The flexible break assignment problem for large tour scheduling problems with an application to airport ground handlers

Ferdinand Kiermaier[1] · Markus Frey[1] · Jonathan F. Bard[2]

Abstract

The paper examines the complexity of assigning multiple breaks to shifts in the context of large-scale tour scheduling. A mixed-integer programming (MIP) model is presented that includes shift and days-off scheduling along with break assignments for a multi-skilled workforce. To achieve tractability, a two-stage decomposition procedure is proposed that separates the tour scheduling problem (TShP) from the break assignment problem (BAP). The former MIP is first solved to determine the shifts and days off for the workforce that minimize labor and shortages costs over the planning horizon. The results are used as input to a second MIP that optimally places the breaks to minimize the costs of working hours and uncovered periods. Three implicit BAP formulations are investigated. To better understand the literature and the models previously developed, a 3-field break classification scheme is introduced. The first field characterizes the number of breaks permitted per shift, the second specifies whether the length of the breaks is fixed or variable, and the third limits their position in a shift. A complexity analysis of the resulting 12 BAPs along with a few special cases is also included. Most problems are shown to be strongly NP-hard. Computations are presented for a wide variety of scenarios for both the TShP and the BAP using data provided by a European airport ground handler company. In all, over 500 instances were investigated using high and low demand fluctuation curves and the various break and shift flexibility options. The results indicate that increasing flexibility in break regulations can make a significant difference in coverage, but the degree depends on the underlying structure of the demand curve as well as on the types of shifts permitted. Formulations with the most flexible shift and break regulations reduced undercoverage by up to 16.68% compared to the most common scenarios in which shifts are limited to a single lunch break.

Keywords Tour scheduling · Break classification · Break scheduling · Hierarchical workforce · Implicit formulations

1 Introduction

In most personnel scheduling problems, workforce demand is a function of the number of required tasks per period and, depending on the application, can have extremely high variance. To optimally cover demand, a tour scheduling prob-lem needs to be solved to determine the days off (days-off scheduling) and shift assignments (shift scheduling) for each worker on each day of the planning horizon. Once the days off are fixed, there are two immediate ways to modify a worker's tour when demand is highly variable: adjust the start time and length of a shift, and redo the assignment of breaks to shifts. Although much has been written about the first option, there is a noticeable absence in the literature when it comes to analyzing different break regulations, especially in the context of tour scheduling. Moreover, there is little practical evidence that planners have exploited the benefits offered by such regulations when it comes to better matching supply with demand.

One of the objectives of this paper is to show how break assignments can be used to improve the scheduling of airport ground handlers. When a flight arrives at an airport, baggage and cargo must be transferred to the terminal, handicapped passengers must be transported to either their next flight or

✉ Ferdinand Kiermaier
ferdinand.kiermaier@tum.de

Markus Frey
markus.frey@tum.de

Jonathan F. Bard
jbard@utexas.edu

1   TUM School Of Management, Technical University Munich, Arcisstr. 21, 80333 Munich, Germany

2   Graduate Program in Operations Research and Industrial Engineering, The University of Texas Austin, Austin, TX 78712-1591, USA

the baggage area, and the aircraft must be prepared for its next trip. These functions fall under the heading of *ground handling* and, depending on the number of arriving passengers and the distance flown, may require a dozen or more service workers to perform (Ashford et al. 1997). Personnel costs are the largest expense faced by the service provider and must be managed carefully in today's competitive environment. Ground handling as well as many other personnel scheduling problems requires the explicit assignment of shifts and days off to individual employees rather than to a generic workforce. This means that information on individual skills, availability, and overtime balances must be taken into account (e.g., see Love and Hoey 1990). In ground handling, the difference between peak and non-peak demand can be as large as three to one over the day. Meeting the demand without an excess of under- or over-coverage is an extremely challenging task. To the best of our knowledge, this paper is the first to analyze the full range of break options in tour scheduling.

The approach we take involves decomposing the full problem into a tour scheduling problem (TShP) and a break assignment problem (BAP). By separating the two, we create a framework in which different break models can be computationally evaluated without controlling for the basic tour scheduling parameters. As stand-alone problem, the BAP arises in daily break planning as well as intraday break assignment and the presented implicit models can incorporate a hierarchical multi-skill workforce. Other decomposition approaches based on branch and price turned out to be intractable for such large-scale real-world application considering individual workers and complex break regulations. The presented heuristic is flexible enough to be adopted to many real-world tour scheduling problems and general enough to incorporate all kinds of break regulations. To the best of our knowledge, there is no approach present in the literature to tackle a TShP of the size and complexity resulting from our ground handling application. The majority of the work in the area of breaks has focused on model development and the design of computational techniques for reducing problem size and runtimes in the context of shift scheduling. The primary example centers on the the implicit modeling of breaks in shift scheduling (see Aykin 1996, Bechtold and Jacobs 1990, Rekik et al. 2010). Our decomposition approach enables us to fully investigate the various implicit formulations in the wider context of tour scheduling and a hierarchical workforce. As part of the developments, we introduce a classification scheme for the BAP and present a complexity analysis of all major variations. Based on the different properties of the accompanying models, the flexibility and operational value of each is evaluated using data from a large European ground handling company. For applications where breaks are assigned dynamically over the day, e.g., for fast food, ground handling, and mail processing operations, the decomposition approach provides an upper bound

on costs. In such environments, the break assignment model can be used in a rolling horizon manner to support reactive planning.

In order to generate tours, we rely on templates that define a set of permissible shifts for each day of the planning horizon. The templates also include the option for a day off and therefore allow us to consider the availability of each employee each day of the week. To show the impact of shift starting time and shift duration flexibility and its interaction with break flexibility, we investigate two different scenarios. In the case of high shift flexibility, when early and late shifts are considered, we get more than 1400 possible shifts for the different combinations of starting times and working hours alone. This is in contrast to less than 30 possible shifts for the low flexibility case. As expected, we show that model size is strongly correlated with our ability to solve realistic instances in a timely manner.

The main contributions of this paper are as follows.

- A classification scheme and a complexity analysis of the BAP for all practical variations.
- The presentation of three efficient implicit formulation for assigning breaks to shifts in a manner that accounts for hierarchical skill levels.
- A heuristic decomposition procedure for scheduling a hierarchical workforce that can accommodate all permissible break placements in a shift (the heuristic supports reactive planning of breaks in response to unforeseen changes in hourly demand) that has successfully been applied at a major European ground handling service company.
- An analysis showing the impact of shift and break flexibility on solution quality and runtime.

The paper is organized as follows. In Sect. 2, we review the literature on ground handling operations and break assignment models. This is followed in Sect. 3 with a detailed description of the full problem, including an example and definitions related to templates and tours. An integrated mixed-integer programming (MIP) formulation of the tour scheduling problem based on weekly templates is presented in "Appendix B". In Sect. 4, we introduce a classification scheme for the different BAPs and analyze the complexity of each. The decomposition approach is highlighted in Sect. 5 along with an efficient implicit model for break assignments for a hierarchical workforce. This is followed by a computational study in Sect. 6 where the various break assignment models and the decomposition procedure are empirically evaluated. We close with some insights and observations on the proposed methodology.

2 Literature review

Personnel scheduling is the process of constructing work timetables for the employees of an organization to best meet the demand for its goods and services (Ernst et al. 2004). An overview of the subject can be found in Tien and Kamiyama (1982), Ernst et al. (2004), and Van den Bergh et al. (2013). In our review, we focus on tour scheduling, break assignments, implicit break formulations, and airport ground staff scheduling.

The first integer programming formulation for personnel scheduling problems was given by Dantzig (1954) who introduced a set-covering model in which each decision variable corresponded to a feasible tour. For all but the simplest of applications, however, problem size grows exponentially and becomes unmanageable when more than a handful of shift types and break options are considered. As a consequence, a large number of alternative formulations have been developed to reduce model size. Foremost are implicit formulations that either solve the problem by aggregating decisions or by modeling the rules for building shifts, breaks, or tours as constraints. Stolletz (2010) introduced a reduced set-covering-based formulation for the tour scheduling problem for check-in counter personnel in which all feasible shifts are enumerated explicitly as columns. A decision variable for each worker, each day of the planning horizon, and each shift is employed while the rules for building tours are modeled as constraints.

Çecik et al. (2001) piece together tours using a network flow formulation. In their approach, integer variables are used to indicate the frequently with which a specific shift is assigned to a specific number of workers. However, the principle of shift aggregation cannot be directly adapted to tour scheduling problems in which individual shift information, such as worker skill levels or over- and undertime, has to be considered.

Recent work to tackle the complexity of tour scheduling by means of branch and price is given by Restrepo et al. (2016). A multi-activity assignment to shifts is included such that skill requirements can be modeled. In the case of our application, though, branch and price leads to a large number of subproblems that are burdened by complex break regulations. Furthermore, shift start and shift length were sufficient in the subproblem presented by Restrepo et al. (2016) to find the best reduced cost-based shift, whereas in our case, breaks influence the net working time of an employee in a shift. An algebraic representation of the constraints that model the rules for building tours is presented by Brunner et al. (2009, 2010) for scheduling physicians in hospitals. In this paper, we extend those formulations to solve the tour scheduling subproblem that results from our decomposition.

The increase in problem complexity for the integrated tour and break scheduling problem is made clear in Brunner and

Stolletz (2014) where meal breaks must be assigned to shifts within a specified time window. While the set-covering formulation proposed by Stolletz (2010) could be solved to optimality with off-the-shelf software (CPLEX) for small instances, as the shift definition expanded and breaks were included it become impossible to find optimal solutions. To tackle the larger instances, branch and price was applied. However, computational results still showed gaps of up to 7% with 65 workers and a 1,000 node limit on the size of the search tree. Those instances also included a 15-day planning horizon with 34 periods per day, and an upper bound of 435 shifts per day (without breaks). In contrast, we consider a 7-day scheduling problem with 96 periods per day, up to 638 shifts per day and significantly higher complexity in break regulations. In Sect. 4, we give an overview of the various break regulations that have appeared in the in literature and classify relevant work.

Contributions to tour scheduling for ground handlers and other airport personnel have been made by Dowling et al. (1997), Chu (2007), Ásgeirsson (2012), and Lusby et al. (2011). In contrast to their work and other tour scheduling research found in the literature (e.g., see Van den Bergh et al. 2013), our framework considers the most flexible break regulations and contains over- and undertime constraints for individual workers with hierarchical skill levels. To the best of our knowledge, this leads to the most general formulations to date. Practical instances of tour scheduling problems for ground handlers can include up to several hundred workers.

Given that the general problem is NP-hard, finding optimal solutions to real-world instances is mostly beyond the reach of current technology. Dowling et al. (1997) developed a system to support the monthly rostering of 500 workers at a large airport. They used simulated annealing to minimize idle time under highly fluctuating demand. A novel component of their approach was an external rule engine to check for feasibility after each neighborhood search. Chu (2007) optimized daily schedules for baggage handlers at the Hong Kong International Airport using a goal programming-based algorithm. Taking a behavior-based approach, Ásgeirsson (2012) designed a heuristic that emulates the logic of personnel managers when creating schedules for an airport ground service company with 53 employees. He begins with a partial schedule derived from employee requests and then builds a full schedule that satisfies the prevailing labor laws and company policies. Solution quality was judged by how closely the days-on and days-off assignments matched employee requests. Finally, Lusby et al. (2011) tackled a ground crew rostering problem for a European airline using a column generation-based heuristic. Their goal was to create long-term rosters based on 6 days-on, 3 days-off work patterns.

Implicit break formulations for shift scheduling are presented by Bechtold and Jacobs (1990), Aykin (1996) and Rekik et al. (2010). Bechtold and Jacobs (1990) reduced

the size of the set-covering-type model by excluding the break information from the shift matrix. That is, breaks are treated in the aggregate and not explicitly assigned to shifts by the MIP. A set of constraints is introduced to guarantee that feasible breaks are available for the shifts chosen. In a post-processing step, breaks are assigned to shifts. A different approach is proposed by Aykin (1996) where a variable is introduced for each combination of shift and eligible break starting times. Again, beak information is removed from the shift matrix. In both formulations, predetermined break time windows and a fixed break duration are assumed.

The modeling of sub-breaks for shift scheduling was introduced by Rekik et al. (2010) who used the term *fractionable breaks*. A fractionable break has an overall duration that can be split into one or more sub-breaks with the restriction that the number of sub-breaks is within a predefined range. In their work, sub-break placement is constrained by work stretch durations between consecutive breaks. They present two implicit model formulations for shift scheduling based on extensions of the models of Bechtold and Jacobs (1990) and Aykin (1996). For reasons similar to those stated for the network flow formulation of Çecik et al. (2001), their modeling approaches cannot be easily adapted to shift scheduling problems that require individual shift information.

While many organizations take advantage of flexible break patterns within a shift, there has been little if any research on their use in the extended problem of tour scheduling. In part, this paper is aimed at filling this void. By separating the tour scheduling problem from the BAP, it is possible to use implicit formulations for dealing with a range of skill requirements. Building on the work of Rekik et al. (2010) and others, we present two implicit formulations for the BAP in which hierarchical skills are taken into account. The one based on Bechtold and Jacobs (1990), however, needs a post-processor to assign breaks to workers and so cannot be applied in all environments (e.g., bus driver scheduling; see Rekik et al. 2010). To circumvent this limitation, we introduce and evaluate an alternative implicit formulation that models the break rules as constraints.

3 Problem description

Our tour scheduling model is built around individual *templates* that specify the feasible set of shifts and days off that may be assigned to a worker each day of the planning horizon. We define a *shift* as a given number of continuous periods with a fixed start time and duration. A *shift type* is a set of shifts whose start times and durations fall within predefined ranges (e.g., early, mid-day or late). The set of all feasible shifts is denoted by $\mathcal{S}$.

Table 1 Example of three different weekly templates for three workers with the following shift types: E = early, L = late, X = day off

Worker	Days of the week						
	Mo	Tue	Wed	Thu	Fri	Sat	Sun
1	X	E/X	E/X	E	E	E/L	L
2	L	L/X	X	X	X	E/X	E
3	E	E	E/L	L	L	L/X	X

To illustrate the concept of a template, consider a one-week planning horizon and three workers, each having a different template as shown in Table 1. The notation E/L means that a worker can be assigned either an early or a late shift; L/X means that a worker can be assigned either a late shift or a day off, depending on demand. In the latter case, some workers may be on while others may be off. If a template offers all shift types on all days, i.e., E/L/X in the case of early and late shifts, the problem is equivalent to a tour scheduling problem without templates.

At the European service company that provided our ground handler application, the templates for each worker for the upcoming week are derived from a cyclic template defined for a flight season (e.g., summer or winter), as well as each worker's individual vacation plan, and are assumed to be given. The templates offer a concept that is in general useful to model many real-world requirements for tour scheduling such as individual holidays, personal preferences and employee carpools, aspects that we had to consider in the ground handling application but which we exclude from this study. For the general motivation behind the cyclic templates and its generation, we refer to Kiermaier et al. (2016). In Kiermaier et al. (2016), it is shown that incorporating stochastic information about employee demand when generating templates has a significantly positive impact especially when templates allow only limited flexibility, e.g., inflexible shift starting times and/or inflexible shift lengths.

The actual tasks for which the worker is responsible are assigned on the day of operation by a dispatcher. For the current tours, we want to assure that sufficient personnel are available to cover the demand. We now formalize the problem of creating a weekly tour for each member of the workforce based on his or her template. The underlying constraints reflect current operations; some are statute-based, while others are a result of union negotiations.

An overview of sets and indices used in the formulation is given in Table 2. The workforce is denoted by $\mathcal{W}$ such that each worker $w \in \mathcal{W}$ is assigned a template $p \in \mathcal{P}$, where p defines the eligible shift types (e.g., early, off, late, combination) for each day $d \in \mathcal{D}$ in the week. The set

Table 2 Sets and indices used in the formulation

Sets	Definition	Sets	Definition
$\mathcal{W}$	Workers, $w \in \mathcal{W}$	$\mathcal{M}$	Shift types, $m \in \mathcal{M}$
$\mathcal{W}(q)$	Workers with skill q	$\mathcal{M}(p)$	Eligible shift types in template p
$\mathcal{S}$	Shifts, $s \in \mathcal{S}$	$\mathcal{M}(d, p)$	Eligible shift types in template p on day d
$\mathcal{S}(d)$	Shifts on day d	$\mathcal{T}$	Number of planning periods per day, $t \in \mathcal{T}$
$\mathcal{S}(q, t)$	Shifts covering demand for skill q in period t	$\mathcal{T}^{\text{shS}}(m)$	Shift starting times for the shift type m
$\mathcal{B}$	Break patterns	$\mathcal{T}^{\text{shE}}$	Shift ending times
$\mathcal{B}(s)$	Break patterns for shift s	$\mathcal{T}^{\text{brS}}(r)$	Break starting times for the r-th sub-break
$\mathcal{D}$	Planning days, $d \in \mathcal{D}$	$\mathcal{T}^{\text{brE}}$	Break ending times of all sub-breaks
$\mathcal{D}(p)$	Planning days in template p	$\mathcal{T}^{\text{brE}}(r)$	Break ending times of the r-th sub-break
$\mathcal{P}$	Weekly templates, $p \in \mathcal{P}$	$\mathcal{Q}$	Skill levels, $q \in \mathcal{Q}$

$\mathcal{M}(d, p) \subseteq \mathcal{M}$ identifies the shift types that are allowed in template p on day d. A day in template p can either be a mandatory working day, an optional working day or a day off.

The set of feasible shifts for day d is given by $\mathcal{S}(d) \subseteq \mathcal{S}$. On mandatory working days, an employee must be uniquely assigned to a shift $s \in \mathcal{S}$, whereas on optional working days he may or may not be assigned to a shift, depending on the demand. Each day is divided into a set of periods $\mathcal{T} = \{1, \ldots, T\}$ of equal length (typically, 15 min). A shift $s \in \mathcal{S}$ in weekly template p has to satisfy the following rules.

S1. Starting and ending time windows: Each shift type m must start within the time band $\mathcal{T}^{\text{shS}}(m) \subset \mathcal{T}$ and must end within the time band $\mathcal{T}^{\text{shE}}$.

