := \mathcal{H}(a)_{[k]_{m_a}} \qquad \forall a \in \mathcal{A}, \forall k \in \mathbb{Z}, \tag{9}$$

where m_a is the length of $\mathcal{H}(a)$, i.e., the number of entries. So constructed, $\nu_{\mathcal{H}}$ is periodic, of period at most M. Note that the choice of M is up to the planner and the model becomes more flexible the more M is divisible.

▶ **Example 6.** Let us consider again Example 3 in the case where we allow to assign each activity to each platform, i.e., $\eta(a_k) = \{e_1, e_2\}$ for all $k \in \{1, 2, 3\}$. The valid pattern function associated to the infrastructure assignment in Figure 1b is given by $\mathcal{H}(a_1) = \mathcal{H}(a_3) = (e_1, e_2), \mathcal{H}(a_2) = (e_2, e_1)$, so that $m_{a_1} = m_{a_2} = m_{a_3} = 2$.

We can now use pattern functions to formulate linear modulo constraints, that generalize the Q0-constraints as presented in (2).

▶ **Theorem 7.** *Consider a* PESP *instance* (G, T, ℓ, u, w)*, an infrastructure map* $\eta\colon \mathcal{A} \to \mathcal{E} \subseteq 2^E$*, headways* $h \in \mathbb{R}^E_{\geq 0}$*, and some* PESP *solution* (π, x)*. Let* $\mathcal{H}$ *be a pattern function, and denote by* m_a *the length of the pattern* $\mathcal{H}(a)$*. Then, the infrastructure assignment* $\nu_{\mathcal{H}}$ *is* h*-conflict-free if and only if:*

(a) *For every arc* $a \in \mathcal{A}$ *and infrastructure element* $e \in \mathcal{H}(a)$*, we have*

$$x_a + h_e \leq m_a T. \tag{10}$$

(b) *For every $a_1, a_2 \in \mathcal{A}$, with $a_1 = (i_1, j_1)$ and $a_2 = (i_2, j_2)$, with images under $\mathcal{H}$ both containing the same infrastructure element e at indices p_1 and p_2 respectively, such that either $a_1 \neq a_2$ or $p_1 \neq p_2$, we have*

$$[\pi_{i_2} + (p_2 + k_2 m_{a_2}) T - \pi_{i_1} - (p_1 + k_1 m_{a_1}) T]_{mT} \geq x_{a_1} + h_e, \quad \forall k_1 \in \left\{0, \dots, \frac{m}{m_{a_1}} - 1\right\}, k_2 \in \left\{0, \dots, \frac{m}{m_{a_2}} - 1\right\}, \tag{11}$$

where $m \coloneqq \operatorname{lcm}(m_{a_1}, m_{a_2})$, and the indexing of the patterns starts at 0.

Proof. ($\implies$): If (a) is violated, then $\nu_{\mathcal{H}}\left(I_a^{(p)}\right) = e = \nu_{\mathcal{H}}\left(I_a^{(p+m_a)}\right)$, for p the index of e in $\mathcal{H}(a)$. Then we find in $\mathcal{I}_e$ the intervals

$$[\pi_i + pT, \pi_i + pT + x_{ij} + h_e) \text{ and } [\pi_i + (p + m_a)T, \pi_i + (p + m_a)T + x_{ij} + h_e), \tag{12}$$

which intersect, since $\pi_i + pT + x_{ij} + h_e > \pi_i + pT + m_a T$.

Suppose instead (a) holds, and (b) is violated, for some $k_1 \in \{0, \dots, {}^{m}/_{m_{a_1}} - 1\}$ and $k_2 \in \{0, \dots, {}^{m}/_{m_{a_2}} - 1\}$. By construction in (9), note that the images under $\nu_{\mathcal{H}}$ of $I_{a_1}^{(p_1 + k_1 m_{a_1})}$ and $I_{a_2}^{(p_2 + k_2 m_{a_2})}$ are both e, in fact for any integral k_1 and k_2. Since $\pi_v \in [0, T)$ for every $v \in V(G)$, we have that

$$[\pi_{i_2} + (p_2 + k_2 m_{a_2}) T - \pi_{i_1} - (p_1 + k_1 m_{a_1}) T]_{mT} < x_{a_1} + h_e, \tag{13}$$

and by construction the content of the modulo operator is either already in $[0, mT)$, or it is in $[-mT, 0)$. If it is non-negative, we find in $\mathcal{I}_e$ the intervals

$$[\pi_{i_1} + (p_1 + k_1 m_{a_1})T, \pi_{i_1} + (p_1 + k_1 m_{a_1})T + x_{a_1} + h_e) \text{ and } [\pi_{i_2} + (p_2 + k_2 m_{a_2})T, \pi_{i_2} + (p_2 + k_2 m_{a_2})T + x_{a_2} + h_e), \tag{14}$$

and they intersect. If instead the content is negative, then the modulo operator will add mT, and we find in $\mathcal{I}_e$ the intervals

$$[\pi_{i_1} + (p_1 + k_1 m_{a_1})T, \pi_{i_1} + (p_1 + k_1 m_{a_1})T + x_{a_1} + h_e) \text{ and } [\pi_{i_2} + (p_2 + k_2 m_{a_2} + m)T, \pi_{i_2} + (p_2 + k_2 m_{a_2} + m)T + x_{a_2} + h_e), \tag{15}$$

and they intersect.