Multiple breaks are permitted per shift, but the actual number must fall between some lower and upper bound. This allows for maximum flexibility when making the break assignments. However, the length of each sub-break must also fall between some given bounds as must the time between them and their total duration. Adopting the nomenclature introduced by Rekik et al. (2010) for fractionable breaks, we impose the following rules:

B1. Number of breaks: There are $B^{\min} \leq r \leq B^{\max}$ sub-breaks in a shift.
B2. Workstretch between first break and shift start: The first break must start within $\left[\Delta^{\text{minBF}}, \Delta^{\text{maxBF}}\right]$ periods after the shift starts.
B3. Single break duration: Break $r \in \left[B^{\min}, B^{\max}\right]$ has a minimum and maximum duration of Δ_r^{minBD} and Δ_r^{maxBD} periods.
B4. Overall break duration: The minimum and maximum overall break duration must be within the interval $\left[\Delta^{\text{minBD}}, \Delta^{\text{maxBD}}\right]$.

B5. Workstretch duration: The workstretch duration after the r-th break has to be within $\left[\Delta_r^{\text{minBW}}, \Delta_r^{\text{maxBW}}\right]$ periods.
B6. Workstretch between last break and shift end: The last break must end within $\left[\Delta^{\text{minBL}}, \Delta^{\text{maxBL}}\right]$ periods before the shift ends.

Given the parameters in B1 to B6, the range of sub-break r's possible starting and ending times can be restricted to the subsets $\mathcal{T}^{\text{brS}}(r)$ and $\mathcal{T}^{\text{brE}}(r)$, respectively. In the following, we define a break pattern $b \in \mathcal{B}$ as a feasible set of sub-breaks with starting times and durations such that (B1) to (B6) are satisfied. The subset of feasible break patterns within a shift s that satisfies B5 and B6 is denoted by $\mathcal{B}(s)$. As we shall see in Sect. 4 these break regulations are the most general and flexible investigated to date. High flexibility, however, is accompanied by high complexity.

An additional attribute of the workforce is the level of skill that each individual brings to the job, whether he is a baggage handler, driver, ramp agent or other designation. Skill levels are ordered hierarchically, meaning that a worker at the higher end of the spectrum can fill in for one at the lower end when needed. Let $Q = \{q_1, \ldots, q_n\}$ be the ordered set of skills, where the relation $q_k \preceq q_l$ means that that a task requiring skill level $q_k \in \mathcal{Q}$ can be performed by any worker having a skill level of at least q_l for $l \geq k$. The vector $K = \left(K_{q,t,d}\right)_{q \in \mathcal{Q}, t \in \mathcal{T}, d \in \mathcal{D}}$ is used to represent the demand of a task, where each element $K_{q,t,d}$ gives the required demand for skill level q in time period t on day d. Each shift-covering demand for skill level q at time t is contained in the set $\mathcal{S}(q, t)$.

At the beginning of the planning horizon, each worker w has an accumulated bank of either overtime ($O_w \geq 0$) or undertime ($O_w < 0$) that must be kept within a target bandwidth $\left[O^-, O^+\right]$. Given a weekly template p_w for worker

Table 3 Cost parameters

Parameter	Definition
c^{work}	Unit cost for a worker when on duty
c^{uc}	Unit cost of undercoverage

w, when generating a tour, the following rules must be taken into account.

T1. Tour bandwidth: The length of a tour must be within the tour bandwidth $[O^-, O^+]$.

T2. Forward rotation: The start time of shift type m must be nondecreasing from one day to the next during the week.

T3. Bandwidth: For each shift type, there is a maximum allowed bandwidth of Δ^{bw} for shift starting times.

T4. Starting times: The number of maximum allowed shift starting times per week for shift type m is limited to K^{shift}.

The main objective of the problem is to minimize labor costs while covering as much demand as possible by fixing shift start and end times, days off, and breaks. Each worker incurs a cost of c^{work} per period which is assumed to be less or equal than the unit cost of undercoverage c^{uc}, i.e., $c^{\mathrm{work}} \leq c^{\mathrm{uc}}$. In determining labor costs, the time that a worker is not on duty, such as during a break, is not factored into the calculations. Table 3 lists the cost parameters.

It is well known that tour scheduling problems are NP-hard (see Lau 1996), which is one explanation for the lengthy runtime we experienced when trying to solve our first MIP model (see "Appendix B") of the ground handler staffing problem (GHSP) with CPLEX. In practice, breaks are often set at the beginning of the working day and not beforehand to better accommodate unforeseen changes in the demand profile. Therefore, to allow for this level of flexibility, as well as to reduce the difficulty of the problem, we have chosen to separate the BAP from the tour scheduling problem (see Sect. 5). In the next section, we categorize the full range of BAPs and undertake a complexity analysis of each.

4 Break assignment problem

The GHSP contains very general and flexible break regulations. Since the need for break assignments is common in many shift and tour scheduling problems, we classify and analyze the complexity of various BAPs found in the lit-

erature. A computational study of different versions of the problem is presented in Sect. 6.

Classification scheme

We use three sets of parameters to classify the BAP:

- *Single(S), multiple(M) or fractionable(F) breaks.* Break assignment formulations can be classified as single, multiple or fractionable break models. In single break BAPs only one break per shift may be assigned, whereas a fixed number $m \in \mathbb{Z}_+$ of breaks are required per shift for multiple BAPs. In contrast, fractionable break models have a lower and upper bound for the number of breaks that can be assigned to a shift.
- *Fixed(X) or variable(V) break length.* For BAPs with single or multiple breaks, each sub-break can either have a fixed length or a lower and upper bound on its length. Considering fractionable breaks, the length of each is inherently variable, but there is an aggregate break length given which is then split into sub-breaks for the shift.
- *Time windows(T) or workstretch(W) durations.* Time windows define the periods of a shift in which a break can start. In contrast, the workstretch duration defines a lower and upper bound on the number of consecutive periods of work in a shift. Workstretch durations implicitly define time windows, with the difference being that the size and position of each are interdependent and therefore not static.

In what follows, we classify the various versions of the BAP using the 3-tuple $[S, M, F | X, V | T, W]$. Figure 1 gives a hierarchical overview of the complexity of the more interesting models based on the findings given below. Those models associated with the parameter combinations not shown are for the single break case where the workstretch restrictions
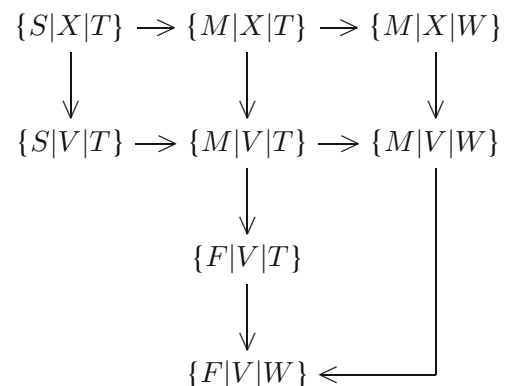
$$\{S|X|T\} \longrightarrow \{M|X|T\} \longrightarrow \{M|X|W\}$$

$$\{S|V|T\} \longrightarrow \{M|V|T\} \longrightarrow \{M|V|W\}$$

$$\{F|V|T\}$$

$$\{F|V|W\} \longleftarrow$$

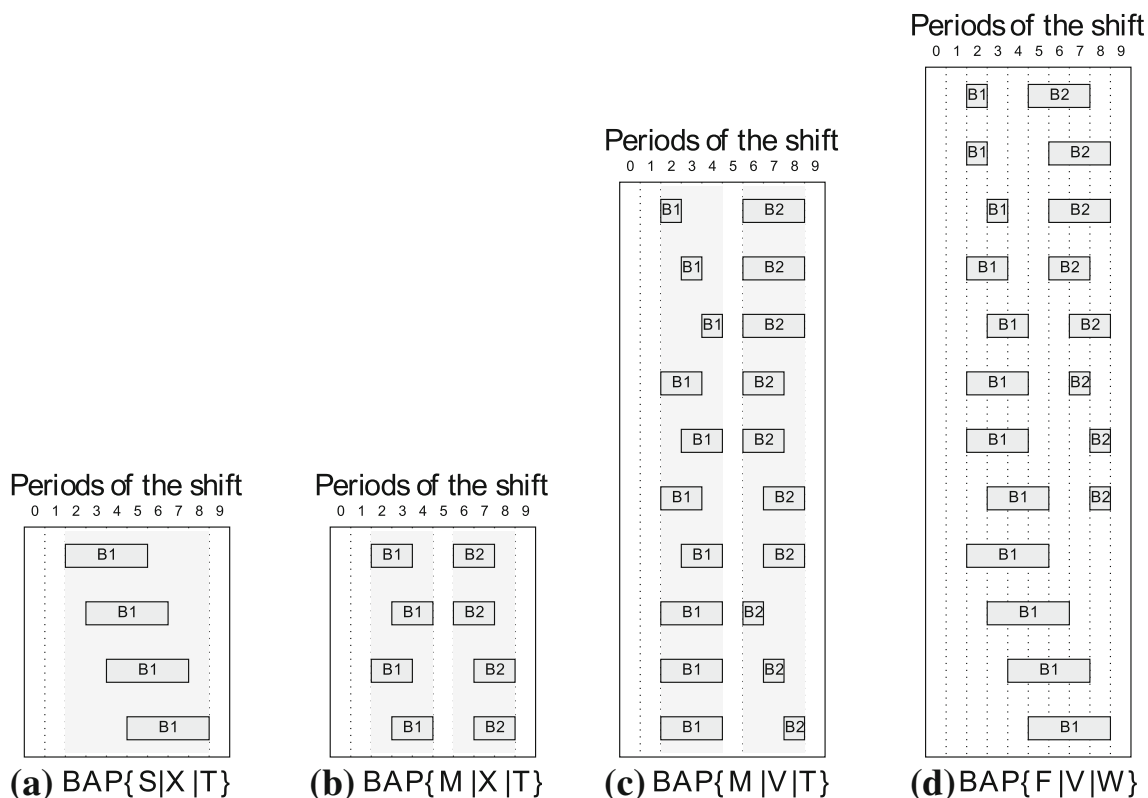**Fig. 1** Hierarchical overview of the complexity of various BAPs

Fig. 2 Break types and their possible assignments in a shift

lead to simple time windows and are therefore effectively the same problem class.

Figure 2 depicts four realizations of break types using the 3-field notation and their possible positions in a shift. In panels (a)–(c), the gray rectangles define the time windows in which a break can take place due to work regulations. In panel (d), there are no predefined break windows as workstretch durations limit the solution space. However, there is a minimum workstretch of 2 periods before the first break and between the first and second breaks, and a minimum workstretch duration of 1 period after the last break. In all examples, the overall break duration is equal to 4 periods. The overall break length in (b) is split equally between the first and second breaks, while in (c), the lengths of the first and second breaks are variable.

The variety of possible break assignments provides increased flexibility but at a cost. This will be confirmed in a theoretical sense by studying the computational complexity of the different break regulations. Table 4 provides an overview of break regulations in the staff scheduling literature. As one can see, there has been little research on flexible break regulations despite their positive impact on solution quality (see Sect. 6).

Table 4 Break regulations in the staff scheduling literature

Break regulation	References
$\{S\|X\|T\}$	Restrepo et al. (2016); Rong (2010); Topaloglu and Ozkarahan (2004); Alfares (2007); Bard (2004); Bard and Wan (2006); Bard et al. (2007); Bard and Wan (2008); Bechtold and Jacobs (1990); Brusco and Jacobs (2000, 1998); Lim et al. (2016); Mac-Vicar et al. (2017)
$\{M\|X\|T\}$	Rekik et al. (2004); Avramidis et al. (2010); Thompson and Pullman (2007); Ni and Abeledo (2007); Aykin (1996); Sungur et al. (2017)
$\{M\|V\|T\}$	Mirrazavi and Beringer (2007)
$\{M\|X\|W\}$	Côté et al. (2007); Kuo et al. (2014)
$\{F\|V\|T\}$	
$\{F\|V\|W\}$	Rekik et al. (2010); Widl and Musliu (2014); Beer et al. (2010)

Complexity analysis

To the best of our knowledge, Widl and Musliu (2014) are the only authors who previously addressed the complexity of the break assignment problem. Their NP-complete proof is based on reduction from 'Exact Cover by 3-Sets' (X3C) and is limited to arbitrary break patterns with 3 or more breaks. The assumptions needed for their proof do not hold for the

break regulations discussed in this paper or for the regulated break patterns found in the literature.

In our analysis, we assume that a set of shifts $\mathcal{S}' \subset \mathcal{S}$ is given such that each $s \in \mathcal{S}'$ is associated with a unique worker $w \in \mathcal{W}$. The objective of the BAP is to assign breaks to each shift s without violating the break constraints, i.e., B1 to B6, so that the costs associated with shift assignments and the uncovered demand are minimized. As mentioned in Sect. 3, each worker incurs a cost of c^{work} per period which is assumed to be less than or equal to the unit cost of undercoverage c^{uc}, that is, $c^{\text{work}} \leq c^{\text{uc}}$; off duty periods during a shift do not incur any cost. In the analysis, we work with a set partitioning formulation for the BAP in which all feasible break patterns $\mathcal{B}$ are enumerated. The set of feasible break patterns for shift s is denoted by $\mathcal{B}(s)$. The following parameters and variables are used in the developments.

Parameters

$c_{s,b}^{\text{work}}$ = cost for a worker assigned shift s with break pattern b

$a_{s,b,q,t} = 1$, if break pattern b in shift s covers demand for skill q in period t, 0 otherwise

$D_{q,t}$ = amount of over-coverage of demand for skill q in period t (can be negative)

Variables

$z_{s,b} = 1$, if break pattern b is assigned to shift s, 0 otherwise

$y_{q,t}^{-}$ = amount of uncovered demand for skill q in period t after break assignments

$y_{q,t}^{+}$ = amount of excess demand for skill q in period t after break assignments

Set partitioning formulation for the BAP

$$\text{Minimize} \sum_{s \in \mathcal{S}} \sum_{b \in \mathcal{B}(s)} c_{s,b}^{\text{work}} \cdot z_{s,b} + \sum_{q \in \mathcal{Q}} \sum_{t \in \mathcal{T}} c^{\text{uc}} \cdot y_{q,t}^{-} \quad (1)$$

subject to

$$\sum_{s \in \mathcal{S}'} \sum_{b \in \mathcal{B}(s)} a_{s,b,q,t} \cdot z_{s,b} + y_{q,t}^{+} - y_{q,t}^{-} = \max\{D_{q,t}, 0\}$$

$$\forall q \in \mathcal{Q}, t \in \mathcal{T} \quad (2)$$

$$\sum_{b \in \mathcal{B}(s)} z_{s,b} = 1 \quad \forall s \in \mathcal{S}' \quad (3)$$

$$z_{s,b} \in \{0, 1\} \quad \forall s \in \mathcal{S}', b \in \mathcal{B}(s) \quad (4)$$

$$y_{q,t}^{+}, y_{q,t}^{-} \geq 0 \quad \forall q \in \mathcal{Q}, t \in \mathcal{T} \quad (5)$$

The objective function (1) minimizes the sum of labor costs and the cost of uncovered demand over all periods. When a subset of periods has an initial shortage, i.e., $D_{q,t} < 0$, it might be desirable to assign a larger positive weight than

c^{uc} to the corresponding variables to discourage additional undercoverage. Constraints (2) account for demand reduction due to the break assignments to the shifts. The first term on the left-hand side reduces the demand in period t for each shift that includes a break in period t. The next two terms are complementary variables. The first represents the amount of remaining coverage in period t after the breaks have been assigned and will be nonnegative if a shortage exists. The second variable represents the undercoverage in period t after the breaks have been assigned and is minimized in (1). The right-hand side of (2) is taken to be nonnegative to avoid penalizing existing undercoverage.

The convexity constraints (3) enforce the requirement that each shift is assigned exactly one break pattern, and constraints (4) and (5) define the variables. In an optimal solution, at most y_t^{+} or y_t^{-} will be positive but never both because we are trying to minimize the latter. Also, it can be seen that these variables will always be integral in an optimal solution so there is no need to impose such a restriction.

We now present several theoretical results that characterize the complexity of various versions of the BAP. Proofs can be found in "Appendix A."

Proposition 1 *When the break length of BAP{S|X|T} is one period, the A-matrix associated with model* (1)–(5) *is totally unimodular (TU) implying that the associated problem is polynomially solvable.*

Corollary 1 *When the length of each sub-break of BAP{M|X |T} is one period the A-matrix associated with model* (1)– (5) *is totally unimodular (TU) implying that the associated problem is polynomially solvable.*

Corollary 2 *For a multi-skilled, hierarchical workforce, when the sub-break length of BAP{M|X|T} is one period, the A-matrix associated with model* (1)–(5) *is totally unimodular (TU), implying that the associated problem is polynomially solvable.*

The implication of Proposition 1 is of particular interest when scheduling relief breaks, which are short breaks of 15 min or less and typically one period in length. Thompson and Pullman (2007) reviewed 75 staff scheduling papers and found that 15% of them contained relief breaks. If considered separately, they can be assigned in polynomial time once the regular breaks are fixed.

When a break consists of more than one period, the sufficiency condition is lost so it is not possible to conclude that the full matrix for the set partitioning-like model (1)–(5) is TU. We now show that solving the problem in which only a single break of arbitrary length is to be assigned is not polynomially solvable.

Theorem 1 *The problem of assigning a single break to each shift to minimize the number of uncovered periods, that is, BAP{S|X|T}, is strongly NP-hard.*

The BAP$\{M|X|T\}$ is equivalent to the break model that occurs as a subproblem in Aykin (1996). By the principle of restriction stated by Garey and Johnson (1979), the BAP$\{M|X|T\}$ is also NP-hard in the strong sense since it includes the BAP$\{S|X|T\}$ as a special case; that is, when $m = 1$. Similarly, because the BAP$\{S|X|T\}$ and the BAP$\{M|X|T\}$ are special cases of the BAP$\{S|V|T\}$ and the BAP$\{M|V|T\}$, respectively, the latter two problems are also strongly NP-hard. Now, noting that the BAP$\{F|V|T\}$ is a generalization of the BAP$\{M|V|T\}$, with BAP$\{M|V|T\}$ being the special case of equalizing the lower and upper bound for the number of sub-breaks of the BAP$\{F|V|T\}$, the relationships in Figure 1 follow. However, the BAP$\{M|X|T\}$ and the BAP$\{S|V|T\}$ cannot be hierarchically related to each other.