($\impliedby$): Suppose now that $\nu_{\mathcal{H}}$ is not h-conflict-free. There is then an element $e \in E$ such that the set $\mathcal{I}_e$ contains two intersecting intervals. Let these be $I_{a_1}^{(s_1)}$ and $I_{a_2}^{(s_2)}$, and without loss of generality let us assume that $\min I_{a_2}^{(s_2)} \in I_{a_1}^{(s_1)}$. We then have that

$$0 \leq \pi_{i_2} + s_2 T - \pi_{i_1} - s_1 T < x_{a_1} - h_e \leq m_{a_1} T, \tag{16}$$

where the last inequality holds if (a) does. Then, since $m_{a_1} T \leq mT$, the $[\cdot]_{mT}$ operator is freely applied, and we have $[\pi_{i_2} + s_2 T - \pi_{i_1} - s_1 T]_{mT} < x_{a_1} + h_e$. For both intervals, another way to write s_i is as $p_i + k_i m_{a_i} + \beta_i m$, with integral β_i and minimal positive integral k_i, by which we have $[\pi_{i_2} + (p_2 + k_2 m_{a_2} + \beta_2 m)T - \pi_{i_1} - (p_1 + k_1 m_{a_1} + \beta_1 m)T]_{mT} < x_{a_1} + h_e$. Then, the two summands $\beta_2 mT$ and $\beta_1 mT$ can be deleted since they do not affect the modulo operation, and we have found a violation of (b). ◀

▶ Remark 8. In the IPESP case, i.e., $|\eta(a)| = 1$ for all $a \in \mathcal{A}$, all patterns $\mathcal{H}_a$ have length $m_a = 1$, so that $m = 1$. Theorem 7 then states that an infrastructure assignment is h-conflict-free if and only if $x_a + h_e \leq T$ for all $e \in E$ and $a \in \mathcal{A}_e$ and the Q0-constraint (2) holds for all pairs of distinct arcs assigned to the same infrastructure element. Indeed, this was our definition of h-conflict-freeness in Section 2. In general, the constraints (11) can be interpreted as Q0-constraints that implicitly capture a time expansion up to period mT.

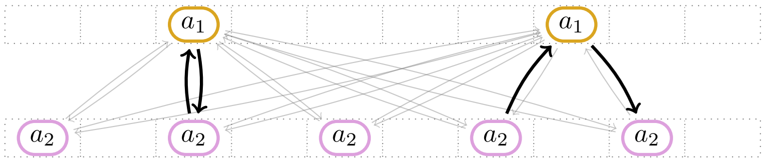


Figure 2 Two time expansions over 10 periods, with occurrences of two activities, a_1 and a_2, with different pattern lengths. Each arc symbolizes one constraint like (19). Only those highlighted in black are needed in case there were upper bounds $u_{a_1}, u_{a_2} \leq T$.

3.3 A MIP Formulation for IPESPA

We now want to use the platforming period bound of Theorem 5 and the inequalities of Theorem 7 to extend the PESP mixed-integer program (1) to IPESPA. We model a pattern function $\mathcal{H}$ by introducing binary variables $\tau_{a\rho}$ for all possible $a \in \mathcal{A}$ and images $\rho \in \mathcal{P} = \bigcup_{i|M} E^i$, whenever ρ is valid for a, i.e., all entries of ρ are in $\eta(a)$. By the bound expressed in Theorem 5 we can choose a finite but sufficiently large M, thereby having finitely many variables $\tau_{a\rho}$. Exactly one pattern has to be activated for each arc, which we express by

$$\sum_{\rho \in \mathcal{P}} \tau_{a\rho} = 1, \qquad \forall a \in \mathcal{A}. \tag{17}$$

We include the inequalities (10), but only activate them if relevant, by having

$$x_a + \max_{e \in \rho} h_e \leq m_a T + (1 - \tau_{a\rho}) B, \tag{18}$$

for every variable $\tau_{a\rho}$. Here $\max_{e\in\rho} h_e$ and m_a are scalars, determined by the pattern ρ, and $B := \max_{a\in\mathcal{A}} u_a + \max_{e\in E} h_e$ is a scalar globally determined by the instance itself. This way, if $\mathcal{H}(a) = \rho$, then (18) is effective, but otherwise it is trivially satisfied.

To linearize the modulo operation in (11) we also introduce binary variables $s_{\rho_1\rho_2}$. These are analogous to the periodic offsets p_{ij} (1e) and can indeed be restricted to $\{0, 1\}$, as in this case the modulo operator is applied to a number in $[-mT, mT)$. Activating the constraint if and only if both patterns ρ_1 and ρ_2 are selected, we have

$$\pi_{i_2} + (p_2 + k_2 m_{a_2})T - \pi_{i_1} - (p_1 + k_1 m_{a_1})T + mTs_{\rho_1\rho_2} \geq x_{a_1} + h_e - (2 - \tau_{a_1\rho_1} - \tau_{a_2\rho_2})B, \tag{19}$$

for every pair of arcs $a_1, a_2 \in \mathcal{A}$, respectively with $\mathcal{H}$-images ρ_1, ρ_2, of lengths m_{a_1}, m_{a_2}, both containing $e \in E$ at indices p_1, p_2 (0-indexed), for every $k_1 \in \{0, \ldots, {}^{m}/_{m_{a_1}} - 1\}$ and $k_2 \in \{0, \ldots, {}^{m}/_{m_{a_2}} - 1\}$, and where $m = \operatorname{lcm}(m_{a_1}, m_{a_2})$. Note that p_i, m_{a_i}, m, and h_e, here are all scalars, determined by ρ_1 and ρ_2.