Proposition 2 *The BAP$\{F|V|W\}$ is strongly NP-hard and includes the BAP$\{F|V|T\}$ as a special case.*

In a similar manner, the relations between BAP$\{M|X|T\}$ and BAP$\{M|X|W\}$, and between BAP$\{M|V|T\}$ and BAP$\{M|V|W\}$ can be shown. Each of these problems is strongly NP-hard.

5 Decomposition procedure

In the GHSP under investigation, we have to tackle two NP-hard problems: the tour scheduling problem (TShP) and the (fractionable) BAP$\{F|V|W\}$. Solving the integrated model given in "Appendix B", however, was seen to be intractable with CPLEX (see Sect. 6), so we adopted a sequential procedure. First, the TShP is solved to get feasible tours for each worker. Because this problem by itself is still intractable when two or more skill levels are considered simultaneously, we use downgrading. This approach starts with the highest skill level and works its way down to the lowest, covering as much demand as possible as the computations progress (Sect. 5.1). In light of the TShP solution, breaks are assigned to each shift by solving the BAP$\{F|V|W\}$ (Sect. 5.2).

5.1 Tour scheduling problem

Model formulation In the TShP, the start and end times of each shift for each worker, as well as their days off, are determined for the planning horizon. When applying the downgrading procedure, the TShP is solved for all workers $w \in \mathcal{W}(q_k)$ of a given single skill level $q_k \in \mathcal{Q}$ one level at a time. Let $\mathcal{Q}(q^+, q_k)$ be the set of skills that are higher or equal to skill q^+ and lower or equal to skill q_k. In each iteration, the set of workers $\mathcal{W}(q_k)$ is scheduled such that they cover the demand for skill levels in $\mathcal{Q}(q^+, q_k)$ with the proviso that they first cover higher level demand. In presenting the model, let TShP(q^+, q_k) be the tour scheduling problem being solved

for workers with skill level q_k with demand restricted to skill levels in $\mathcal{Q}(q^+, q_k)$.

Variables

$y_{m,w,d,t} = 1$, if shift of type m for worker w starts in period t on day d, 0 otherwise

$y_{w,d,t}^{\text{end}} = 1$, if shift for worker w ends in period t on day d, 0 otherwise

$z_{w,q,d,t} = 1$, if worker w is active during period t on day d at skill level q, 0 otherwise

$v_{m,w} =$ worker w's earliest starting time for shift type m

$w_{w,t} = 1$, if a shift for worker w starts in period t, 0 otherwise

$y_{q,d,t}^{+} =$ shortfall in demand for skill level q in period t on day d

TShP(q_k, q^+)

Minimize $\sum_{q \in \mathcal{Q}(q^+,k)} \sum_{d \in \mathcal{D}} \sum_{t \in \mathcal{T}} \left(\sum_{w \in \mathcal{W}(q_k)} \left(c^{\text{work}} + p_q \right) \right.$

$$\left. \cdot z_{w,q,d,t} + c^{\text{uc}} \cdot y_{q,d,t}^{+} \right) \tag{6}$$

subject to

Objective constraints

$$\sum_{w \in \mathcal{W}(q_k)} z_{w,q,d,t} + y_{q,d,t}^{+} \geq K_{q,d,t} \quad \forall q \in \mathcal{Q}(q^+,k) \in \mathcal{D}, t \in \mathcal{T} \tag{7}$$

$$\sum_{m \in \mathcal{M}(d,p_w)} \sum_{l \in \mathcal{T}^{\text{shS}}(m):l \leq t} y_{m,w,d,l} - \sum_{l \in \mathcal{T}^{\text{shE}}:l \geq t} y_{w,d,l}^{\text{end}}$$

$$= z_{w,q,d,t} \quad \begin{array}{l} \forall w \in \mathcal{W}(q_k), d \in \mathcal{D}(p_w), \\ t \in \mathcal{T} \end{array} \tag{8}$$

$$\sum_{q \in \mathcal{Q}(q^+,k)} z_{w,q,d,t} \leq 1 \quad \forall w \in \mathcal{W}(q_k), d \in \mathcal{D}(p_w), t \in \mathcal{T} \tag{9}$$

Shift constraints

$$\mathbb{1}_{\mathcal{D}^{\text{fix}}(p_w)}(d) \leq \sum_{m \in \mathcal{M}(d,p_w)} \sum_{t \in \mathcal{T}^{\text{shS}}(m)} y_{m,w,d,t} \leq 1$$

$$\forall w \in \mathcal{W}(q_k), d \in \mathcal{D}(p_w) \tag{10}$$

$$0 \leq \sum_{t \in \mathcal{T}^{\text{shE}}} t \cdot y_{w,d,t}^{\text{end}} -$$

$$\sum_{m \in \mathcal{M}(d,p_w)} \sum_{t \in \mathcal{T}^{\text{shS}}(m)} \left(t + \Delta^{\text{minS}} \right) \cdot y_{m,w,d,t}$$

$$\leq \Delta^{\text{maxS}} \quad \forall w \in \mathcal{W}(q_k), d \in \mathcal{D}(p_w) \tag{11}$$

Tour constraints

$$O^{-} - O_w \leq \sum_{d \in \mathcal{D}(p_w)} \left(\sum_{t \in \mathcal{T}^{\text{shE}}} t \cdot y_{w,d,t}^{\text{end}} - \right.$$

$$\sum_{m \in \mathcal{M}(d,p_w)} \sum_{t \in \mathcal{T}^{\text{shS}}(m)} t \cdot y_{m,w,d,t} \Bigg) \leq O^+ - O_w \quad \forall w \in \mathcal{W}(q_k)$$

$$(12)$$

$$\sum_{t \in \mathcal{T}^{\text{shS}}(m)} t \cdot \left(y_{m,w,d,t} - y_{m,w,d',t} \right) \cdot y_{m,w,d',t}$$

$$\leq 0 \quad \begin{array}{l} \forall m \in \mathcal{M}(d,p_w), w \in \mathcal{W}(q_k), \\ d, d' \in \mathcal{D}(p_w) : d < d' \end{array} \quad (13)$$

$$t \cdot y_{m,w,d,t} \geq v_{m,w} \cdot y_{m,w,d,t} \quad \forall m \in \mathcal{M}(d,p_w), \quad (14)$$

$$w \in \mathcal{W}(q_k), d \in \mathcal{D}(p_w), t \in \mathcal{T}^{\text{shS}}(m) \quad (15)$$

$$\sum_{t \in \mathcal{T}^{\text{shS}}(m)} t \cdot y_{m,w,d,t} - v_{m,w} \leq \Delta^{\text{bw}}$$

$$\forall m \in \mathcal{M}(d, p_w), w \in \mathcal{W}(q_k), d \in \mathcal{D}(p_w) \quad (16)$$

$$w_{w,t} \cdot M \geq \sum_{d \in \mathcal{D}(p_w)} y_{m,w,d,t}$$

$$\forall m \in \mathcal{M}(p_w), w \in \mathcal{W}(q_k), t \in \mathcal{T}^{\text{shS}}(m) \quad (17)$$

$$\sum_{\substack{t \in \bigcup_{\substack{d \in \mathcal{D}(p_w), \\ m \in \mathcal{M}(d, p_w)}} \mathcal{T}^{\text{shS}}(m)}} w_{w,t} \leq K^{\text{shifts}} \quad \forall w \in \mathcal{W}(q^{\text{work}}) \quad (18)$$

$$y_{m,w,d,t}, w_{w,t}, y_{w,d,t'}^{\text{end}}, z_{w,q,d,t'} \in \{0,1\}$$

$$\forall w \in \mathcal{W}(q_k), q \in \mathcal{Q}(q^+, k), d \in \mathcal{D}(p_w),$$
$$m \in \mathcal{M}(d, p_w), t \in \mathcal{T}^{\text{shS}}(m), t' \in \mathcal{T}^{\text{shE}} \quad (19)$$

$$v_{m,w} \in \mathbb{N} \quad \forall m \in \mathcal{M}, w \in \mathcal{W}(q_k) \quad (20)$$

$$y_{q,d,t}^{\text{uc}} \geq 0 \quad \forall q \in \mathcal{Q}(q^+, k), d \in \mathcal{D}, t \in \mathcal{T} \quad (21)$$

$$z_{w,q,d,t} \in \{0,1\} \quad \forall w \in \mathcal{W}(q_k), q \in \mathcal{Q}(q^+, k), d \in \mathcal{D}, t \in \mathcal{T} \quad (22)$$

Objective function The objective function (7) minimizes the sum of labor costs and the cost of not meeting the demand. When workers with skill level q_k are being considered, demand coverage for skill levels $q_k > \ldots > q_1$ is penalized with $p_{q_k} = 0 < \ldots < p_{q_1}$ such that workers with skill level q_k are prevented from covering demand that requires lower skills before covering demand requiring skill q_k.

Objective function constraints Constraints (7) ensure that either the demand for the skill levels in $\mathcal{Q}(q^+, k)$ is covered or a shortage is identified. Variable $z_{w,q,d,t}$, which indicates whether or not worker w is on duty in period t on d at skill level q, i.e., $z_{w,q,d,t} = 1$ or 0, is set in constraints (8). Constraints (9) guarantee that worker w is assigned at most one skill level in each period.

Shift constraints (S1) Constraints (10) set the start time of the shifts in worker w's weekly template, $p_w \in \mathcal{P}$. Here, the indicator function $\mathbb{1}_{\mathcal{D}^{\text{fix}}(p_w)}(d)$ is equal to 1 if day d is an obligatory day on for worker w, and 0 otherwise. (S2) The minimum and maximum shift length inclusive of breaks is bounded in the two-sided constraints in (11).

Tour constraints (T1) Bounds are placed on the maximum allowed undertime O^- and overtime O^+ for each worker

w in constraints (12). (T2) The forward rotation of shift starting times over the week is enforced by constraints (13). (T3) The earliest starting time for shift type m is determined by constraints (15), while the maximum permitted deviation between start times of each shift type during the week is restricted by constraints (16). (T4) The maximum number of shift starts in a week is bounded by constraints (17) and (18). The remaining constraints define the variables.

Sequential assignment procedure The sequential procedure for assigning shifts to workers to cover demand for all skill levels in each time period of the planning horizon is outlined in Algorithm 1. Assume that Algorithm 1 is currently at skill level q_2 in step 3, i.e., $q_k = q^+ = q_2$. If it is possible to cover the entire demand $K_{q_2,t,d}$ for skill level q_2 in all periods t on any day $d \in \mathcal{D}$ (see Step 8), Steps 9 through 11 are executed; that is, skill level q^+ is decremented by 1 such that $q^+ = q_1$, and demand $K_{q_1,d,t}$ for skill level q_1 on day d for all periods t is added to constraints (39). In objective function (7), covering demand for skill level $q < q_k$ is penalized by p_q to force the workers being scheduled to cover tasks requiring their skill level first. When TShP(q_k, q^+) is solved for the first time, we set $q_k = q^+ = q_n$ in Step 3.

Algorithm 1 Sequential Downgrading MIP-heuristic

1: Input: Set of skills $\mathcal{Q}$ and demand vector K
2: Output: Set of shifts $\mathcal{S}^*(d)$ for each day $d \in \mathcal{D}$
3: **for** $q_k = q^+ = q_n$ to q_1 **do**
4:     $next \leftarrow$ TRUE
5:     **while** $next$ **do**
6:         $next \leftarrow$ FALSE
7:         solve TShP(q_k, q^+)
8:         **if** $K_{q,d,t}$ is covered for any $d \in \mathcal{D}$ in all $t \in \mathcal{T}$ for all $q \in \mathcal{Q}(q^+, k)$ with $q \leq q_k$ **then**
9:             put $q^+ \leftarrow q^+ - 1$
10:             add demand $K_{q^+,d,t}$ on day d for all periods t to constraints (7)
11:             $next \leftarrow$ TRUE
12:         **end if**
13:     **end while**
14:     compute residual demand $K_{q,d,t}$ for all $q \in \mathcal{Q}(q^+, k), d \in \mathcal{D}$ and $t \in \mathcal{T}$
15: **end for**

The sub-procedure in Steps 5 – 13 continues until under-coverage appears for some demand $K_{q,d,t}$ with $q \in \mathcal{Q}(q^+, k)$ on any day d in some period t. Before decrementing q_k and starting the next iteration in which TShP(q_k, q^+) is solved for workers with the next lower skill level, we reduce the demand $K_{q,d,t}$ for $q \in \mathcal{Q}(q^+, k)$ in Step 14 in periods that are covered by workers in $\mathcal{W}(q_k)$. For example, consider two time periods t_1 and t_2 on day d and two skill levels q_1 and q_2 with the following demand.

$$K_{q_1,d,t_1} = 2, K_{q_1,d,t_2} = 1$$

$K_{q_2,d,t_1} = 1$, $K_{q_2,d,t_2} = 1$

For skill level q_2, assume that two workers are scheduled in t_1 and t_2, i.e., the demand for skill level q_2 is completely covered. Hence, the next lower skill level is added, i.e., $q^+ = q_1$. When running $\text{TShP}(q_k, q^+)$ again, the entire demand for skill level q_2 is covered, while one additional unit of demand is covered by one worker in period t_2 for skill level q_1. Finally, updating the residual demand for skill level q_1 gives $K_{q,d,t_1} = 1$ and $K_{q,d,t_1} = 0$.

5.2 Break assignment problem

Algorithm 1 terminates with a set of shifts $\mathcal{S}^*(d)$ for each day d, where each shift $s \in \mathcal{S}^*(d)$ uniquely corresponds to one worker $w \in \mathcal{W}$. We now assign break patterns to these shifts by solving the BAP. Because there are no restrictions on break patterns from one day to the next, the computations can be done independently for each day d.

5.2.1 Implicit formulations

The set partitioning formulation (1)–(5) explicitly assigns break patterns to shifts $\mathcal{S}(d)$. If we consider, for example, 400 shifts in which up to 3 sub-breaks are permitted, and each sub-break can start within a time window of 12 periods and have a duration between 2 and 3, we obtain $|\mathcal{B}| = 5,529,600$. This value defines the number of columns (and hence variables) in the set partitioning formulation. The resultant problem is much too large to solve in practice. Consequently, we have adopted an implicit modeling approach to the BAP with non-trivial model extensions to consider a hierarchical multi-skill workforce and investigate three different formulations.

The primary aim of implicit modeling is to reduce the number of decision variables by avoiding the need to enumerate each possible combination of shift and sub-break sequence. This is achieved by modeling the rules for building breaks as constraints. The implicit formulation associated with the GHSP is given in "Appendix D". The other two formulations are based on the shift scheduling work of Rekik et al. (2010) who extended the multiple break models of Aykin (1996) and Bechtold and Jacobs (1990) to include fractionable breaks. However, they found that due to the large number of variables still required by Aykin's formulation, computational difficulties arose when there is a high degree of break flexibility. With this in mind, we present an implicit formulation based on Bechtold and Jacobs (1990) that can accommodate significant break flexibility as well as a hierarchical workforce. Insights into the computation times of the different formulations are provided in Sect. 6.

5.2.2 Notation and components of implicit BAP formulations

To help define our model, we introduce the terms *break profile* and *shift profile*. Each break profile $\beta \in \mathcal{BP}$ corresponds to a number of sub-breaks denoted by B_β and their respective lengths. For example, using minutes as the unit of time, $\beta_1 = 15/30/15$ and $\beta_2 = 30/15/15$ are two break profiles for a fractionable break with an overall length of 60 min. A sub-break $k \in \mathcal{K}$ is defined for each break profile, each position in the break profile, each possible starting time and each skill level. Notice that we need a unique sub-break for each skill level as we have to consider the skill of the worker that is taking the break. In Rekik et al. (2010), a *shift profile* (or a *shift* in their terminology) $j \in \mathcal{J}$ is defined as a unique combination of start time, length, and admissible break profile. Because we are considering a hierarchical workforce, we have extended the definition of a shift profile to include a specific skill level. Some additional notation follows.

Sets

$\mathcal{J}_{(\beta,q)}$ = set of shift profiles associated with break profile β and skill q
$\mathcal{B}_{(\beta,r,q)}$ = set of sub-breaks associated with break profile β, position r and skill q

Parameters

$\rho_{t,k} = 1$, if break k covers period t, 0 otherwise
$h_{s,q}$ = number of workers assigned to shift s having skill q
$l_{\hat{q},q,t}$ = number of shifts for workers with skill $\hat{q}$ covering demand for jobs requiring skill q in period t
d_k = length of break k

Variables

S_j = number of workers assigned to shift profile $j \in J$
E_k = number of workers that are given sub-break k
$P_{\hat{q},q,t}$ = number of workers having skill $\hat{q}$ who are given a break in period t from a job requiring skill q

The implicit aspect of the BAP formulation requires constraints that:

Match shifts with break profiles From the TShP, we obtain the set of shifts as well as the skill of the corresponding worker for each day of the planning horizon. When solving the BAP, we consider workers with the same shift and skill jointly. Therefore, instead of explicitly assigning each worker's shift a break profile, we split the shifts into groups that are then associated with eligible break

profiles. *Example:* Suppose that 5 workers with skill q_2 are assigned a shift from 4 am to 11 am that is associated with two permissible break profiles $\beta_1 = 15/15/15$ and $\beta_2 = 15/30$. The workers are considered as a group, e.g., with 2 workers being assigned to β_1 and 3 workers to β_2.

Match sub-breaks with shift profiles The number of sub-breaks that are eligible in the r-th position of break profile β is selected by means of forward and backward constraints. For each break profile, each position within it, and each skill level, these constraints ensure the feasibility of a transportation problem (to be defined presently) between the break profiles' corresponding shift profiles (supply nodes) and the break profiles' eligible sub-breaks (demand nodes). Thus, it is not necessary to explicitly assign sub-breaks to each shift profile.

Meet workstretch durations between sub-breaks To ensure that the workstretch duration restriction between consecutive sub-breaks is met, a set of forward and backward constraints is established. For each break profile, each position in the break profile but the last, and each skill level, these constraints ensure the feasibility of a transportation problem that balances the number of sub-breaks with the number of their corresponding successor sub-breaks.