▶ **Example 9.** For an illustration of which constraints (19) are applied, we refer to Figure 2. There we have two activities a_1 and a_2, assignable to the same infrastructure element, with $p_1 = 2$, $m_1 = 5$, and $p_2 = 0$, $m_2 = 2$. Each arc symbolizes one constraint like (19), of which there is two per pairing, i.e., 20 in total. However, in many cases, if $p_2 + k_2 m_{a_2} - p_1 - k_1 m_{a_1} > \lceil (u_{a_1} + h_e)/T \rceil$, then (11) is always satisfied, and we can exclude it a priori. This means that, depending on the PESP upper bounds, a significant drop in the number of constraints is possible. The bold arcs in Figure 2 are the 4 constraints that would be kept if we had, for instance, $u_{a_1}, u_{a_2} \leq T$.

Including in a PESP mixed-integer program such as (1) these two types of binary variables $\tau_{a\rho}$ and $s_{\rho_1\rho_2}$, together with (17), (18), and (19), yields a mixed-integer program formulation for IPESPA. In (20) we can see it in full.

$$\min \sum_{a\in A(G)} w_a x_a \tag{20a}$$

$$\text{s.t.} \quad (\pi, x) \text{ solves PESP on } G, \tag{20b}$$

$$\sum_{\rho\in\mathcal{P}} \tau_{a\rho} = 1 \qquad \forall a \in \mathcal{A}, \tag{20c}$$

$$x_a + \max_{e\in\rho} h_e \le m_a T + (1 - \tau_{a\rho})B \qquad \begin{array}{l}\forall a \in \mathcal{A} \text{ and}\\ \forall \text{ valid } \rho \in \mathcal{P},\end{array} \tag{20d}$$

$$\begin{pmatrix} \pi_{i_2} + (p_2 + k_2 m_{a_2})T \\ -\pi_{i_1} - (p_1 + k_1 m_{a_1})T \\ +mTs_{\rho_1\rho_2} \end{pmatrix} \ge \begin{pmatrix} x_{a_1} + h_e \\ -(2-\tau_{a_1\rho_1} - \tau_{a_2\rho_2})B \end{pmatrix} \qquad \forall a_1, a_2 \text{ s.t. } (\star), \tag{20e}$$

$$\tau_{a\rho} \in \{0, 1\} \qquad \begin{array}{l}\forall a \in \mathcal{A} \text{ and}\\ \forall \text{ valid } \rho \in \mathcal{P},\end{array} \tag{20f}$$

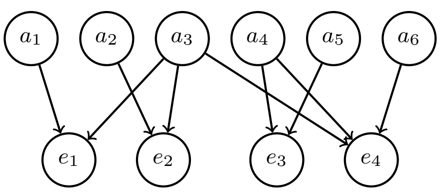
$$s_{\rho_1\rho_2} \in \{0, 1\} \qquad \forall \rho_1, \rho_2 \in \mathcal{P}, \tag{20g}$$

where by $(\star)$ we mean the conditions of (19).

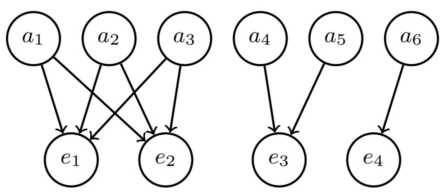
This formulation allows for a great degree of flexibility, giving practitioners a direct handle on the maximum infrastructural period M. In fact, although fixing M to the bound proven in Theorem 5 ensures that no PESP solution is excluded, it is entirely possible that in a practical setting one would want to limit it further, so as to bound the complexity of the infrastructural assignment. To that same purpose, the formulation also allows to forcibly forbid individual patterns if desired.

4 Partitionable Infrastructure Maps

In practice, the infrastructure map $\eta\colon \mathcal{A} \to \mathcal{E}$ is oftentimes not completely flexible, as shown in Figure 3a, but instead comes with additional structure. Of particular interest is the case illustrated in Figure 3b, where $\mathcal{E}$ is a partition of all infrastructure elements, i.e., the infrastructure elements are all grouped and every activity can be freely assigned to all elements in one of such groups. An omnipresent example is a station with two platforms that serves lines in two directions, with the lines in each direction having a dedicated platform. It turns out that when η is *partitionable*, the IPESPA boils down to a rather compact form.



(a) Non-partitionable η.



(b) Partitionable η.

Figure 3 Two infrastructure maps for an instance with six activities and four infrastructure elements.

Formally, we define the following variant of the IPESPA.

▶ **Definition 10.** *Let* (G, T, ℓ, u, w) *be a* PESP *instance and* $\eta\colon \mathcal{A} \to \mathcal{E}$ *an infrastructure map, where* $\mathcal{E}$ *is a partition of* E*. The* Infrastructure-Aware PESP with Capacities (IPESPC) *is to find a solution* (π, x) *to* PESP *on* (G, T, ℓ, u, w)*, together with a valid and conflict-free platform assignment* ν*, such that the solution is optimal, or to decide that no such solution exists.*

Alternatively, IPESPC is equivalent to IPESP with the additional feature that every $e \in E$ has now a capacity $k_e \in \mathbb{N}$. In other words, $e \in E$ no longer corresponds to a single infrastructure element, but to a group of k_e equivalent elements. For ease of exposition, we stick to this perspective going forward.

We use a matching-based approach to solve IPESPC. To this end, we expand G with auxiliary arcs between activities that can use the same group of infrastructure elements. Formally, let G' be the graph arising from G by adding for each $e \in E$ and $a_1 = (i_1, j_1), a_2 = (i_2, j_2) \in \mathcal{A}_e$ a new arc α from j_1 to i_2. We refer to α as the auxiliary arc from a_1 to a_2, and much like the headway arcs a^{I} used in [2] and in the Q4 butterfly constraints (3), we set $\ell_\alpha := h_e$, $u_\alpha := T - h_e$, $w_\alpha := 0$ if $a_1 \neq a_2$, and $\ell_\alpha := h_e, u_\alpha := T, w_\alpha := 0$ in the case $a_1 = a_2$. Let $\mathcal{A}'$ denote the set of all auxiliary arcs, let $\mathcal{A}'_e$ denote all auxiliary arcs associated to e, and let G'_e be the subgraph of G' on the arcs in $\mathcal{A}'_e$. For $S \subseteq V(G) \times V(G)$, we will use the notation $G[S]$ for the graph $(V(G), A(G) \cup S)$, so that, e.g., $G' = G[\mathcal{A}']$. The following theorem compactly characterizes when a PESP solution admits a feasible infrastructure assignment in the context of IPESPC:

▶ **Theorem 11.** *Consider a* PESP *instance* (G, T, ℓ, u, w)*, the expanded graph* G'*, a partitionable infrastructure map* $\eta\colon \mathcal{A} \to \mathcal{E}$*, headways* $h \in \mathbb{R}^E_{\geq 0}$*, capacities* $k \in \mathbb{N}^E$*, and some* PESP *solution* (π, x) *on* G*. Then, there exists a valid and* h*-conflict-free infrastructure assignment* ν *if and only if for each* $e \in E$ *there exists a perfect matching* $\mathcal{M}'_e \subseteq \mathcal{A}'_e$ *of* G'_e *and* (π, x) *can be extended to a* PESP *solution on* $G[\mathcal{M}'_e]$ *such that*

$$\sum_{a \in \mathcal{A}_e} x_a + \sum_{a \in \mathcal{M}'_e} x_a \leq k_e T. \tag{21}$$

Proof. First, assume that there exist perfect matchings $\mathcal{M}'_e$ of G'_e satisfying (21) for all $e \in E$. The matching $\mathcal{M}'_e$ together with the arcs $\mathcal{A}_e$ forms a set of disjoint directed cycles, where every cycle consists of arcs that alternatingly belong to $\mathcal{A}_e$ and $\mathcal{A}'_e$.

Let then $C^e_1, C^e_2, \ldots, C^e_{m_e}$ denote the cycles corresponding to $e \in E$, and define $p^e_j := \frac{1}{T} \sum_{a \in C^e_j} x_a$ for $j = 1, \ldots, m_e$. Because the timetable (π, x) is feasible on $G[\mathcal{M}'_e]$, p^e_j is integer by the cycle periodicity property [7, Lemma 6.39]. Since it holds that

$$\sum_{j=1}^{m_e} p^e_j = \frac{1}{T} \sum_{j=1}^{m_e} \sum_{a \in C^e_j} x_a = \frac{1}{T} \left(\sum_{a \in \mathcal{A}_e} x_a + \sum_{a \in \mathcal{M}'_e} x_a \right) \leq k_e, \tag{22}$$

it suffices to show that just the activities appearing in C^e_j can be assigned on a group of capacity p^e_j. The total tension along the cycle is $p^e_j T$, so in a $p^e_j T$-periodic schedule it is straightforward to fit all the activities in $C^e_j \cap \mathcal{A}$, simply following the order in which they appear through the cycle and with timestamps agreeing with π modulo T. Then, having an available capacity of p^e_j, that schedule can be repeated over each of the p^e_j equivalent elements, each time shifted forward by T. This construction implies adherence to the T-periodic timetable π, and a $p^e_j T$-periodic infrastructure assignment, valid since each $\eta(C^e_j \cap \mathcal{A}) = e$, and h-conflict-free by the bounds on the auxiliary arcs.

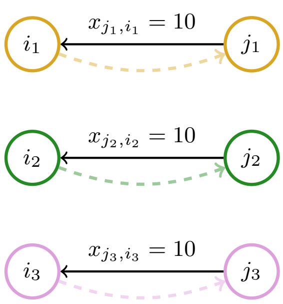
Suppose the contrary instead, that no perfect matching in $\mathcal{A}'_e$ respects (21). Notice how, given that π is fixed and the headway h_e is the same on all elements in the group e, we can without loss of generality lengthen all activities in $\mathcal{A}_e$ by h_e and then assume that the new minimum headway is 0 instead. Now, let $\mathcal{M}^*$ be a minimum tension perfect matching in G'_e. We then have that for some $e \in E$ there is an integer $K_e > k_e$ such that

$$\sum_{a \in \mathcal{A}_e} x_a + \sum_{a \in \mathcal{M}^*} x_a = K_e T > k_e T. \tag{23}$$

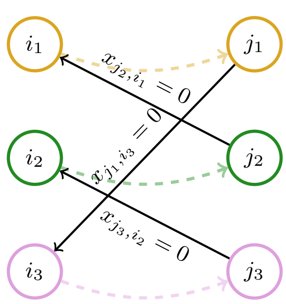
Now, for $t \in [0, T)$ and $S \subseteq \mathcal{A}_e \cup \mathcal{A}'_e$, let the *inventory function* $I_e(t, S)$ denote the number of h-overlapping activities in S at time t. By (23) we have that $I_e(t, \mathcal{A}_e \cup \mathcal{M}^*) = K_e$ for all $t \in [0, T)$. Moreover, since $\mathcal{M}^*$ is a minimum tension perfect matching, then there is a $t^* \in [0, T)$ such that $I_e(t^*, \mathcal{M}^*) = 0$. That is because if $\mathcal{M}^* = \{(e_1, f_1), \ldots, (e_m, f_m)\}$ was h-overlapping everywhere instead, then without loss of generality we can assume that the timestamps would be

$$\pi_{e_1} < \pi_{f_m} < \pi_{e_2} < \pi_{f_1} < \ldots < \pi_{e_m} < \pi_{f_{m-1}}, \tag{24}$$

where a shorter matching is immediately apparent, negating the minimality of $\mathcal{M}^*$. See [17, Lemma 3] for further details. By the above, we then have that $I_e(t^*, \mathcal{A}_e) = I_e(t^*, \mathcal{A}_e \cup \mathcal{M}^*) - I_e(t^*, \mathcal{M}^*) = K_e > k_e$, implying there are more simultaneous activities than there is infrastructure capacity to host them. In particular, there is no h-conflict-free infrastructure assignment. ◀



(a) Matching corresponding to Figure 1a.