Assign breaks to shifts When sub-break k covers period t, the demand coverage in period t has to be reduced by the number of workers that are given sub-break k. For each skill $\hat{q}$, we must assure that the number of sub-breaks associated with skill $\hat{q}$ and covering period t does not exceed the number of workers with skill $\hat{q}$ that are scheduled to work in period t in the TShP solution. Furthermore, in the context of downgrading, it is necessary to consider whether workers with skill $\hat{q}$ are covering demand requiring lower skill. Accordingly, for each period t and skill q, the number of sub-breaks of type k covering period t is assigned to demand q for $q \leq \hat{q}$ in an aggregated way such that the solution of TShP is respected. By using aggregation, we do not have to explicitly reduce the demand for a specific skill for each sub-break and each period the sub-break covers. *Example* In period 5, 4 workers with skill q_2 are assigned to a job requiring skill q_2 and 2 workers are assigned to a job requiring skill q_1. BAP has to ensure that not more than 4 (2) workers with skill q_2 are given a break in period 5 from a job requiring skill q_2 (q_1). If 3 workers with skill q_2 have a break in period 5, their breaks can be assigned aggregately to jobs by either assigning 0, 1, or 2 breaks to a job requiring skill q_1 while assigning 3, 2, or 1 breaks, respectively, to a job requiring skill q_2.

Before presenting our model, we outline the idea of using forward and backward constraints to ensure the feasibility of certain balanced transportation problems that were first introduced in Rekik et al. (2004) and Çecik and Günlük (2004). We also give a high level description of the transportation problems included in our models. In all cases, the sum of the supply equals the sum of the demand.

Given a bipartite graph with a set of arcs A connecting the two sets of nodes N_1 and N_2, with each node $i \in N_1$ having supply O_i and each node $j \in N_2$ having demand D_i, we wish to determine the flow $Y_{i,j}$ from node i to node j for each $(i, j) \in A$. The constraints for the corresponding transportation problem $T(N_1, N_2)$ can be stated as follows.

$$\sum_{j:(i,j)\in\mathcal{A}} Y_{i,j} = O_i \qquad \forall\, i \in N_1 \qquad (23)$$

$$\sum_{i:(i,j)\in\mathcal{A}} Y_{i,j} = D_j \qquad \forall\, j \in N_2 \qquad (24)$$

Under the assumptions that a total order relation $\prec$ can be defined on set N_2, each supply node i is connected to a set of consecutive demand nodes P_i, and there exists no extraordinary overlap, i.e., there exists no two nodes i_1 and i_2 such that $P_{i_1} \subset P_{i_2}$, then the feasibility of transportation problem (23) – (24) is guaranteed by a set of forward, backward, and flow balance constraints. Using the following sets

$$N_2^s = \bigcup_{i=N_1} \{min(P_i)\}$$

$$N_2^e = \bigcup_{i=N_1} \{max(P_i)\}$$

$$N_2^B(j) = \left\{ j' \in N_2 | j \preceq j' \right\} \quad \forall\, j \in N_2^s$$

$$N_2^F(j) = \left\{ j' \in N_2 | j' \preceq j \right\} \quad \forall\, j \in N_2^e$$

$$N_1^B(j) = \left\{ i \in N_1 | P_i \subseteq N_2^B(j) \right\} \quad \forall\, j \in N_2^s$$

$$N_1^F(j) = \left\{ i \in N_1 | P_i \subseteq N_2^F(j) \right\} \quad \forall\, j \in N_2^e$$

and letting $n_2^s = min(N_2)$ and $n_2^e = max(N_2)$, the necessary constraints are

$$\sum_{j'\in N_2^F(j)} D_{j'} \geq \sum_{i\in N_1^F(j)} O_i \quad \forall\, j \in N_2^e \setminus \left\{ n_2^e \right\}$$

$$\sum_{j'\in N_2^B(j)} D_{j'} \geq \sum_{i\in N_1^B(j)} O_i \quad \forall\, j \in N_2^s \setminus \left\{ n_2^s \right\}$$

$$\sum_{i\in N_1} O_i - \sum_{j\in N_2} D_j = 0$$

To match sub-breaks with shift profiles, a transportation problem for each break profile, each position within the break profile, and each skill level has to be considered, i.e., a transportation problem $T(\mathcal{J}_{(\beta,q)}, \mathcal{B}_{(\beta,r,q)})$ for each $\beta \in \mathcal{BP}$,

$r \in \{1, \ldots, B_\beta\}$ and $q \in \mathcal{Q}$. The supply associated with node j is S_j, whereas the demand for node k is E_k. An arc (j, k) exists between a shift $j \in \mathcal{J}_{(\beta, q)}$ and a sub-break $k \in \mathcal{B}_{(\beta, r, q)}$ if break k is eligible in the r-th break position of shift j. The sub-breaks' starting times define a total order on set $\mathcal{B}_{(\beta, r, q)}$. Finally, we assume no extraordinary overlap between breaks of different shift profiles in $J_{\beta, q}$. This is not a limitation for the tour scheduling problem under consideration or for most practical applications since there exist extensions that allow for extraordinary overlap by modifying the forward and backward constraints, e.g., see Addou and Soumis (2007). Note that to adapt the forward and backward constraints presented in Rekik et al. (2010) to respect hierarchical skills, besides associating each $j \in \mathcal{J}$ and $k \in \mathcal{K}$ with a specific skill level, a separate transportation problem has to introduced for each skill level.

To obtain a feasible solution with respect to workstretch durations, the feasibility of a second set of transportation problems, $T(\mathcal{B}_{(\beta, r, q)}, \mathcal{B}_{(\beta, r+1, q)})$, must be assured. Here, the nodes correspond to breaks and are defined for all $\beta \in \mathcal{BP}$, $r \in \{1, \ldots, B_\beta - 1\}$ and $q \in \mathcal{Q}$. The supply and demand of each node k is E_k. Nodes $k_1 \in \mathcal{B}_{(\beta, r, q)}$ and $k_2 \in \mathcal{B}_{(\beta, r+1, q)}$ are connected if the workstretch duration between their corresponding breaks is within δ^{minBW} and δ^{maxBW}. Further, based on the break starting times, a totally ordered relationship $\prec$ on $\mathcal{B}_{(\beta, r, q)}$ is defined which per se excludes extraordinary overlap.

5.2.3 Implicit BAP formulation based on the Bechtold and Jacobs model

Our implicit formulation of the BAP is based on the model developed by Bechtold and Jacobs. It embodies the four components mentioned above along with a set of demand constraints. The objective here and for the two other formulations in Appendices C and D is to minimize the cost of uncovered periods minus the savings that result when workers are on their breaks.

Implicit BAP I (IBAP1)

$$\text{Minimize} \sum_{q \in \mathcal{Q}} \sum_{t \in \mathcal{T}} c^{\text{uc}} \cdot y_{q,t}^- - \sum_{j \in \mathcal{J}} \sum_{r \in \{1, \ldots, B_j\}} \sum_{k \in \mathcal{K}_{j(r)}} c^{\text{work}} \cdot d_k \cdot E_k$$

$$(25)$$

subject to

$$\sum_{\hat{q} \geq q} P_{\hat{q}, q, t} + y_{q,t}^+ - y_{q,t}^- = D_{q,d,t} \quad \forall q \in \mathcal{Q}, t \in \mathcal{T} \tag{26}$$

$$\sum_{j \in \mathcal{J}_{s,q}} S_j = h_{s,q} \quad \forall s \in \mathcal{S}, q \in \mathcal{Q} \tag{27}$$

$$\sum_{q \leq \hat{q}} P_{\hat{q}, q, t} = \sum_{k \in \mathcal{K}_{\hat{q}}} \rho_{t,k} \cdot E_k \quad \forall \hat{q} \in \mathcal{Q}, t \in \mathcal{T} \tag{28}$$

$$P_{\hat{q}, q, t} \leq l_{\hat{q}, q, t} \quad \forall \hat{q}, q \in \mathcal{Q} : q \leq \hat{q}, t \in \mathcal{T} \tag{29}$$

$$\sum_{k' \in \mathcal{B}_{(\beta, r, q)}^F (k)} E_{k'} - \sum_{j \in \mathcal{J}_{(\beta, q)}^F (k)} S_j \geq 0 \quad \begin{array}{l} \forall \beta \in \mathcal{BP}, r \in \{1, \ldots, B_\beta\}, q \in \mathcal{Q}, \\ k \in \mathcal{B}_{(\beta, r, q)}^{\text{e}} \setminus \{k_{(\beta, r, q)}^{\text{e}}\} \end{array}$$

$$(30)$$

$$\sum_{k' \in \mathcal{B}_{(\beta, r, q)}^B (k)} E_{k'} - \sum_{j \in \mathcal{J}_{(\beta, q)}^B (k)} S_j \geq 0 \quad \begin{array}{l} \forall \beta \in \mathcal{BP}, r \in \{1, \ldots, B_\beta\}, q \in \mathcal{Q}, \\ k \in \mathcal{B}_{(\beta, r, q)}^{\text{s}} \setminus \{k_{(\beta, r, q)}^{\text{s}}\} \end{array}$$

$$(31)$$

$$\sum_{k \in \mathcal{B}_{(\beta, r, q)}} E_k - \sum_{j \in \mathcal{J}_{(\beta, q)}} S_j = 0 \quad \forall \beta \in \mathcal{BP}, r \in \{1, \ldots, B_\beta\}, q \in \mathcal{Q} \tag{32}$$

$$\sum_{k' \in \mathcal{B}_{(\beta, r+1, q)}^F (k)} E_{k'} - \sum_{k' \in \mathcal{B}_{(\beta, r, q)}^F (k)} E_{k'} \geq 0$$

$$\begin{array}{l} \forall \beta \in \mathcal{BP}, r \in \{1, \ldots, B_\beta - 1\}, q \in \mathcal{Q}, \\ k \in \mathcal{B}_{(\beta, p+1, q)}^{\text{e}} \setminus \{k_{(\beta, p+1, q)}^{\text{e}}\} \end{array}$$

$$(33)$$

$$\sum_{k' \in \mathcal{B}_{(\beta, r+1, q)}^B (k)} E_{k'} - \sum_{k' \in \mathcal{B}_{(\beta, r, q)}^B (k)} E_{k'} \geq 0$$

$$\begin{array}{l} \forall \beta \in \mathcal{BP}, r \in \{1, \ldots, B_\beta - 1\}, q \in \mathcal{Q}, \\ k \in \mathcal{B}_{(\beta, r+1, q)}^{\text{s}} \setminus \{k_{(\beta, r+1, q)}^{\text{s}}\} \end{array}$$

$$(34)$$

$$\sum_{k \in \mathcal{B}_{(\beta, r+1, q)}} E_k - \sum_{k \in \mathcal{B}_{(\beta, r, q)}} E_k = 0 \quad \forall \beta \in \mathcal{BP}, r \in \{1, \ldots, B_\beta - 1\}, q \in \mathcal{Q}$$

$$(35)$$

$$P_{\hat{q}, q, t} \in \mathbb{Z}_+ \quad \forall \hat{q}, q \in \mathcal{Q} : q \leq \hat{q}, t \in \mathcal{T} \tag{36}$$

$$S_j, E_k \in \mathbb{Z}_+ \quad \forall j \in \mathcal{J}, k \in \mathcal{K} \tag{37}$$

Constraints (26) determine the amount of demand for skill level q on day d in period t that cannot be covered. While Rekik et al. (2010) considered a shift scheduling problem, for the BAP that we are solving, the set of shifts are derived from the TShP and taken as input. To match shifts with permissible sub-breaks, each shift needs to be associated with a break profile. In constraints (27), the linking of shifts with break profiles is done implicitly, i.e., instead of explicitly assigning a break profile to each shift, identical shifts assigned to workers with the same skill are aggregated based on admissible break profiles. Constraints (28) ensure that the number of sub-breaks given in period t for workers with skill $\hat{q}$ are only distributed among jobs that require at most skill $\hat{q}$. Note that this is only possible when each sub-break k is associated with a specific skill. In constraints (29), the assignment of sub-breaks is bounded by $l_{\hat{q}, q, t}$ such that breaks can only be given when the underlying jobs are being performed by workers having the specified skill. Again, workers with skill $\hat{q}$ who are performing a job that requires skill q in period t are grouped together. This arrangement guarantees that a feasible solution to the problem of explicitly assigning sub-breaks to workers can be found in a post-processing step.

The feasibility of the transportation problems for workstretch duration restrictions between successive sub-breaks is ensured by the forward, backward and flow balance constraints given in (31), (32) and (33), respectively. Similarly, constraints (33), (34) and (35) are respectively the forward, backward and flow balance constraints that ensure that enough sub-breaks matching the shift profiles are assigned. Variables are defined in (36) and (37).

Table 5 Parameter values for the fixed and flexible work templates

| Type | Start time window | | Duration (Periods) |
	Early shift	Late shift	
Fix	[4 am, 6 am]	[12 am, 2 pm]	41
Flex	[3 am, 10 am]	[11 am, 6 $text$ pm]	[36, 46]

Table 6 Work templates—fix and flex

| Week | Day | | | | | | |
	Mo	Tu	We	Th	Fr	Sa	Su
(a) Fix							
1				E	E	L	L
2	L	L					
3	E	E	E/L	L	L		
4				E	E	E	E
5	L	L				E	E
6	E	E	L	L			
7			E	E	L	L	L
8	L			E	E	E	
9	L	E	E				
10		E	E	L	L	L	L
(b) Flex							
1	E	E	E/L	L	L	L/X	
2			E	E	E/L	L	
3	L	L/X					E
4	E	E/L	L	L	L/X		
5		E	E	E/L	L	L	
6	L/X					E	E
7	E/L	L	L	L/X			
8		E	E	E/L	L	L	L/X
9			E	E	E/L		
10	L	L	L/X				

6 Computational study

The computations were performed on a Windows 7 platform with 4 GB RAM and a 2.8 GHz processor. All models were implemented in JAVA and solved with CPLEX 12.6. The data sets were provided by AeroGround GmbH, a major European ground handling service company. Each working day was divided into ninety-six, 15-min periods.

Our study is based on two templates: 'Fix' and 'Flex,' differing in the length of the start time window, the duration of a shift (see Table 5), and in the flexibility of the possible shift types per day (see Table 6). Both templates are currently being used by AeroGround. The resulting number of shifts for the Fix template is at least the number of starting times, which is 8, while for Flex templates there are at least 319 ($= 29 \times 11 =$ number of start times multiplied by the number of shift lengths). Note that each hour is divided in four, 15-minute periods. If on one day, two shift types are possible, e.g., see Table 6 in week 1 on Wednesday, then the number of feasible shifts doubles.

We consider from 10 to 300 workers and five skill levels, where it is assumed that the demand for each level is 20% of the total. The maximum amount of overtime and undertime per worker is given by $O^+ = O^- = 100$; the individual amount of overtime or undertime per worker was randomly assigned from the interval [0,100]. Since undercoverage in a period indicates missing workers, we assume that the cost per period of undercoverage is equal to the labor cost per period which is 1.

Table 7 presents the settings for the break regulations. The total break length (see column "Dur tot") in each break regulation is set to 6 periods. The first sub-break starts between periods 8 and 24 periods after the shift start (column "FB"). The last break (column "LB") for multiple or fractional breaks has to start between 3 and 16 periods before the shift ends. For all regulations with multiple breaks, 3 sub-breaks have to be assigned. To guarantee a total break length of 6, in multiple break scenario $\{M|X|T\}$ with fixed sub-break length, each sub-break is 2 periods, while in multiple break scenario $\{M|V|T\}$ with variable sub-break length, each sub-break takes between 1 and 4 periods (column "Dur SB"). After the shift starts, the time windows of the second and third breaks are [26, 29] and [30, 33], respectively (column "TW"). In the two scenarios with fractional breaks, $\{F|V|T\}$

and $\{F|V|W\}$, 1 to 3 sub-breaks can be assigned to each shift, which are either based on time windows or workstretch durations. The time windows for the second and third sub-breaks associated with $\{F|V|T\}$ are equal to those in the multiple break regulation case. For fractional break regulation $\{F|V|T\}$, the workstretch duration between the successively sub-breaks after the first sub-break is from 3 to 24 periods (column "WS").

With respect to demand, both low- and high-level scenarios were investigated along with two different demand profiles: "varying demand" (VD) and "stable demand" (SD). Figure 3 depicts the profiles VD and SD for Monday to Sunday where each day is divided into 98 periods. The VD high level demand curve represents the true demand during the course of a week for each qualification of the sponsoring company. The VD low-level demand curves were derived from this curve by randomly lowering the demand. "Appendix E" yields the three stable demand curves for the same days and periods.

To uniquely identify a test instance, we use the 4-tuple representation $I\{wor, pro, dem, temp\}$ specifying the workforce size $wor \in \{10, \ldots, 50, 100, 150, 200, 250, 300, 350, 400\}$, the demand profile $pro \in \{VD, SD\}$, the level $dem \in$
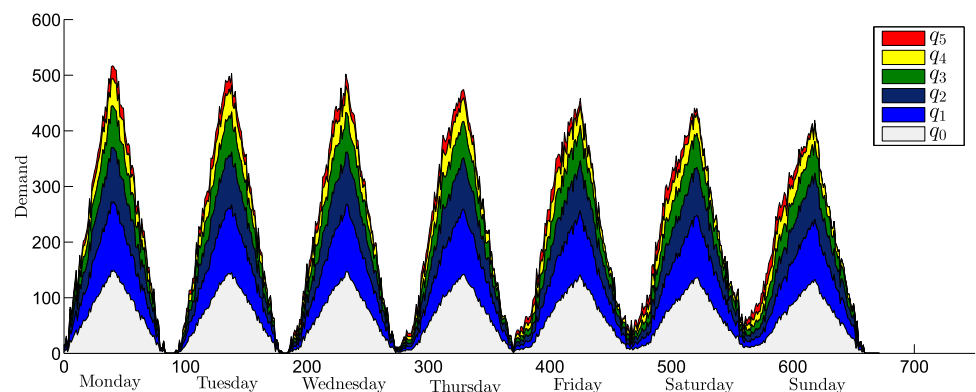
Table 7 Settings for the break regulations

Break regulation	Setting						
	#Breaks	Dur tot	Dur SB	WS	TW	FB	LB
$\{S\|X\|T\}$	1	6				[8, 24]	
$\{M\|X\|T\}$	3	6	2		[26, 29], [30, 33]	[8, 24]	
$\{M\|V\|T\}$	3	6	[1, 4]		[26, 29], [30, 33]	[8, 24]	[3, 16]
$\{F\|V\|T\}$	[1, 3]	6	[1, 6]		[26, 29], [30, 33]	[8, 24]	[3, 16]
$\{F\|V\|W\}$	[1, 3]	6	[1, 6]	[3, 24]		[8, 24]	[3, 16]

Fig. 3 Profiles for high-level demand
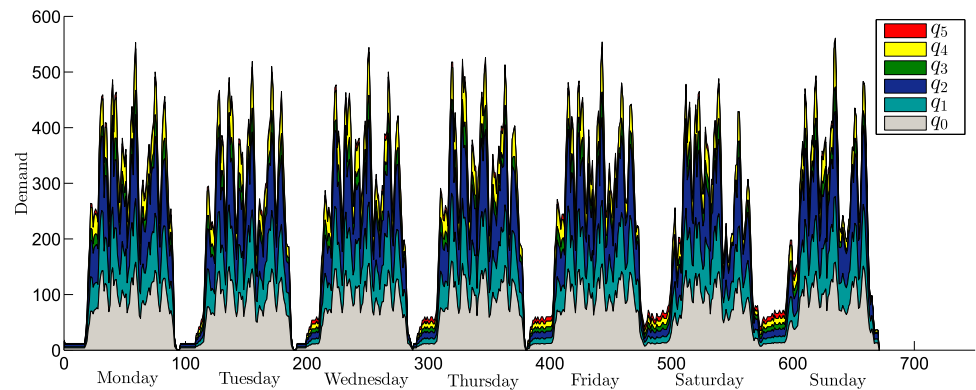
(a) Varying high level demand

(b) Stable high level demand

{low, high}, and the work template in Table 6 $temp \in$ {Fix, Flex}.