(b) Matching corresponding to Figure 1b.

Figure 4 The two matchings corresponding to Example 3.

▶ **Example 12.** We illustrate on the basis of Example 3 how matchings relate to infrastructure assignments in Figure 4. In the IPESPA situation of Figure 1a, we have three infrastructure elements e_1, e_2, e_3 of capacity one each. For each element e_k, the matching in Figure 4a creates a directed cycle of tension $1 \cdot T = 30$ containing the dwelling activity $a_k = (i_k, j_k)$ and the auxiliary arc (j_k, i_k). Figure 1b is an IPESPC situation with one infrastructure element e. Here, the matching in Figure 4b induces a directed cycle of tension $2 \cdot T = 60$, we hence use a capacity of two. The cycle encodes that each platform is used by a_1, a_3, a_2 in this cyclic order.

Theorem 11 allows formulating IPESPC very compactly compared to IPESPA. Introducing matching variables as binary decision variables y_a for all $a \in \mathcal{A}'$, we have the following MIP:

$$\min \sum_{a \in A(G)} w_a x_a \tag{25a}$$

$$\text{s.t.} \quad (\pi, x) \text{ solves PESP on } G, \tag{25b}$$

$$\sum_{a \in \delta^-_{G'_e}(i)} y_a = 1 \qquad \forall e \in E, \forall (i,j) \in \mathcal{A}_e, \tag{25c}$$

$$\sum_{a \in \delta^+_{G'_e}(j)} y_a = 1 \qquad \forall e \in E, \forall (i,j) \in \mathcal{A}_e, \tag{25d}$$

$$\sum_{a \in \mathcal{A}_e} x_a + \sum_{a \in \mathcal{A}'_e} x_a y_a \leq k_e T \qquad \forall e \in E, \tag{25e}$$

$$\pi_j - \pi_i + T p_a = x_a \qquad \forall a = (i,j) \in \mathcal{A}', \tag{25f}$$

$$\ell_a y_a \leq x_a \leq u_a y_a + (T-1)(1-y_a) \qquad \forall a \in \mathcal{A}', \tag{25g}$$

$$y_a \in \{0,1\} \qquad \forall a \in \mathcal{A}'. \tag{25h}$$

Constraints (25c) and (25d) define a unique predecessor and successor for each activity by requiring that y corresponds to a perfect matching $\mathcal{M}'_e$ in $\mathcal{A}'_e$ for each $e \in E$. Constraints (25e) ensure that the total time of the activities scheduled on a platform group and the selected auxiliary activities is at most the total available time on that platform group. These constraints can be linearized by introducing for each $a \in \mathcal{A}'$ a real variable z_a, with bounds $0 \leq z_a \leq u_a$, and constrained as $x_a - (1-y_a)u_a \leq z_a \leq x_a$, so that it is equal to the product $x_a y_a$. The constraints (25f) tie the tensions x_a on the auxiliary activities $a \in \mathcal{A}'$ to the timetable π. Finally, (25g) ensures that those tensions adhere to their bounds when they are part of the selected matching, and impose no restrictions otherwise.

There is a clear resemblance between (25) and the formulation [17] proposes for jointly optimizing a periodic timetable and vehicle circulation. This is no coincidence: as long as the corresponding mapping from activities to resources is a partition, any resource schedule associated to a periodic timetable can be described by a matching. In [17] the resources are vehicles, whereas in the present paper infrastructure elements, e.g., platforms. It immediately follows that the results established in [17] carry over to our setting. Most notably, given a feasible timetable, a greedy algorithm can actually be used to find an infrastructure assignment with the minimum number of required infrastructure elements.

For groups consisting of a single element, i.e., $k_e = 1$, the matching formulation can be enhanced using the surprising insight that in this case it is not necessary to compute the matching explicitly. We have the following theorem:

▶ **Theorem 13.** *Suppose $k_e = 1$ and let (π, x) be a* PESP *solution on G. Then $\mathcal{A}_e$ is h-conflict-free if and only if (π, x) extends to a* PESP *solution on G'_e such that*

$$\sum_{a \in \mathcal{A}_e} x_a + \left(\frac{1}{|\mathcal{A}_e|} \sum_{a \in \mathcal{A}'_e} x_a \right) = \frac{|\mathcal{A}_e| + 1}{2} T. \tag{26}$$

Proof. Suppose that $\mathcal{A}_e$ is h-conflict-free. This means that for any $a_s = (i_s, j_s), a_t = (i_t, j_t) \in \mathcal{A}_e$ with $a_s \neq a_t$ the Q4-constraints (3) must hold, namely

$$x_{i_s,j_s} + x'_{j_s,i_t} + x_{i_t,j_t} + x'_{j_t,i_s} = T, \tag{27}$$

where we use the notation $x'_a := [\pi_j - \pi_i - \ell_a]_t + \ell_a$ to indicate tensions on an auxiliary arc $a = (i,j)$.