The analysis is presented in two separate subsections: In Sect. 6.1, we compare the solution quality of our MIP-heuristic with solutions obtained with the compact MIP (CMIP) formulation in "Appendix B," which integrates the tour scheduling problem and the BAP. In Sect. 6.2, we evaluate the "cost" of break flexibility by first generating tours by solving TShP (7)–(21). In light of those results, we study the different combinations of BAP formulations and BAP regulations to determine runtime and solution quality. To the best of our knowledge, this is the first study in which real data are used to determine the effect of break regulations on large tour scheduling problems.

6.1 Performance MIP-Heuristic

To study the performance of the MIP-heuristic in comparison with the CMIP, we restrict the workforce size to be between 10 and 65 employees for the low demand scenario. The break regulations investigated are $\{S|X|T\}$, which represents the most common break regulations considered in literature (see Table 4), and BAP$\{F|V|T\}$, which offers the greatest flexibility (see Fig. 1). In striving for a fair comparison, we solve Implicit BAP formulation III in "Appendix D" in the second step of the MIP-heuristic due to its structural similarity to the break assignment constraints in the CMIP.

The results for BAP$\{S|V|T\}$ and BAP$\{F|V|W\}$ for a workforce size of 10 and 50 employees are summarized in Tables 8 and 9, respectively. The results for all instances

Table 8 Results for the CMIP and the MIP-heuristic for break regulation $\{S|V|T\}$

Instance	CMIP			MIP-heuristic			
	Obj	Undercov	Runtime (s)	Obj	Undercov	Runtime (s)	GAP*
{10, low, SD, fix}	7570	6520	15.77	7570	6520	3.68	0.00
{50, low, SD, fix}	9805	2630	268.53	9975	2800	9.51	1.7
{10, low, SD, flex}	7350	6033	33.9	7350	5690	8.82	0.00
{50, low, SD, flex}	7350	3328	10, 074.84	7350	3010	12.72	0.00
{10, low, VD, fix}	9086	8036	2.6	9086	8036	1.34	0.00
{50, low, VD, fix}	10, 188	3013	830.01	10, 486	3311	12.58	2.84
{10, low, VD, flex}	9084	7857	20.55	9089	7133	4.93	0.01
{50, low, VD, flex}	9084	4546	8343.92	9185	3503	87.52	1.10

Table 9 Results for the CMIP and the MIP-heuristic for break regulation $\{F|V|W\}$

Instance	CMIP			MIP-heuristic				
	Obj	Undercov	Runtime (s)	Obj	Undercov	Runtime (s)	GAP*	GAP
{10, low, SD, fix}	7510	6460	32.32	7510	6460	3.56	0.00	
{50, low, SD, fix}	9487	2312	2744.08	9640	2465	22.21	1.59	
{10, low, SD, flex}	7350	5690	203.92	7350	5690	9.05	0.00	
{50, low, SD, flex}				7350	3010	17.71		1.12
{10, low, VD, fix}	9084	8034	31.97	9086	8036	3.54	0.02	
{50, low, VD, fix}	9784	2564	10800.3	10243	3, 068	18.21	4.48	
{10, low, VD, flex}	9084	7580	843.80	9084	7128	8.72	0.00	
{50, low, VD, flex}				9160	3478	98.51		1.35

between 10 and 65 employees are shown in Tables 16 and 17 of "Appendix F." Column "Instance" identifies the scenario with respect to number of workers, work template, and demand characteristics. Columns "obj" and "undercov" report the objective value and undercoverage costs, respectively. The computation times are given in column "runtime" in seconds. Column "GAP*" gives the % gap between the optimal solution of the CMIP and the MIP-heuristic. In Table 9, we have added the column "GAP" to report the % gap between the solution found by the MIP-heuristic and the best LP bound obtained when no solution was found for the CMIP within the 5-h (18,000 s) limit placed on each run.

The computations reveal that the MIP-heuristic performs well with respect to the CMIP. The largest gaps of 7.52% and 9.33% for the variable and stable demand curves, respectively, are reached when no feasible or optimal solution for CMIP is found (see Table 17 of "Appendix F," instances {65, low, SD,, fix}) and {65, low, VD, fix})). If we restrict the results to those instances for which the optimal solution of CMIP is found, the gaps of the MIP-heuristic are at most 4.48% (see Table 9, instance {50, low, SD, fix}). The gaps between CMIP and the MIP-heuristic for the instances with the fix work templates are on average greater than those for the flex work template. In the first step of the MIP-

Table 10 Number of workers considered for the two demand levels

Demand curve	# Workers
Low	[10, 50, 100]
High	[200, 250, 300]

heuristic, the shifts obtained under flexible work templates provide more demand coverage than the shifts obtained with fix work templates and, therefore, lower the benefits of break flexibility.

A closer look at Tables 8 and 9 reveals the impact of break flexibility. Consider, for example, the instances with 50 workers. In the singe break scenario $\{S|V|T\}$, if we compare instances for fixed and flexible work templates based on the stable demand curve, e.g., instances {50, low, SD, fix} and {50, low, SD, fix}, flexible work templates improve the solution by $2, 455 (= 9, 805 - 7, 350)$ or 25%. In contrast, comparing instance {50, low, SD, flex} for break regulations $\{S|V|T\}$ and $\{F|V|W\}$, we see that latter improves the solution by $318 (= 9, 805 - 9, 487)$, which represents 13% $(= 100\% \times 318/2, 455)$ of the improvement in the previous comparison. The improvement between the two break regulations becomes more significant when we focus on instances with varying demand. The difference between instances

{50, low, VD, fix} and {50, low, VD, flex} with break regulation $\{S|V|T\}$ is 1, 104 (= 10, 188 − 9084) or 10.8%. The comparisons between the two break regulations $\{S|V|T\}$ and $\{F|V|W\}$ for these instances leads to a decrease in the objective of 404 (= 10, 188 − 9, 784) which is equivalent to 37% of the improvement that is obtained when flexible rather than fixed work templates are employed.

Our two-step approach is seen to provide high-quality solutions in less than 100 seconds. Timely computations are critical when there is a frequent need to replan shifts and breaks due to workforce and demand fluctuations. Moreover, when the demand is largely uncertain from day to day, the only option is to reset the breaks since shift start and end times cannot be adjusted during the course of the day or even for the upcoming day without the likelihood of violating labor laws or incurring overtime costs. In the next section, break flexibility is explored in more detail.

6.2 Benefits of break flexibility

In the second phase of our experiments, we examined the relationships among break flexibility, undercoverage, and runtime for the various scenarios in Table 7. To begin, Algorithm 1 is solved to generate shifts and then the BAP is solved using the three different implicit formulations. The study demonstrates the degree to which the various break regulations can serve to reduce undercoverage when the shifts are given. For the variable and stable demand curves with both low and high demand, we consider the workforce sizes given in Table 10.

Table 11 provides the statistics obtained from Algorithm 1 in the first step of the MIP-heuristic. Column "time (s)" reports the runtime in seconds. The average number of workers and the resulting number of shifts per day are shown in column "#worker" and "#shifts," respectively. Based on these shifts, the columns under the major heading "#sub-breaks" report the number of sub-breaks for each break regulation in Table 7. It can be seen that for fixed work templates fewer shifts are generated, and hence there are fewer sub-breaks for each break regulation compared to the instances that have flexible work templates. For example, for any number of workers associated with instances {·, high, VD, flex}, there are significantly more sub-breaks generated for any consid-

Table 11 Results of the ThSP

| Instance	Time (s)	# Workers	#Shifts	#Sub-breaks				
				$\{S|V|T\}$	$\{M|X|T\}$	$\{M|V|T\}$	$\{F|V|T\}$	$\{F|V|W\}$
{10, low, SD, fix}	0.79	4.28	1.1	87.43	82.29	97.71	1429.71	4278.86
{50, low, SD, fix}	4.25	29.29	3.86	240	308.57	3685.71	5093.14	11, 460
{100, low, SD, fix}	8.56	58.57	4.71	260.57	336.86	4122.86	5696.57	12, 303.43
{10, low, SD, flex}	3.4	7.14	1.43	316.29	128.57	1285.71	2108.57	6378
{50, low, SD, flex}	11.45	25	5	468	425.14	4251.43	6227.14	10, 354.29
{100, low,SD, flex}	1552.43	50	7.71	379.71	281.14	2811.43	4238.57	10, 675.34
{10, low, VD, fix}	1.96	4.28	2.09	104.57	122.57	1465.71	2027.14	4981.71
{50, low, VD, fix}	4.53	29.29	12.14	282	418.29	4662.86	6428.57	13, 182
{100, low,VD, fix}	9.22	58.57	11.29	281.14	421.71	4714.29	6498.86	13, 146.86
{10, low, VD, flex}	8.21	7.14	5.29	371.14	250.29	2502.86	4228.29	10, 798.29
{50, low, VD, flex}	66.15	28.14	26.71	578.57	795.42	7954.29	11, 544	16, 317.43
{100, low, VD, flex}	2567.69	51.71	20.28	478.29	798	7980	11.490	16, 263.43
{200, high, SD, fix}	51.82	11, 714	5.29	253.71	339.43	3, 394.29	5, 374.29	12, 022.29
{250, high, SD, fix}	60.75	146.43	8.71	273.43	397.71	3, 977.14	6, 147.43	12, 830.57
{300, high, SD, fix}	40.6	175.71	9.14	282	424.29	4, 242.86	6, 534	13, 182
{200, high, SD, flex}	2102.32	112	19	418	404.57	4, 045.71	6, 167.14	11, 882.57
{250, high, SD, flex}	2503.74	125	18.57	408.57	487.71	4877.14	7311.43	12, 538.29
{300, high, SD, flex}	7628.69	150	19.85	436.86	524.57	5254.29	7518	12, 467
{200, high, VD, fix}	33.15	117.14	11.57	282	400.29	4002.86	6182.57	13, 182
{250, high, VD, fix}	35.71	164.43	11.28	277.71	423.43	4234.29	6498.86	13, 006.29
{300, high, VD, fix}	40.50	175.71	11.28	279.43	405.43	4054.29	6252.86	13, 076, 57
{200, high, VD, flex}	3212.99	107.42	24.57	468.86	799.71	7997.14	12, 022.29	15, 997.71
{250, high, VD, flex}	4590.07	142.86	24.86	452.57	789.43	7894.29	11, 478.86	15, 223.71
{300, high, VD, flex}	4625.97	152.57	24.71	420.86	833.14	8331.43	12, 081.43	16, 368

(a) $\{S|V|T\}$

(b) $\{F|V|W\}$
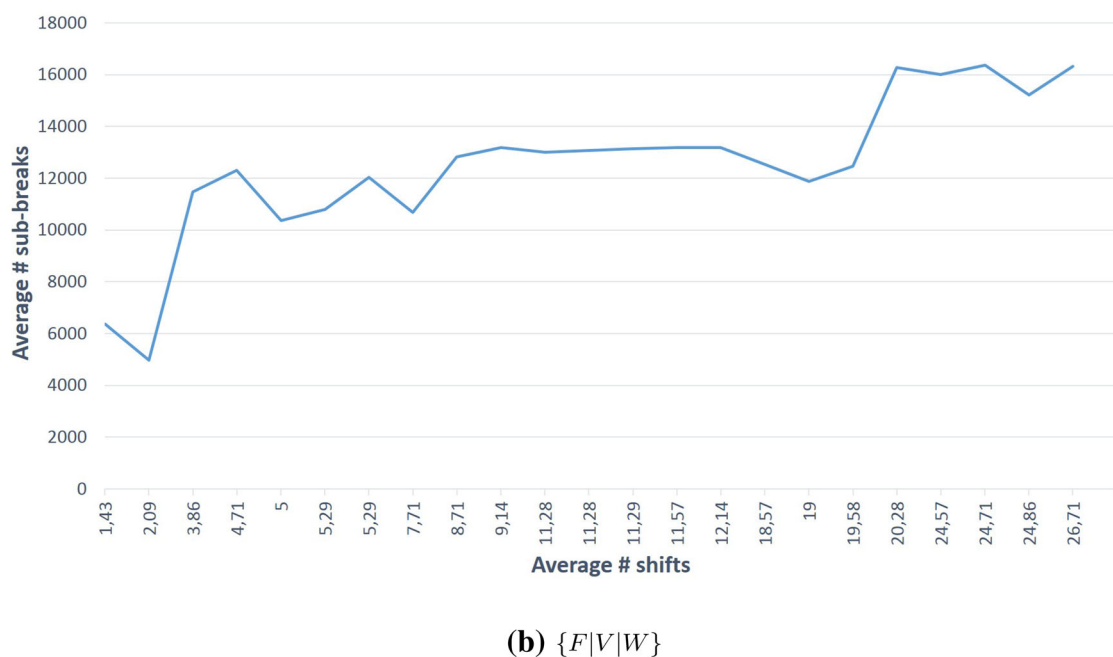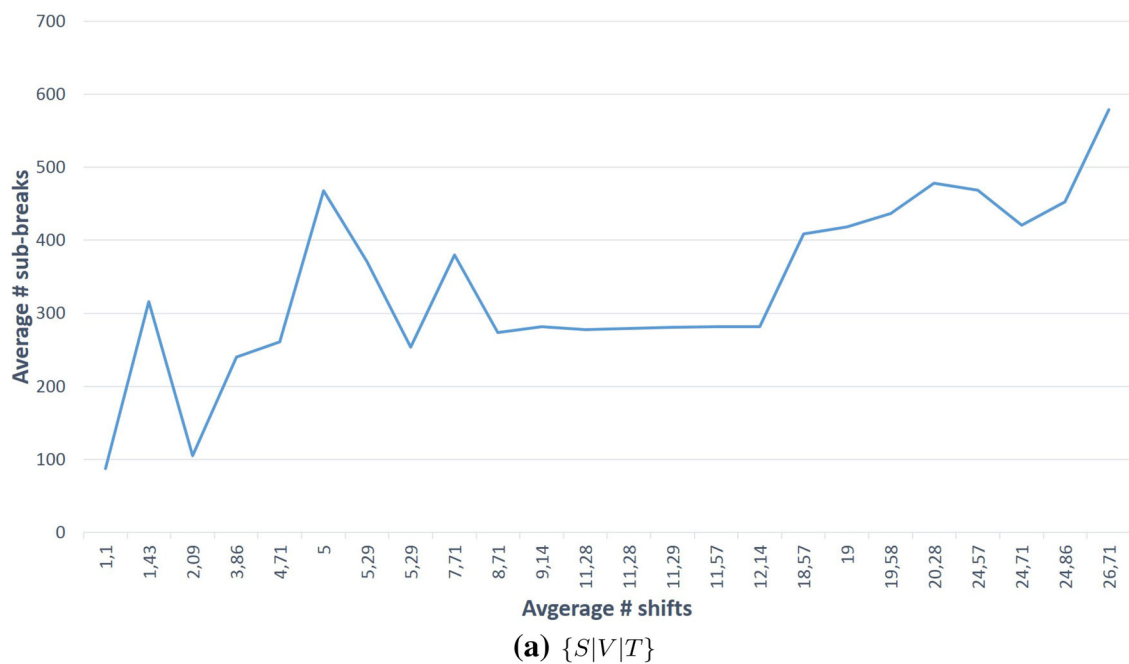
Fig. 4 Number of sub-breaks generated for $\{S|V|T\}$ and $\{F|V|W\}$ as the number of shifts increases

ered break regulation than for instances $\{\cdot, \text{high}, VD, \text{fix}\}$ with the same number workers.

The results in Table 11 reveal that as the workforce increases so does the runtime of Algorithm 1 when flexible work templates are used. Also, the type of demand curve is seen to affect runtime; instances with higher fluctuating demand take longer to converge. The number of shift types depends on the workforce size and on the work template type. For flexible work templates, more individual tours are generated to cover the demand. Of course, it is no surprise that the greater the flexibility available to generate shifts, the greater the number shifts and, hence, sub-breaks. Similar to the hypothetical example in Fig. 2, the number of sub-breaks increases geometrically with the flexibility of the break regulations. We also observed that as soon as a distinct number of shifts is used as a basis for the generation of sub-breaks,

increasing the number of shifts beyond that threshold leads to a decrease in the number of newly generated sub-breaks. For example, for break regulations $\{S|V|T\}$ and $\{F|V|W\}$ the number of newly generated sub-breaks only increases slightly when 5 or more shifts are used (see Figure 4). A similar effect can be observed for the other three break regulations in Table 11. Hence, for any break regulation there exist a bounded set of shifts leading to a bounded set of all possible sub-breaks.

In Table 12, the three BAP formulations are compared: Implicit BAP I given in Sect. 5.2.3, Implicit BAP II given in "Appendix C", and Implicit BAP III given in "Appendix D", for break regulation $\{F|V|W\}$. The results of the comparison for break regulation $\{S|V|T\}$ can be found in Table 18 of "Appendix F". Column "comp (s)" reports the running time of CPLEX in seconds for each BAP formulation summed over the 7 planning days. For Implicit BAP I and II, it is necessary to derive the sets and parameters accompanying the models, given the set of shifts for each day, e.g., for Implicit BAP formulation I, we need sub-break set $\mathcal{B}$ and parameters $h_{s,q}$. The time in seconds for all the pre-processing summed over the 7 planning days is reported in column "pre (s)." The average number of variables and constraints for each BAP formulation per day is given in columns "#var" and "#constr," respectively. If the fields in the table are empty, the corresponding BAP formulation could not find the optimal solution within the allowed time limit of 30 min (1800 seconds).