Let $q(a_s, a_t)$ denote the butterfly-shaped 4-cycle given by $(i_s, j_s, i_t, j_t, i_s)$. There are $\sum_{i=1}^{|\mathcal{A}_e|-1} i = |\mathcal{A}_e|(|\mathcal{A}_e|-1)/2$ many such butterfly cycles, each auxiliary arc $(j_s, i_t) \in \mathcal{A}'_e$ with $s \neq t$ is in exactly one such cycle, while each arc $(i_s, j_s) \in \mathcal{A}_e$ is in $|\mathcal{A}_e| - 1$ many of them. Moreover, there are exactly $|\mathcal{A}_e|$ auxiliary arcs of the form (j_s, i_s), for which holds that

$$h_e \leq x_{i_s,j_s} + x'_{j_s,i_s} \leq 2T - h_e, \tag{28}$$

since $h_e \leq x'_{j_s,i_s} \leq T$ and h-conflict freeness implies $x_{i_s,j_s} \leq T - h_e$. Due to the cycle periodicity property,

$$x_{i_s,j_s} + x'_{j_s,i_s} = T, \tag{29}$$

unless $h_e = 0$, but then we can subtract T from $x'_{j_s,i_s} = T$ and maintain the feasibility of (π, x) on G'_e.

Summing up over all butterfly constraints (27) for each pair of activities and all the self-cycles (29) of each activity in $\mathcal{A}_e$, we obtain

$$|\mathcal{A}_e| \sum_{s=1}^{|\mathcal{A}_e|} x_{i_s,j_s} + \sum_{a \in \mathcal{A}'_e} x'_a = \left(\frac{|\mathcal{A}_e|(|\mathcal{A}_e|-1)}{2} + |\mathcal{A}_e| \right) T = \frac{|\mathcal{A}_e|(|\mathcal{A}_e|+1)}{2} T. \tag{30}$$

For the other direction, suppose that $\mathcal{A}'_e$ is not h-conflict free, but (π, x) extends to PESP solution on G'_e. Then one of (27) or (29) must be violated. Let $q(a_s, a_t)$ be a butterfly cycle with $a_s = (i_s, j_s)$ and $a_t = (i_t, j_t)$. Then

$$x_{i_s,j_s} + x'_{j_s,i_t} + x_{i_t,j_t} + x'_{j_t,i_s} \geq \ell_{i_s,j_s} + \ell_{i_t,j_t} + 2h_e > 0 \tag{31}$$

since we assumed that at least lower bounds or minimum headway times are positive. Due to the cycle periodicity property of periodic timetables,

$$x_{i_s,j_s} + x'_{j_s,i_t} + x_{i_t,j_t} + x'_{j_t,i_s} \geq T. \tag{32}$$

Moreover, considering the cycles comprised of $(i_s, j_s) \in \mathcal{A}_e$ and the auxiliary arc $(j_s, i_s) \in \mathcal{A}'_e$, we have

$$x_{i_s,j_s} + x'_{j_s,i_s} \geq \ell_{i_s,j_s} + h_e > 0, \tag{33}$$

so that

$$x_{i_s,j_s} + x'_{j_s,i_s} \geq T. \tag{34}$$

Therefore, if one of (27) or (29) is violated, we must have a strict inequality in (32) or (34). Taking the sum,

$$|\mathcal{A}_e| \sum_{s=1}^{|\mathcal{A}_e|} x_{i_s,j_s} + \sum_{a \in \mathcal{A}'_e} x'_a > \left(\frac{|\mathcal{A}_e|(|\mathcal{A}_e|-1)}{2} + |\mathcal{A}_e| \right) T = \frac{|\mathcal{A}_e|(|\mathcal{A}_e|+1)}{2} T, \tag{35}$$

so (26) cannot hold. ◀

A direct implication of Theorem 13 is a new formulation for IPESP, provided that all infrastructure elements have unit capacities:

$$\min \sum_{a \in A(G)} w_a x_a \tag{36a}$$

$$\text{s.t.} \quad (\pi, x) \text{ solves PESP} \quad \text{on } G', \tag{36b}$$

$$\sum_{a \in \mathcal{A}_e} x_a + \frac{1}{|\mathcal{A}_e|} \sum_{a \in \mathcal{A}'_e} x_a = \frac{|\mathcal{A}_e|+1}{2} T \quad \forall e \in E. \tag{36c}$$

In contrast to the original formulation proposed in [2] and to the matching approach (25), formulation (36) introduces neither additional integer variables nor quadratic terms.

5 Experiments

In the sequel, we will evaluate different formulations for IPESP and IPESPC on a set of realistic instances. We omit the IPESPA model, as for our datasets there hardly is added value compared to IPESPC. We use Gurobi 11 as a MIP solver on an Intel Xeon E3-1270 3.80 GHz CPU with 32 GB RAM.

5.1 Instances

We evaluate our models on instances we constructed based on publicly available timetable information, platform usage, and track data. Additionally, some track information was provided to us by DB InfraGO AG. The instances are:

- `S-Bahn`, the full network of S-Bahn Berlin, a suburban commuter rail network with 16 lines. On several sections, there are as much as 7 trains per track and direction within the period time of 20 minutes. Our IPESP instance is based on the annual timetable, assuming fixed driving times, but flexible dwelling and turnaround times. However, there are several places in the network where multiple platforms are available, and this builds our corresponding IPESPC instance.
- `Tram`, the full tram network of Berlin, comprising 22 lines operated with a period time of 20 minutes, with frequencies ranging between 1 and 6. The difficulty here does not lie in associating driving and dwelling times, which are fixed, but in fulfilling synchronization constraints and deciding infrastructure assignments at the terminal stations: The turnaround times are flexible and capacities in the turning loops are scarce. This is inherently an IPESPC instance, that, in fact, cannot be transformed into a feasible IPESP instance, since all T-periodic assignments are infeasible.
- `Corridor`, the central longitudinal railway corridor of regional and long-distance trains in Berlin, as well as a subsection of it, from Ostkreuz to Friedrichstraße, which we denote as `ShortCorridor`. Many stations have multiple platforms, and the trains have different stopping patterns. The period time is 60 minutes, driving, dwelling, and turnaround times are all variable. This, too, is inherently an IPESPC instance, from which we created an IPESP instance based on the annual timetable.

We use only a simple objective function for the timetabling part: Driving and dwelling activities are weighted by 2, turnarounds by 1, and all other arcs by 0. Additionally, note that we do not include any transfer arcs, in part because we have no data available regarding the flow of passengers, and in part because the scope of this work rather focuses on operational capabilities instead.

5.2 IPESP Experiments

For the unit capacity case of IPESP, we have now several formulations at hand: The Q4 butterfly constraints (3), the matching model (25), and the special formulation (36). We test these formulations and their combinations on the `S-Bahn` and `Corridor` instance, with a wall time limit of one hour. We further include versions where the matching variables y are relaxed to be continuous. Our results are collected in Table 1.

On `S-Bahn`, not all formulations found a solution, but those that did also managed to prove optimality within the time limit. All such formulations contain the Q4 butterfly constraints (3), and none of them use binary matching variables. The combination of Q4 with the special sum constraint (36) worked best, followed by pure Q4. When initialized with the optimal solution as MIP start, almost all formulations proved its optimality within the time frame of one hour, or managed a very thin optimality gap.

On `Corridor`, no formulation found solutions, and no attempt at providing initial partial solutions was successful. On `ShortCorridor` all formulations quickly found a primal solution within seconds, and an optimal solution within minutes, but none managed to prove that optimality within one hour. Only when starting `ShortCorridor` with the best solution found so far, a proof to optimality was reached, and only by the formulation using (25), i.e., matching with binary variables, together with the special constraints (36c). This was also the fastest formulation to reach optimality to begin with.

Table 1 Timed results for IPESP tests on `S-Bahn` and `Corridor`, expressed in seconds. Tests denoted with Q4 use Q4-constraints as in (3). Tests denoted with M use matching constraints as in (25), with capacity set to 1. Tests denoted with Mc use the same constraints as M, but with relaxed continuous variables instead. Tests denoted with S use constraints as in (36). The first column details the time to the first primal solution that was found, the second column the time to the optimal objective value (3058 for `S-Bahn` and 10 for `Corridor`), and the third column the time to fully close the optimality gap. The last column details the time needed to prove optimality when given the optimal solution from the beginning. The time limit was 1 hour per configuration.

`S-Bahn`	s to primal	s to optimal	s to proof	s to proof (warm)
Q4	64	227	227	95
Q4+M	–	–	–	1221
Q4+Mc	338	1672	1672	140
Q4+S	40	156	156	150
Q4+M+S	–	–	–	– (.58%)
Q4+Mc+S	172	738	738	766
M	–	–	–	– (.61%)
S	–	–	–	595
M+S	–	–	–	2952
Mc+S	–	–	–	1402
`ShortCorridor`	s to primal	s to optimal	s to proof	s to proof (warm)
Q4	0	110	–	–
Q4+M	5	87	–	–
Q4+Mc	0	82	–	–
Q4+S	2	48	–	–
Q4+M+S	1	15	–	–
Q4+Mc+S	0	23	–	–
M	0	44	–	–
S	1	576	–	–
M+S	4	11	–	3268
Mc+S	6	50	–	–

5.3 IPESPC Experiments

For the instances with capacities larger than one we used formulation (25). To aid the solution process, all infrastructure that still had capacity one has been modelled using Q4 constraints, and used the matching variables where necessary, i.e., for all larger infrastructure. We tested this formulation on the `S-Bahn`, the `Tram`, and the `Corridor` instance, with a time limit of four hours.

On `S-Bahn` no primal solution was found, but we were able to warm start the models with solutions found in the corresponding IPESP tests instead. In that case, the more flexible IPESPC within seconds improved the optimal IPESP solution (albeit by a measly 1.1%), and within approximately 4 more minutes reached the final primal value. In the course of the first hour, Gurobi managed to reduce the optimality gap to 0.2%, where it remained until the time limit.

On `Tram` proven optimality was reached within 10 seconds. Notably, however quick to solve this instance was, it is infeasible to formulate with simple IPESP. Only using the higher capacities enabled by IPESPC it was at all possible to generate a feasible solution.

On `Corridor`, again, no solution was found, but providing a partial starting solution on just the section of `ShortCorridor` was enough to be completed to a full solution for the whole network, which then was improved to proven optimality in only 2 minutes and 25 seconds. On `ShortCorridor` the first primal was found in under 10 minutes and was directly proven to be optimal. Notably, its objective value was better than the IPESP case, now reaching 0 slack. Starting the same test with a solution from the IPESP case reached proven optimality in 3 seconds.

6 Conclusion

This work extends the Infrastructure-Aware PESP (IPESP) framework. One of our new problem formulations, Infrastructure-Aware PESP with Assignment (IPESPA), does so by integrating the choice of the infrastructure assignment within IPESP, allowing for more flexible use of the available infrastructure. In fact, this flexibility can lead to higher efficiency, as well as improved timetables. Although extremely general in its assumptions, we prove that IPESPA can be formulated as a mixed integer linear program (20).

Moving on to a more restricted, but highly realistic scenario, we consider the case when infrastructure elements can be effectively considered as equivalent, and formulate this special version of IPESPA as well, namely Infrastructure-Aware PESP with Capacities (IPESPC). In this case the assignment structure is of note, since it can be seen as a matching problem on a complete bipartite graph, connecting the ends of activities to the starts of the next ones. This gives us not only a compact mixed integer linear program formulation (25), but also novel formulations for IPESP itself, seen as a case of IPESPC with only unit capacities.