All problem instances with break regulation $\{S|V|T\}$ could be solved for all of three BAP formulations (see Table 18). Comparing the total of runtime plus pre-processing time, Implicit BAP I and II slightly outperformed BAP III. The reason may be found in the increased number of variables in BAP III. However, when considering the results for the most flexible break regulation $\{F|V|T\}$ in Table 12, Implicit BAP II performed worse than its other two competitors and is only solvable for small instances with less than 100 workers and fixed work templates. For a larger workforce, or for flexible work templates, the large number of sub-breaks ($> 10,000$ sub-breaks, see Table 11), leads to more than $500,000$ variables in Implicit BAP formulation II, which makes the problem intractable. However, not all the instances associated with Implicit BAP III could be solved either due to the huge number of variables and constraints. On the other hand, for almost all instances with up to 100 workers, the fastest results were obtained with newly introduced formulation Implicit BAP III.

In addition to the above comparisons of the computational tractability of the different formulations, Table 14 highlights the objective function deviation in % of the four BAP regulations $\{S|X|T\}$, $\{S|V|T\}$, $\{M|V|T\}$ and $\{F|V|T\}$ from the most flexible break regulation $\{F|V|W\}$. While the gaps between break regulations $\{S|V|T\}$ and $\{F|V|W\}$

are rather large with deviations up to 17.61% (see instance $\{50,$ low, VD, fix$\}$), break regulations with fractional breaks and time windows show gaps of at most 3.56% (see instance $\{100,$ high, VD, fix$\}$).

The improvements from models with less flexibility to models with more flexibility are highlighted in Table 13. For example, when break regulation $\{M|V|T\}$ is used, undercoverage decreases by 4.11%, on average, compared to the results obtained with break regulation $\{S|V|T\}$ (see row "Fix and Flex" in Table 13 which considers all templates in Tables 5 and 6). If we restrict the benefits of increased break flexibility to instances with fixed work templates only, then we obtain average improvements of 3.19%, 0.34%, 0.22% and 0.24%, respectively (see Table 13 in row "Fix"). Hence, the greatest average benefits that result from increasing break flexibility are realized for the scenarios associated with fixed work templates and variable demand (Table 14).

One possible explanation for this result stems from the fact that flexible work templates do a better job at covering demand than fixed work templates in the first step of the MIP-heuristic. For the high-demand scenarios, then, there is likely to be more uncovered demand when fixed work templates are used, so there is more opportunity to take advantage of break flexibility in the second step.

7 Discussion and conclusions

The modeling and analysis in this paper addresses several unexplored aspects of tour scheduling. The first was the idea of generating tours based on individual work templates, thereby allowing planners to take into account the availability, skill level, and overtime budget of each employee. For a wide range of service organizations, reducing labor costs by exploiting shift and break flexibility can provide a critical competitive advantage. For greater insight into the different break regulations discussed in the literature, we introduced a three-field classification scheme for the various models. An accompanying complexity analysis showed that all but the simplest of problems are strongly NP-hard. Our computations confirmed that as break flexibility increases, the corresponding break assignment problems become exponentially more difficult to solve.

Allowing variability in break length, fractionable breaks, and the use of workstretch durations in contrast to predefined break time windows was seen to reduce demand undercoverage and labor cost. The best improvement, though, was obtained by changing from single to multiple breaks per shift. A higher degree of break flexibility turned out to be especially valuable when shift regulations are rather rigid. Also, when the demand fluctuates greatly over the day, break flexibility can compensate for inflexible shift regulations and hence represents an alternative to loosening them up.

Table 12 Comparison of implicit BAP formulations for break regulation $\{F|V|W\}$

Instance	IBAP I				IBAP II				IBAP IIi		
	Comp (s)	Pre (s)	#Var	#Constr	Comp (s)	Pre (s)	#Var	#Constr	Time (s)	#Var	#Constr
{10, low, SD, fix}	3.9	1.49	7008	3173.14	56.49	41.42	87,178.29	4,160.29	1.17	790.29	2,823.43
{50, low, SD, fix}	14.18	63.43	14,689.71	4096.29					17.95	12,573.43	2,040.29
{100, low, SD, fix}	20.28	146.87	15,546.86	4214.57	512.45		516,442.3	11,948.29	25.26	23,994.86	3973.14
{10, low, SD, flex}	5.52	2.84	9568.86	3761.14					1.02	3847.71	920.29
{50, low, SD, flex}	11	8.18	13,602.29	4202					3.2	9222	1,786
{100, low,SD, flex}	30.36	8.37	14,521.71	4590.43					50.23	37,453.23	5433.33
{10, low, VD, fix}	5.63	10.16	7726.86	3311.14	3.19	11.09	52,421.86	3690.43	0.56	2823.43	790.29
{50, low, VD, fix}	31.39	749.23	16,544.29	5239.71					52.13	12,573.43	2040.29
{100, low,VD, fix}	43.1	791.62	16,495.43	5,121.43	125.23	2435.46	516,442.3	11,948.29	34.5	23,994.86	3504.57
{10, low, VD, flex}	21.78	275.38	14,050.86	4293.43					1.42	4143.71	962.57
{50, low, VD, flex}	81.2	1401	19,912.86	7032.43					7.6	1,809.56	38,201.5
{100, low, VD, flex}	70.445	43.25	19,756	6330.86					18.29	15,996	2399.43
{200, high, SD, fix}	18.93	120.32	15,274.86	4293.43					60.01	46,837.71	7,370.29
{250, high, SD, fix}	27.23	363.83	16,138	4766.57							
{300, high, SD, fix}	26.24	393.88	16,496.29	4825.71					2,109.57	69,680.57	10,767.43
{200, high, SD, flex}	43.5	1182.73	15,973.43	6329.14					157	46,837.71	7370.29
{250, high, SD, flex}	46.10	643.39	16,074.14	6074.86							
{300, high, SD, flex}	90.01	295.13	15,973.43	6329.14							
{200, high, VD, fix}	33.18	621.77	16,535.14	5160.86							
{250, high, VD, fix}	33.62	585.01	16,354.86	5121.43							
{300, high, VD, fix}	32.4	598.46	16,425.14	5121.43							
{200, high, VD, flex}	91.87	1580.92	19,558.14	6628.14							
{250, high, VD, flex}	82.06	1109.23	18,782.57	6541.29							
{300, high, VD, flex}	76.21	815.15	19,728.29	6689							

Table 13 Average benefits from break flexibility

Work template	Benefits (%)			
	$\{S|V|T\} \mapsto \{M|V|T\}$	$\{M|X|T\} \mapsto \{M|V|T\}$	$\{M|V|T\} \mapsto \{F|V|T\}$	$\{F|V|T\} \mapsto \{F|V|W\}$
Fix and flex	4.11	0.48	0.38	0.42
Flex	0.92	0.13	0.15	0.17
Fix	3.19	0.34	0.22	0.24

Table 14 Comparison of break flexibility with respect to $\{F|V|W\}$

Instance	Break regulation			
	$\{S|V|T\}$	$\{M|X|T\}$	$\{M|V|T\}$	$\{F|V|T\}$
{10, low, SD, fix}	0.15	0.07	0.07	0.00
{50, low, SD, fix}	5.44	1.23	0.00	0.00
{100, low, SD, fix}	0.78	0.78	0.70	0.25
{10, low, SD, flex}	0.00	0.00	0.00	0.00
{50, low, SD, flex}	6.02	3.34	0.98	0.97
{100, low, SD, flex}	1.71	0.41	0.20	0.17
{10, low, VD, fix}	17.61	3.02	1.62	0.00
{50, low, VD, fix}	2.85	1.16	0.45	0.14
{100, low, VD, fix}	4.96	3.56	3.04	3.04
{10, low, VD, flex}	1.09	0.12	0.02	0.00
{50, low, VD, flex}	0.06	0.00	0.00	0.00
{100, low, VD, flex}	2.88	1.22	1.2	0.43
{200, high, SD, fix}	13.04	0.99	0.15	0.04
{250, high, SD, fix}	16.68	1.25	0.69	0.31
{300, high, SD, fix}	8.96	0.95	0.19	0.10
{200, high, SD, flex}	11.43	1.32	1.13	0.01
{250, high, SD, flex}	3.37	1.78	1.65	1.23
{300, high, SD, flex}	0.56	0.02	0.02	0.02
{200, high, VD, fix}	9.62	0.37	0.03	0.00
{250, high, VD, fix}	11.39	3.65	2.07	0.67
{300, high, VD, fix}	4.59	2.56	2.27	1.35
{200, high, VD, flex}	0.25	0.20	0.20	0.01
{250, high, VD, flex}	2.11	1.43	1.43	1.02
{300, high, VD, flex}	3.83	1.23	1.02	0.42

a minute. For all larger instances with 200 to 300 workers, only the formulation based on Bechtold and Jacobs (1990) led to solution times of less than 30 min.

Currently, no procedure exists for solving the integrated problem to optimality in reasonable time. Its practical relevance offers ample justification for more algorithmic development. In the future, we plan to explore exact decomposition approaches to tackle the ThSP and the flexible BAPs in an integrated way. The study of the BAP alone is also of interest, especially for intraday schedule adjustments. During the course of the day, shifts cannot in general be adjusted when a the demand profile changes, but it may be possible to reassign the breaks. In this regard, we are now designing a rolling horizon framework for calibrating the value of flexible breaks in short-term disruption scenarios.

Because a compact formulation of the tour scheduling problem could not be solved for large-scale instances, a MIP-decomposition heuristic was proposed that separates the break assignments from tour scheduling. The approach led to promising results with respect to runtimes and solution quality for all instances with up to 300 workers. In contrast, the compact formulation could only solve those instances with up to 50 workers and little shift flexibility.

To solve the break assignment problem, three implicit formulations were investigated. BAP III in "Appendix D" has the advantage of not requiring any pre- or post-processing and could solve all instances with up to 100 workers in less than

Appendix A: Proofs of complexity results

Proof of Proposition 1 First, let us assume that we have only one skill level for the workforce and the more general case in which breaks can be more than one period. Consider shift s and its corresponding set of feasible break patterns $\mathcal{B}(s)$. By examining constraint (2), we see that each column possesses the consecutive 1's property. Let $A^1 \equiv (a_{i,j}^1)$ be the matrix associated with the first shift in (2). Such a matrix is TU (Nemhauser and Wolsey 1988). As an example, consider a shift of 10 periods that requires a break of either 3 or 4 periods starting in either periods 3, 4 or 5, as shown in Figure 5.

The row of 1's below A^1 in this figure corresponds to constraint (3); the second matrix, call it Y, on the right corresponds to the y_t^+ or y_t^- variables. Although constraints (3) destroy the consecutive 1's property in general, when the break length for each shift is one period the portion of the A-matrix in (2) associated with each shift has a single 1 in each column. This entry appears in a row in which a break is permissible (this is not the case shown in Figure 5). The remaining row entries in the column are 0. Thus each column in the full matrix has at most two elements that are either 1 or -1, and, in light of the special structure of constraints (3), we see that it satisfies the sufficiency condition for TU. That is, there exits a partition of the set M of rows into two sets M_1 and M_2 such each column j containing two nonzero elements satisfies $\sum_{i \in M_1} a_{i,j} - \sum_{i \in M_2} a_{i,j} = 0$. Note that the Y-matrix can be written as $Y = [I, -I]$, where I is the $|T| \times |T|$ identity matrix, so each column contains only a single nonzero element. □

Proof of Corollary 1 Again, let A^1 be the matrix associated with the first shift in (2) and let A^2 be the matrix associated with constraints (3), also for the first shift. As an example, Figure 6 depicts a portion of the constraint matrix for a shift with 7 periods that requires two breaks of one period each. For feasibility, the first must start in either period 2 or 3 and the second in either period 5 or 6. As indicated in the figure, when the length of each sub-break is equal to one period, each column of A^1 contains a single entry of 1 in the row where the sub-break exists with the remaining entries being 0. In a solution, one of the first two columns and one of the second two columns must be selected.

Now, when adding time windows to regulate sub-break starting times, there is no interdependency between the possible starting times of different sub-breaks, i.e., the starting time of the i-th sub-break is independent of the starting time of the j-th sub-break for all i, $j \in [1, B^{\max}]$. Each column of A^2 also contains a single entry of 1 or -1 with the other entries being 0 due to the fact that each column of A^1 and A^2 corresponds to a single sub-break. Therefore, as in the proof of Proposition 1, the sufficiency condition for TU still holds in the case where the sub-break length is one period for each shift. □

Proof of Corollary 2 Following the proof of Corollary 1, let A^1 be the matrix associated with the first shift in (2) and let A^2 be the matrix associated with constraints (3), also for the first shift. As an example, Figure 7 shows a portion of the constraint matrix for a shift with 5 periods that requires one break. The break must start in either period 2, 3 or 4 of the shift and lasts for one period. We assume that there are two skill levels q_1 and q_2 and that the worker associated with the first shift is scheduled to work a job requiring skill q_1 in the second period and q_2 in the third and fourth periods, respectively. In periods 1 and 5, he is either working a job

requiring q_1 or q_2 but, due to the time window, can not have a break. The submatrix above the horizontal line illustrates constraints (2) for q_1 and the lower submatrix the corresponding constraints for q_2. When the length of the sub-break is equal to one period, each column of A^1 contains a single entry of 1 in the row where the sub-break exists (columns 1 and 2) with the remaining entries being 0. Column 3 shows that the consecutive 1's property is lost when a sub-break incorporates more than a single skill.

Similar to the single skill case, when employing time windows to regulate sub-break starting times, there is no interdependency between the possible starting times of different sub-breaks. Therefore, each column of A^2 also contains a single entry of 1 or -1 with the other entries being 0 due to the fact that each column of A^1 and A^2 corresponds to a single sub-break. Now again, similar to the proof of Proposition 1, the sufficiency condition for TU still holds in the case where the sub-break length is one period for each shift. □

Proof of Theorem 1 We will show that an instance of the single machine scheduling problem with release times and deadlines, which Garey and Johnson (1979) state is strongly NP-hard, can be polynomially transformed into an instance of BAP$\{S|X|T\}$.

INSTANCE: Set $\mathcal{J}$ of jobs, for each job $j \in \mathcal{J}$, a processing time $p_j \in \mathbb{Z}_+$, a release time $r_j \in \mathbb{Z}_+$, and a deadline $e_j \in \mathbb{Z}_+$.

QUESTION: Is there a single machine schedule for $\mathcal{J}$ that satisfies the release time constraints and meets all the deadlines?

Given an instance of the single machine scheduling problem, we equate a shift with a job, so $\mathcal{S}$ is equivalent to $\mathcal{J}$. For shift $s \in \mathcal{S}$, let r_s be the period in which the break can start (release time), e_s the last period in which the break is allowed (deadline), and p_s the number of periods of the break (processing time). Also, let the surplus of coverage in each period be 1.

Claim: There exists a feasible assignment of breaks to shifts such that there are no manpower shortages in any period if and only if there is a feasible schedule of jobs for the single machine problem.

The condition of no manpower shortage in any period means that no two breaks overlap. This is equivalent to a schedule on a single machine in which each job is processed between its release time and deadline without conflicting with any other job. Given that the above transformation is linear in the number of shifts, and that any candidate solution to the BAP can be checked for feasibility in polynomial time, the statement of the theorem follows. □

Proof of Proposition 2 For the BAP$\{F|V|T\}$ with n distinct time windows, assume that the overall break length ranges between $\Delta^{\min BD}$ and $\Delta^{\max BD}$ periods and that the min-

Fig. 5 Constraint matrix for BAP$\{S|X|T\}$ with one skill level

$$
(a_{i,j}^1) = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \cdots
\begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & & & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & & & & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & & & \cdots & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & & & & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{pmatrix}
$$

$$
\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \cdots \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}
$$

Fig. 6 Constraint matrix for BAP$\{M|X|T\}$ with two break windows and one skill level

$$
(a_{i,j}^1) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \cdots
\begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & & & & \\ 0 & 0 & 0 & 0 & & & & & \\ 0 & 0 & 0 & 0 & & & & & \\ 0 & 0 & 0 & 0 & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{pmatrix}
$$

$$
(a_{i,j}^2) = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} \cdots \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}
$$

Fig. 7 Constraint matrix for BAP$\{M|X|T\}$ with two skill levels

$$
(a_{i,j}^1) = \left(\begin{array}{ccc} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{array} \right) \cdots
\left(\begin{array}{ccccccccc} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & & & & \\ 0 & 0 & 0 & 0 & & & & & \\ 0 & 0 & 0 & 0 & & & & & \\ \hline 0 & 0 & 0 & 0 & & & & & \\ 0 & 0 & 0 & 0 & & & & & \\ 0 & 0 & 0 & 0 & & & & & \\ 0 & 0 & 0 & 0 & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{array} \right)
$$

$$
(a_{i,j}^2) = \begin{pmatrix} 1 & 1 & 1 & 1 \end{pmatrix} \cdots \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}
$$

imum and maximum number of sub-breaks are between $B^{\min}$ and $B^{\max}$. With each time window $\left[t_r^s, t_r^e\right], r \in \{1, \ldots, n\}$, we associate a minimum break length of $d_r^{\min}$ and a maximum break length of $d_r^{\max}$. Accordingly, for each $i \in \{1, \ldots, B^{\max}\}$ the i-th sub-break can start in the (i)-th,...,$(n - B^{\min} + i)$-th time window and the length of the i-th sub-break is between $lb_i^{\mathrm{bl}} = min_{r \in \{i, \ldots, n - B^{\min} + i\}}(d_r^{\min})$ and $ub_i^{\mathrm{bl}} = max_{r \in \{i, \ldots, n - B^{\min} + i\}}(d_r^{\max})$ periods.

For the BAP$\{F|V|W\}$, we set the range for the overall break length as well as the minimum and maximum number of sub-breaks equal to those of the BAP$\{F|V|T\}$. Further, for each sub-break $i \in \{1, \ldots, B^{\max}\}$, we set the minimum and the maximum break length equal to lb_i^{bl} and ub_i^{bl}, respectively. To proof our statement, it is sufficient to set the minimum and maximum workstretch after each sub-break

equal to 0 and the shift length, respectively. Based on the assumptions, BAP$\{F|V|W\}$ includes all the breaks of the BAP$\{F|V|T\}$ but not vice versa.