Finally, on the practical side, we tested the new matching-based IPESP formulations, as well as the IPESPC formulation. On the unit capacity side, our tests went through various combinations of approaches, and although caution is advised when drawing conclusions, it seems that on the `S-Bahn` instance the Q4-based formulations fared better, whereas on `ShortCorridor`, which has a higher density of larger infrastructure elements to deal with, matching-based formulations had more success. With instances of larger capacity, instead, our tests show that our modelling approach can be of interest in real-world scenarios.

For future work, on the theoretical side we suggest proving tighter bounds on the maximum platforming period, as well as trying to generalize the Q4-constraints much like we here generalized the Q0-constraints. On the practical side, we suggest an iterative approach that uses a separate matching solver to concurrently feed the main model with partial solutions, and to develop heuristic approaches to quickly generate initial solutions.

References

1 E. Bortoletto, N. Lindner, and B. Masing. Tropical Neighbourhood Search: A New Heuristic for Periodic Timetabling. In M. D'Emidio and N. Lindner, editors, *22nd Symposium on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2022)*, volume 106 of *Open Access Series in Informatics (OASIcs)*, pages 3:1–3:19, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/OASIcs.ATMOS.2022.3.

2 E. Bortoletto, N. Lindner, and B. Masing. Periodic Timetabling with Cyclic Order Constraints. In D. Frigioni and P. Schiewe, editors, *23rd Symposium on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2023)*, volume 115 of *Open Access Series in Informatics (OASIcs)*, pages 7:1–7:18, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/OASIcs.ATMOS.2023.7.

3 G. Caimi, L. Kroon, and C. Liebchen. Models for railway timetable optimization: Applicability and applications in practice. *Journal of Rail Transport Planning & Management*, 6(4):285–312, 2017. doi:10.1016/j.jrtpm.2016.11.002.

4 D. Frigioni and P. Schiewe, editors. *OASIcs, Volume 115, ATMOS 2023, Complete Volume*, volume 115 of *Open Access Series in Informatics (OASIcs)*, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/OASIcs.ATMOS.2023.

5 F. Fuchs, A. Trivella, and F. Corman. Enhancing the interaction of railway timetabling and line planning with infrastructure awareness. *Transportation Research Part C: Emerging Technologies*, 142:103805, September 2022. doi:10.1016/j.trc.2022.103805.

6 J. Korst, E. Aarts, J. K. Lenstra, and J. Wessels. Periodic multiprocessor scheduling. In E. H. L. Aarts, J. van Leeuwen, and M. Rem, editors, *Parle '91 Parallel Architectures and Languages Europe*, pages 166–178, Berlin, Heidelberg, 1991. Springer Berlin Heidelberg. doi:10.1007/978-3-662-25209-3_12.

7 C. Liebchen. Periodic timetable optimization in public transport. In K.-H. Waldmann and U. M. Stocker, editors, *Operations Research Proceedings 2006*, pages 29–36, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg. doi:10.1007/978-3-540-69995-8_5.

8 C. Liebchen and R. H. Möhring. The Modeling Power of the Periodic Event Scheduling Problem: Railway Timetables — and Beyond. In F. Geraets, L. Kroon, A. Schoebel, D. Wagner, and C. D. Zaroliagis, editors, *Algorithmic Methods for Railway Optimization*, Lecture Notes in Computer Science, pages 3–40, Berlin, Heidelberg, 2007. Springer. doi:10.1007/978-3-540-74247-0_1.

9 N. Lindner and B. Masing. On the Split Closure of the Periodic Timetabling Polytope, 2023. doi:10.48550/arXiv.2306.02746.

10 B. Masing, N. Lindner, and C. Liebchen. Integrating Line Planning for Construction Sites into Periodic Timetabling via Track Choice. In D. Frigioni and P. Schiewe, editors, *23rd Symposium on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2023)*, volume 115 of *Open Access Series in Informatics (OASIcs)*, pages 5:1–5:15, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/OASIcs.ATMOS.2023.5.

11 B. Masing, N. Lindner, and C. Liebchen. Periodic timetabling with integrated track choice for railway construction sites. *Journal of Rail Transport Planning & Management*, 28:100416, 2023. doi:10.1016/j.jrtpm.2023.100416.

12 K. Nachtigall. *Periodic Network Optimization and Fixed Interval Timetables*. Habilitation Thesis, Universität Hildesheim, 1998.

13 K. Nachtigall and J. Opitz. Solving Periodic Timetable Optimisation Problems by Modulo Simplex Calculations. In M. Fischetti and P. Widmayer, editors, *8th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems (ATMOS'08)*, volume 9 of *OpenAccess Series in Informatics (OASIcs)*, Dagstuhl, Germany, 2008. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/OASIcs.ATMOS.2008.1588.

14 A. Hickford S. Blainey and J. Preston. Barriers to passenger rail use: A review of the evidence. *Transport Reviews*, 32(6):675–696, 2012. doi:10.1080/01441647.2012.743489.

15 P. Schiewe and A. Schöbel. Periodic Timetabling with Integrated Routing: Toward Applicable Approaches. *Transportation Science*, 54(6):1714–1731, 2020. Publisher: INFORMS. `doi:10.1287/trsc.2019.0965`.

16 P. Serafini and W. Ukovich. A mathematical model for periodic scheduling problems. *SIAM J. on Discret. Math.*, 2(4):550–581, 1989. `doi:10.1137/0402049`.

17 R. N. Van Lieshout. Integrated periodic timetabling and vehicle circulation scheduling. *Transportation Science*, 55(3):768–790, 2021. `doi:10.1287/trsc.2020.1024`.