Now, place a cost of $M > 0$ on each sub-break in the BAP$\{F|V|W\}$ that is not feasible to BAP$\{F|V|T\}$. The procedure can be done in polynomial time as for each shift, the overall number of breaks in the BAP$\{F|V|W\}$ is bounded by $\prod_{i \in \{1, \ldots, B^{\max}\}}(ub_i^{\mathrm{bl}} - lb_i^{\mathrm{bl}}) \cdot \Delta^{\max S}$. Given both models, we claim that the BAP$\{F|V|T\}$ has a solution with no uncovered periods if and only if the BAP$\{F|V|W\}$ has a solution with no uncovered periods and a cost of 0. First suppose that the BAP$\{F|V|T\}$ has a solution with no uncovered periods. Then the same break patterns can be used for the BAP$\{F|V|W\}$ to obtain the same solution. This proves the "if" part of the statement. Now suppose that the opti-

mal solution to the BAP$\{F|V|W\}$ has a cost greater than 0. By implication, this solution contains sub-breaks that have a cost M and hence is not feasible to the BAP$\{F|V|T\}$. Therefore, BAP$\{F|V|T\}$ does not have a solution with uncovered periods. This proves the "only if" part of the statement. Now observe that the BAP$\{F|V|T\}$ does not include the BAP$\{F|V|W\}$ since the break windows of two or more consecutive sub-breaks can overlap. Consider a BAP$\{F|V|W\}$ with two sub-breaks. Let Δ^{minBF} and Δ^{maxBF} be the minimum and maximum workstretch duration before the first sub-break, respectively. Further, let Δ^{durB} be the minimum duration of the first sub-break and let Δ^{minBW} be the minimum workstretch duration between the first and second sub-breaks. When $\Delta^{\mathrm{minBF}} + \Delta^{\mathrm{durB}} + \Delta^{\mathrm{minBW}} < \Delta^{\mathrm{maxFB}}$, the break windows of the first and second sub-breaks overlap. In general, in the presence of workstretch durations, the break window for a particular sub-break can only be derived after the realization of the other sub-breaks. $\qquad\square$

Appendix B: Compact mixed-integer program

The ground handler staffing problem (GHSP) is to minimize the total cost of the workforce plus the cost for undercoverage for a given planning horizon. The compact MIP (CMIP) below determines shifts with fractionable breaks for each day in the week and creates a duty roster for each worker such that as much demand as possible is covered in each period for the different skill levels.

Variables

$b^{\mathrm{start}}_{r,w,d,t} = 1$, if sub-break r for worker w starts in period t on day d, 0 otherwise

$b^{\mathrm{end}}_{r,w,d,t} = 1$, if sub-break r for worker w ends in period t on day d, 0 otherwise

$b^{\mathrm{last}}_{w,d,t} = 1$, if the last sub-break for worker w ends in period t on day d, 0 otherwise

CMIP

$$\text{minimize} \sum_{q \in \mathcal{Q}} \sum_{d \in \mathcal{D}} \sum_{t \in \mathcal{T}} \left(\sum_{w \in \mathcal{W}} c^{\mathrm{work}} \cdot z_{w,q,d,t} + c^{\mathrm{uc}} \cdot y^{\mathrm{uc}}_{q,d,t} \right) \tag{38}$$

subject to $(10) - -(21)$

$$\sum_{q' \in \mathcal{Q}: q' \geq q} \sum_{w \in \mathcal{W}(q')} z_{w,q,d,t} + y^+_{q,d,t} - y^-_{q,d,t} = K_{q,d,t}$$
$$\forall d \in \mathcal{D}, t \in \mathcal{T}, q \in \mathcal{Q} \tag{39}$$

$$\sum_{m \in \mathcal{M}(d,p_w)} \sum_{l \in \mathcal{T}^{\mathrm{shS}}: l \leq t} \big(y_{m,w,d,l} -$$

$$\left(\sum_{r=1}^{B^{\max}} \left(\sum_{l \in \mathcal{T}^{\mathrm{brS}}(r): l \geq t} b^{\mathrm{start}}_{r,w,d,l} - \sum_{l \in \mathcal{T}^{\mathrm{brE}}(r): l \geq t} b^{\mathrm{end}}_{r,w,d,l} \right) + \right. \tag{40}$$

$$\left. \sum_{l \in \mathcal{T}^{\mathrm{shE}}: l \geq t} y^{\mathrm{end}}_{w,d,l} \right) = z_{w,q,d,t}$$

$$\forall w \in \mathcal{W}, d \in \mathcal{D}(p_w), t \in \mathcal{T}$$

$$\sum_{q \in \mathcal{Q}} z_{w,q,d,t} \leq 1 \quad \forall w \in \mathcal{W}, d \in \mathcal{D}(p_w), t \in \mathcal{T} \tag{41}$$

Break regulations

$$\sum_{r=1}^{B^{\max}} \sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} b^{\mathrm{start}}_{r,w,d,t} \geq \sum_{t \in \mathcal{T}^{\mathrm{brS}}(1)} b_{1,w,d,t} \cdot B^{\min}$$
$$\forall w \in \mathcal{W}, d \in \mathcal{D}(p_w) \tag{42}$$

$$\Delta^{\mathrm{maxBF}} \geq \left(\sum_{t \in \mathcal{T}^{\mathrm{brS}}(1)} t \cdot b^{\mathrm{start}}_{1,w,d,t} - \right.$$

$$\left. \sum_{m \in \mathcal{M}(d,p_w)} \sum_{t \in \mathcal{T}^{\mathrm{shS}}(m)} \left(t - \Delta^{\mathrm{minBF}} \right) \cdot y_{m,w,d,t} \right) \geq 0$$

$$\forall w \in \mathcal{W}, d \in \mathcal{D}(p_w) \tag{43}$$

$$\Delta^{\mathrm{maxBD}}_r \geq \sum_{t \in \mathcal{T}^{\mathrm{brE}}(r)} t \cdot b^{\mathrm{end}}_{r,w,d,t} - \sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} t \cdot b^{\mathrm{start}}_{r,w,d,t} \geq$$

$$\Delta^{\mathrm{minBD}}_r \cdot \sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} b^{\mathrm{start}}_{r,w,d,t} \quad \forall w \in \mathcal{W}, d \in \mathcal{D}(p_w), 1 \leq r \leq B^{\max} \tag{44}$$

$$\Delta^{\mathrm{maxBD}} \geq \sum_{r=1}^{B^{\max}_p} \left(\sum_{t \in \mathcal{T}^{\mathrm{brE}}(r)} t \cdot b^{\mathrm{end}}_{r,w,d,t} - \sum_{\mathcal{T}^{\mathrm{brS}}(r)} t \cdot b^{\mathrm{start}}_{r,w,d,t} \right) \geq$$

$$\Delta^{\mathrm{minBD}} \cdot \sum_{m \in \mathcal{M}(d,p_w)} \sum_{t \in \mathcal{T}^{\mathrm{shS}}(m)} y_{m,w,d,t} \quad \forall w \in \mathcal{W}, d \in \mathcal{D}(p_w) \tag{45}$$

$$\Delta^{\mathrm{maxBW}}_r \geq \sum_{t \in \mathcal{T}^{\mathrm{brS}}_m(r)} t \cdot b^{\mathrm{start}}_{r+1,w,d,t} -$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brE}}(r)} t \cdot b^{\mathrm{end}}_{r,w,d,t} \geq \Delta^{\mathrm{minBW}}_r \cdot \sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} b^{\mathrm{start}}_{r+1,w,d,t}$$

$$\forall w \in \mathcal{W}, d \in \mathcal{D}(p_w),$$
$$1 \leq r < B^{\max} \tag{46}$$

$$\sum_{t \in \mathcal{T}^{\mathrm{shE}}} y^{\mathrm{end}}_{w,d,t} - \Delta^{\mathrm{minBL}} \geq \left(\sum_{t \in \mathcal{T}^{\mathrm{brE}}} t \cdot b^{\mathrm{last}}_{w,d,t} \right) \geq$$

$$\sum_{t \in \mathcal{T}^{\mathrm{shE}}} y^{\mathrm{end}}_{w,d,t} - \Delta^{\mathrm{maxBL}} \quad \forall w \in \mathcal{W}, d \in \mathcal{D} \tag{47}$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brE}}} t \cdot b^{\mathrm{last}}_{w,d,t} \geq \left(\sum_{t \in \mathcal{T}^{\mathrm{brE}}(r)} t \cdot b^{\mathrm{end}}_{r,w,d,t} \right)$$

$$\forall w \in \mathcal{W}, d \in \mathcal{D}, B^{\min} \leq r \leq B^{\max} \tag{48}$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brE}} \setminus \{0\}} (t-1) \cdot b^{\mathrm{last}}_{w,d,t} \leq \sum_{t \in \mathcal{T}^{\mathrm{brE}}(r)} t \cdot b^{\mathrm{end}}_{r,w,d,t} +$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} b^{\mathrm{start}}_{r+1,w,d,t} \cdot T \quad \forall \, w \in \mathcal{W}, d \in \mathcal{D}, B^{\min} \le r < B^{\max} \tag{49}$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brE}} \setminus \{0\}} (t-1) \cdot b^{\mathrm{last}}_{w,d,t} \le \sum_{t \in \mathcal{T}^{\mathrm{brE}}(B^{\max})} t \cdot b^{\mathrm{end}}_{B^{\max},s,t} +$$

$$\left(1 - \sum_{t \in \mathcal{T}^{\mathrm{brS}}(B^{\max})} b^{\mathrm{start}}_{B^{\max},w,d,t} \right) \cdot T \quad \forall \, w \in \mathcal{W}, d \in \mathcal{D} \tag{50}$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brE}} \setminus \{0\}} b^{\mathrm{last}}_{w,d,t} \le 1 \quad \forall \, w \in \mathcal{W}, d \in \mathcal{D} \tag{51}$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brS}}(r-1)} b^{\mathrm{start}}_{r-1,w,d,t} \ge \sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} b^{\mathrm{start}}_{r,w,d,t}$$

$$\forall \, w \in \mathcal{W}, d \in \mathcal{D}(p_w), 1 < r \le B^{\max} \tag{52}$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} b^{\mathrm{start}}_{r,w,d,t} \le 1 \,\wedge\, \sum_{t \in \mathcal{T}^{\mathrm{brE}}(r)} b^{\mathrm{end}}_{r,w,d,t} \le 1$$

$$\forall \, w \in \mathcal{W}, d \in \mathcal{D}(p_w), 1 \le r \le B^{\max} \tag{53}$$

$$\sum_{r=1}^{B^{\max}} \sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} b^{\mathrm{start}}_{r,w,d,t} = \sum_{r=1}^{B^{\max}} \sum_{t \in \mathcal{T}^{\mathrm{brE}}(r)} b^{\mathrm{end}}_{r,w,d,t}$$

$$\forall \, w \in \mathcal{W}, d \in \mathcal{D}(p_w) \tag{54}$$

$$z_{w,q,d,t} \in \{0,1\} \quad \forall \, w \in \mathcal{W}, d \in \mathcal{D}(p_w), t \in \mathcal{T} \tag{55}$$

$$b^{\mathrm{start}}_{r,w,d,t}, b^{\mathrm{end}}_{r,w,d,t'}, b^{\mathrm{last}}_{w,d,t'} \in \{0,1\}$$

$$\forall \, w \in \mathcal{W}, d \in \mathcal{D}(p_w), t \in \mathcal{T}^{\mathrm{brS}}, t' \in \mathcal{T}^{\mathrm{brE}} \, 1 \le r \le B^{\max} \tag{56}$$

The objective function (38) along with the constraints (39)–(41) is similar to (7) and (7)–(9), respectively, but now all skill levels are included. Constraints (42)–(54) describe the break regulations. With respect to B1, for each worker w at least $B^{\min}$ sub-breaks per shift are set in constraints (42), while the minimum and maximum number of periods from the start of the shift to the first sub-break, as indicated in B2, are set in constraints (43)). The length of each sub-break, as restricted by B3, and the overall break length, as restricted by B4, are bounded by constraints (44) and (45), respectively. The minimal and maximal bandwidth between two sub-breaks, B5, is set in constraints (46). The distance of the last sub-break to the end of the shift, B6, is modeled in constraints (47). The decision variables $b^{\mathrm{last}}_{w,t,d}$, which control the latest ending time over all sub-breaks assigned to worker w on day d are determined by (48)–(51).

The last set of constraints (52)–(54) define general modeling requirements. Logically, the r-th sub-break can only be started if its immediate predecessor sub-break $r-1$ has been started [constraints (52)]. Also, the unique starting period of each sub-break is defined in constraints (53) and, due to constraints (54), a sub-break can only end if it has been started.

If we consider breaks with time windows in the GHSP, i.e., BAP$\{\cdot | \cdot | T\}$, the above model has to be slightly modified. Let Δ^{minTW} and Δ^{maxTW} be the minimal and maximal number of periods of the r-th sub-break, respectively, for

Table 15 Sets used in implicit break assignment model IBAP2

Sets	Description
$\mathcal{J}_{s,q}$	Set of shift profiles associated with shift s and skill q
$\mathcal{K}_{j(r)}$	Set of sub-breaks admissible as r'th sub-break in shift profile j
$\mathcal{Q}$	Set of skills

$r > 1$ from the shift start. Then we replace the two-sided constraints (46) with

$$\Delta^{\mathrm{maxTW}}_r \ge \sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} t \cdot b^{\mathrm{start}}_{r,w,d,t} - \sum_{m \in \mathcal{M}(d,p_w)} \sum_{t \in \mathcal{T}^{\mathrm{shS}}(m)} t \tag{46'}$$

$$\cdot \, y_{m,w,d,t} \Delta^{\mathrm{minTW}}_r$$

$$\ge \sum_{m \in \mathcal{M}(d,p_w)} \Delta^{\mathrm{minTW}}_r \cdot y_{m,w,d,t}$$

$$\forall \, w \in \mathcal{W}, d \in \mathcal{D}(p_w), 1 < r \le B^{\max}$$

which is analogous to constraints (43) for the first sub-breaks.

Appendix C: Implicit BAP formulation based on Aykin's approach

For shift scheduling, Rekik et al. (2010) present a model with fractionable breaks based on the implicit formulation for multiple breaks first introduced by Aykin (1996). In the following, we show how their approach can be adapted for the tour scheduling problem with a hierarchical workforce as was discussed in Sect. 5.2. The additional sets required for the model are listed in Table 15.

We make use of the definitions of shift profiles and break profiles in Sect. 5.2 and similarly, define a sub-break $k \in \mathcal{K}$ for each break profile, each position in the break profile, each possible starting time, and each skill level. Also, we need a unique sub-break for each skill level since we have to consider the skill of the worker that is taking the break. Instead of assigning sub-breaks explicitly to shifts, a variable $X_{j,k}$ is defined that stores how often sub-break $k \in \mathcal{K}$ is assigned to shift profile $j \in J$.

To obtain a feasible solution with regard to workstretch durations, the feasibility of transportation problems between successive breaks has to be ensured. The following set is used to construct the problems.

$$\mathcal{K}_{j(r)} = set\ of\ sub\text{-}breaks\ admissible\ as\ r\text{-}th\ sub$$
$$\text{-}break\ in\ shift\ profile\ j$$

Based on the break starting times, a totally ordered relationship $\prec$ on $\mathcal{K}_{j(r)}$ is defined. Then, a transportation problem $T(\mathcal{K}_{j(r)}, \mathcal{K}_{j(r+1)})$ from the nodes in $\mathcal{K}_{j(r)}$ to the nodes in $\mathcal{K}_{j(r+1)}$ can be established for each $j \in \mathcal{J}$ and each

$r \in 1, \ldots, B_j - 1$. Nodes $k_1 \in \mathcal{K}_{j(r)}$ and $k_2 \in \mathcal{K}_{j(r+1)}$ are connected if the workstretch duration between their corresponding breaks is within the interval $[\delta^{minBW}, \delta^{maxBW}]$. The feasibility of the transportation problem ensures that there are enough successor breaks for all sub-breaks k that are assigned to shift profile j and is guaranteed by a set of forward and backward constraints based on the principles described in Sect. 5.2.

Also, because there are no pairs of nodes $k_1, k_2 \in \mathcal{K}_{j(r)}$ for which $\mathcal{K}_{k_1}^{suc} \subset \mathcal{K}_{k_2}^{suc}$, there is no extraordinary overlap.

In the formulation, we make use of the following additional parameters and variables: **Parameters**

B_j = number of sub-breaks in shift j

Variables

$X_{j,k}$ = number of workers assigned to shift j that are given sub-break k

Implicit BAP formulation II (IBAP2)

$$\text{Minimize} \sum_{q \in \mathcal{Q}} \sum_{t \in \mathcal{T}} c^{uc} \cdot y_{q,t}^- - \sum_{j \in \mathcal{J}} \sum_{r \in \{1,\ldots,B_j\}} \sum_{k \in \mathcal{K}_{j(r)}} c^{work}$$
$$\cdot d_k \cdot X_{j,k} \tag{57}$$

subject to

$$\sum_{\hat{q} \geq q} P_{\hat{q},q,t} + y_{q,t}^+ - y_{q,t}^- = D_{q,d,t} \quad \forall q \in \mathcal{Q}, t \in \mathcal{T} \tag{58}$$

$$\sum_{j \in \mathcal{J}_{s,q}} S_j = h_{s,q} \quad \forall s \in \mathcal{S}, q \in \mathcal{Q} \tag{59}$$

$$\sum_{k \in \mathcal{K}_{j(r)}} X_{j,k} - S_j = 0 \quad \forall j \in \mathcal{J}, r \in \{1,\ldots,B_j\} \tag{60}$$

$$P_{\hat{q},q,t} \leq l_{\hat{q},q,t} \quad \forall \hat{q}, q \in \mathcal{Q} : q \leq \hat{q}, t \in \mathcal{T} \tag{61}$$

$$\sum_{q \leq \hat{q}} P_{\hat{q},q,t} = \sum_{j \in \mathcal{J}_{\hat{q}}} \sum_{r \in \{1,\ldots,B_j\}} \sum_{k \in \mathcal{K}_{j(r)}} X_{j,k} \cdot \rho_{t,k} \quad \forall \hat{q} \in \mathcal{Q}, t \in \mathcal{T} \tag{62}$$

$$\sum_{k' \in \mathcal{K}_{j(r+1)}^F(k)} X_{j,k'} - \sum_{k' \in \mathcal{K}_{j(r)}^F(k)} X_{j,k'} \geq 0$$
$$\forall j \in \mathcal{J}, r \in \{1,\ldots,B_j-1\},$$
$$k \in \mathcal{K}_{j(r+1)}^e \setminus \left\{ k_{j(r+1)}^e \right\} \tag{63}$$

$$\sum_{k' \in \mathcal{K}_{j(r+1)}^B(k)} X_{j,k'} - \sum_{k' \in \mathcal{K}_{j(r)}^B(k)} X_{j,k'} \geq 0$$
$$\forall j \in \mathcal{J}, r \in \{1,\ldots,B_j-1\},$$
$$k \in \mathcal{K}_{j(r+1)}^s \setminus \left\{ k_{j(r+1)}^s \right\} \tag{64}$$

$$S_j, w_{w,t}, P_{\hat{q},q,t} \in \mathbb{Z}_+ \quad \forall j \in \mathcal{J}, \hat{q}, q \in \mathcal{Q} : q \leq \hat{q}, t \in \mathcal{T} \tag{65}$$

$$X_{j,k} \in \mathbb{Z}_+ \quad \forall j \in \mathcal{J}, k \in \bigcup_{r \in \{1,\ldots,B_j\}} \mathcal{K}_{j(r)} \tag{66}$$

Constraints (58) determine the amount of demand for skill level q on day d in period t that cannot be covered, while constraints (59) ensure that all shifts of type s for workers with skill q are connected to a shift profile j. Due to constraints (60), the number of sub-breaks for shift j in each break position $r \in \{1, \ldots, B_j\}$ corresponds to the number of shifts of type j. Further, there cannot be more workers with skill $\hat{q}$ that take a break from a job requiring skill q in period t then there are workers in the input tour plan with skill $\hat{q}$ carrying out a job in period t that requires skill q. This is enforced by (61). In addition, all breaks taken in period t by workers with skill level $\hat{q}$ must be distributed among jobs requiring at most skill $\hat{q}$. This is enforced by (62). Constraints (63) and (64), respectively, are the forward and backward constraints used to ensure that the total number of sub-breaks is correct. Equality constraints for the total sub-break supply and demand are not needed due to constraints (60). Variables are defined in (65) and (66).

Appendix D: Implicit BAP formulation III

The BAP formulation contained in CMIP is stated below using the following additional notation.
Parameters

S_s^{start} = start time of shift s
S_s^{end} = end time of shift s
C_s^{work} = cost for shift s (without breaks)

Variables

$b_{r,s,t}^{start} = 1$, if the r-th sub-break for shift s starts at time t, 0 otherwise
$b_{r,s,t}^{end} = 1$, if the r-th sub-break for shift s ends at time t, 0 otherwise
$b_{s,t}^{last} = 1$, if the last sub-break for shift s ends at time t, 0 otherwise
$a_{s,t} = 1$, if there is a break in shift s at time t, 0 otherwise
$z_{s,q,t} = 1$, if period t is a working period for shift s that covers skill level q, 0 otherwise

Implicit BAP formulation III (IBAP3)

$$\text{Minimize} \sum_{t \in \mathcal{T}} \left(\sum_{s \in \mathcal{S}} c^{work} \cdot a_{s,t} + \sum_{q \in \mathcal{Q}} c^{uc} \cdot y_{q,t}^- \right) \tag{67}$$

subject to

$$\sum_{s \in \mathcal{S}(q,t)} a_{s,t} - y_{q,t}^- + y_{q,t}^+ = D_{q,d,t} \quad \forall q \in \mathcal{Q}, t \in \mathcal{T} \tag{68}$$

$$\sum_{r=1}^{B^{\max}} \sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} b_{r,s,t}^{\mathrm{start}} \geq B^{\min} \quad \forall s \in \mathcal{S}^*(d) \tag{69}$$

$$\Delta^{\mathrm{minBF}} \leq \left(\sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} t \cdot b_{1,s,t}^{\mathrm{start}} - S_s^{\mathrm{start}} \right) \leq \Delta^{\mathrm{maxBF}} \quad \forall s \in \mathcal{S}^*(d) \tag{70}$$

$$\Delta_r^{\mathrm{minBD}} \cdot \sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} b_{r,s,t}^{\mathrm{start}} \leq \sum_{t \in \mathcal{T}^{\mathrm{brE}}(r)} t \cdot b_{r,s,t}^{\mathrm{end}} -$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} t \cdot b_{r,s,t}^{\mathrm{start}} \leq \Delta_r^{\mathrm{maxBD}} \quad \forall s \in \mathcal{S}^*(d), 1 \leq r \leq B^{\max} \tag{71}$$

$$\Delta^{\mathrm{minBD}} \leq \sum_{r=1}^{B^{\max}} \left(\sum_{t \in \mathcal{T}^{\mathrm{brE}}(r)} t \cdot b_{r,s,t}^{\mathrm{end}} - \sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} t \cdot b_{r,s,t}^{\mathrm{start}} \right)$$

$$\leq \Delta^{\mathrm{maxBD}} \quad \forall s \in \mathcal{S}^*(d) \tag{72}$$

$$\Delta_r^{\mathrm{minBW}} \cdot \sum_{t \in \mathcal{T}^{\mathrm{brS}}(r+1)} b_{r+1,s,t}^{\mathrm{start}} \leq \sum_{t \in \mathcal{T}^{\mathrm{brS}}(r+1)} t \cdot b_{r+1,s,t}^{\mathrm{start}} -$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brE}}(r)} t \cdot b_{r,s,t}^{\mathrm{end}} \leq \Delta_r^{\mathrm{maxBW}} \quad \forall s \in \mathcal{S}^*(d), 1 \leq r \leq B^{\max} - 1$$

$$\tag{73}$$

$$S_{s,d}^{\mathrm{end}} - \Delta^{\mathrm{minBL}} \geq \left(\sum_{t \in \mathcal{T}^{\mathrm{brE}}} t \cdot b_{s,t}^{\mathrm{last}} \right) \geq S_s^{\mathrm{end}} - \Delta^{\mathrm{maxBL}} \quad \forall s \in \mathcal{S}^*(d)$$

$$\tag{74}$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brE}}} t \cdot b_{s,t}^{\mathrm{last}} \geq \left(\sum_{t \in \mathcal{T}^{\mathrm{brE}}(r)} t \cdot b_{r,s,t}^{\mathrm{end}} \right) \quad \forall s \in \mathcal{S}^*(d), B^{\min} \leq r \leq B^{\max}$$

$$\tag{75}$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brE}}} (t-1) \cdot b_{s,t}^{\mathrm{last}} \leq \sum_{t \in \mathcal{T}^{\mathrm{brE}}(r)} t \cdot b_{r,s,t}^{\mathrm{end}} +$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} b_{r+1,s,t}^{\mathrm{start}} \cdot S_{s,d}^{\mathrm{end}} \quad \forall s \in \mathcal{S}^*(d), B^{\min} \leq r < B^{\max} \tag{76}$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brE}}} (t-1) \cdot b_{s,t}^{\mathrm{last}} \leq \sum_{t \in \mathcal{T}^{\mathrm{brE}}(B^{\max})} t \cdot b_{B^{\max},s,t}^{\mathrm{end}} +$$

$$\left(1 - \sum_{t \in \mathcal{T}^{\mathrm{brS}}(B^{\max})} b_{B^{\max},s,t}^{\mathrm{start}} \right) \cdot S_{s,d}^{\mathrm{end}} \quad \forall s \in \mathcal{S}^*(d) \tag{77}$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brE}}} b_{s,t}^{\mathrm{last}} = 1 \quad \forall s \in \mathcal{S}^*(d) \tag{78}$$

$$\sum_{r=1}^{B^{\max}} \sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} b_{r,s,t}^{\mathrm{start}} = \sum_{r=1}^{B^{\max}} \sum_{t \in \mathcal{T}^{\mathrm{brE}}(r)} b_{r,s,t}^{\mathrm{end}} \quad \forall s \in \mathcal{S}^*(d) \tag{79}$$

$$\sum_{r=1}^{B^{\max}} \left(\sum_{l \in \mathcal{T}^{\mathrm{brS}}(r):l \geq t} b_{r,s,l}^{\mathrm{start}} - \sum_{l \in \mathcal{T}^{\mathrm{brE}}(r):l \geq t} b_{r,s,l}^{\mathrm{end}} \right) = a_{s,t}$$

$$\forall s \in \mathcal{S}^*(d), S_s^{\mathrm{start}} \leq t \leq S_s^{\mathrm{end}} \tag{80}$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} b_{r,s,t}^{\mathrm{start}} \leq 1 \wedge \sum_{t \in \mathcal{T}^{\mathrm{brE}}(r)} b_{r,s,t}^{\mathrm{end}} \leq 1 \quad \forall 1 \leq r \leq B^{\max}, s \in \mathcal{S}^*(d)$$

$$\tag{81}$$

$$\sum_{t \in \mathcal{T}^{\mathrm{brS}}(r+1)} b_{r+1,s,t}^{\mathrm{start}} \leq \sum_{t \in \mathcal{T}^{\mathrm{brS}}(r)} b_{r,s,t}^{\mathrm{start}} \forall 1 \leq r < B^{\max}, s \in \mathcal{S}^*(d)$$

$$\tag{82}$$

$$b_{r,s,t}^{\mathrm{start}}, b_{r,s,t'}^{\mathrm{end}}, b_{s,t''}^{\mathrm{last}} \in \{0,1\}$$

$$\forall s \in \mathcal{S}^*(d), B^{\min} \leq r \leq B^{\max}, t \in \mathcal{T}^{\mathrm{brS}}(r), t' \in \mathcal{T}^{\mathrm{brE}}(r),$$
$$t'' \in \mathcal{T}^{\mathrm{brE}} \tag{83}$$

$$a_{s,t} \in \{0,1\} \quad \forall s \in \mathcal{S}^*(d), q \in \mathcal{Q}, S_s^{\mathrm{start}} \leq t \leq S_s^{\mathrm{end}} \tag{84}$$

$$y_{q,t}^+, y_{q,t}^- \geq 0 \quad \forall q \in \mathcal{Q}, S_s^{\mathrm{start}} \leq t \leq S_s^{\mathrm{end}} \tag{85}$$

Constraints (68)–(85) are equivalent to (42)–(54), but now workers' shifts $\mathcal{S}^*(d)$ for each day $d \in \mathcal{D}$ are provided by Algorithm 1.

Appendix E: Low demand curves

See Fig. 8.

Fig. 8 Variable and stable low demand curves

(a) Variable demand

(b) Stable demand
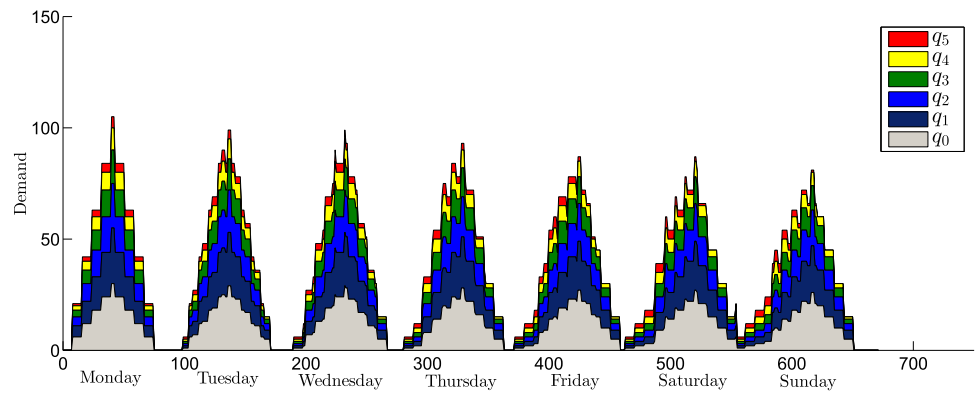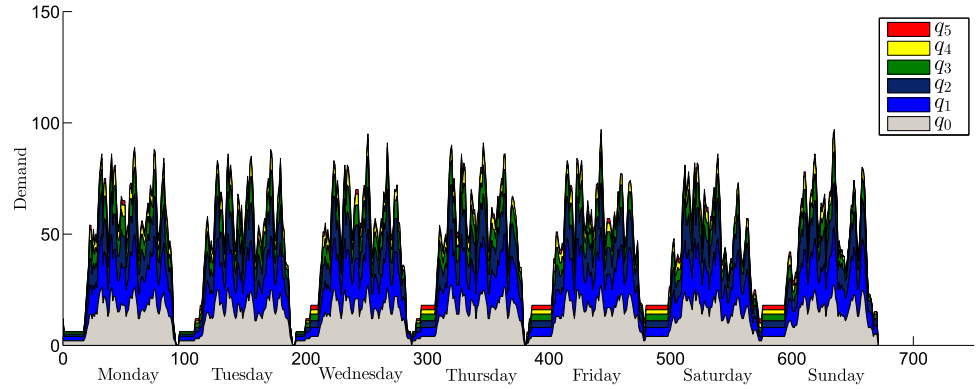
Appendix F: Results of computational study

See Tables 16, 17 and 18.

Table 16 Results for the CMIP and the MIP-heuristic for break regulation $\{S|V|T\}$

Instance	CMIP			MIP-heuristic			
	Obj	Undercov	Runtime (s)	Obj	Undercov	Runtime	GAP*
{10, low, SD, fix}	7, 570	6, 520	15.77	7, 570	6, 520	3.68	0.00
{20, low, SD, fix}	7, 680	5, 055	30.92	7, 680	5, 055	4.45	0.00
{30, low, SD, fix}	8, 035	4, 010	78.81	8, 035	4, 010	8	0.00
{40, low, SD, fix}	8, 695	3, 095	272	8, 750	3, 150	9.42	0.63
{50, low, SD, fix}	9, 805	2, 630	268.53	9, 975	2, 800	9.51	1.7
{65, low, SD, fix}	11, 150	2, 050	55.3	11, 450	2, 360	11.8	2.71
{10, low, SD, flex}	7, 350	6, 033	33.9	7, 350	5, 690	8.82	0.00
{20, low, SD, flex}	7, 350	5, 447	94.78	7, 350	4, 595	6.83	0.00
{30, low, SD, flex}	7, 350	4, 652	1, 147.93	7, 350	3, 620	8.53	0.00
{40, low, SD, flex}	7, 350	3, 857	2, 868.06	7, 350	3, 220	13.45	0.00
{50, low, SD, flex}	7, 350	3, 328	10, 074.84	7, 350	3, 010	12.72	0.00
{65, low, SD, flex}	7, 380	2, 077	14, 074	7, 420	2, 495	57.436	0.54
{10, low, VD, fix}	9, 086	8, 036	2.6	9, 086	8, 036	1.34	0.00
{20, low, VD, fix}	9, 121	6, 496	8.48	9, 141	6, 517	2.14	0.22
{30, low, VD, fix}	9, 283	5, 258	99.49	9, 307	5, 245	5.92	0.26
{40, low, VD, fix}	9, 663	3, 957	748.99	9, 851	5, 599	10.64	1.91
{50, low, VD, fix}	10, 188	3, 013	830.01	10, 486	3, 311	12.58	2.84
{65, low, VD, fix}	11, 234	2, 134	499, 07	11, 671	2, 572	12, 76	3.74
{10, low, VD, flex}	9, 084	7, 857	20.55	9, 089	7, 133	4.93	0.01
{20, low, VD, flex}	9, 084	7, 070	436.42	9, 105	5, 759	11.72	0.23
{30, low, VD, flex}	9, 084	6, 158	2, 243.9	9, 113	4, 740	21.34	0.32
{40, low, VD, flex}	9, 084	5, 028	5, 326.08	9, 147	3, 850	36.08	0.69
{50, low, VD, flex}	9, 084	4, 546	8, 343.92	9, 185	3, 503	87.52	1.10
{65, low, VD, flex}	9, 368	3, 727	18, 000	9, 518	3, 225	614.95	1.58

Table 17 Results for the CMIP and the MIP-heuristic for break regulation $\{F|V|W\}$

Instance	CMIP			MIP-heuristic				
	Obj	Undercov	Runtime (s)	Obj	Undercov	Runtime (s)	GAP*	GAP
{10, low, SD, fix}	7, 510	6, 460	32.32	7, 510	6, 460	3.56	0.00	
{20, low, SD, fix}	7, 590	4, 965	189.32	7, 590	4, 965	4.10	0.00	
{30, low, SD, fix}	7, 890	3, 865	1, 076.60	7, 900	3, 875	13.92	0.13	
{40, low, SD, fix}	8, 412	2, 813	2, 454.28	8, 550	2, 740	19.55	1.61	
{50, low, SD, fix}	9, 487	2, 312	2, 744.08	9, 640	2, 465	22.21	1.59	
{65, low, SD, fix}				11, 164	2, 064	29.47		7.52
{10, low, SD, flex}	7, 350	5, 690	203.92	7, 350	5, 690	9.05	0.00	
{20, low, SD, flex}				7, 350	4, 595	11.16		0.32
{30, low, SD, flex}				7, 350	3, 620	13.62		0.78
{40, low, SD, flex}				7, 350	3, 220	15.20		0.96
{50, low, SD, flex}				7, 350	3, 010	17.71		1.12
{65, low, SD, flex}				7, 420	2, 495	59.18		0.94
{10, low, VD, fix}	9, 084	8, 034	31.97	9, 086	8, 036	3.54	0.02	
{20, low, VD, fix}	9, 108	6, 483	164.98	9, 133	6, 500	4	0.27	
{30, low, VD, fix}	9, 247	5, 222	4, 345.21	9, 247	5, 222	13.32	0.43	
{40, low, VD, fix}	9, 545	3, 839	7, 938, 34	9, 545	3, 862	14.29	1.37	
{50, low, VD, fix}	9, 784	2, 564	10, 800.3	10, 243	3, 068	18.21	4.48	
{65, low, VD, fix}				11, 443	2, 343	42.01		9.33
{10, low, VD, flex}	9, 084	7, 580	843.80	9, 084	7, 128	8.72	0.00	
{20, low, VD, flex}				9, 087	5, 741	12.20		0.12
{30, low, VD, flex}				9, 093	4, 720	28.30		0.42
{40, low, VD, flex}				9, 128	3, 831	43.16		1.00
{50, low, VD, flex}				9, 160	3, 478	98.51		1.35
{65, low, VD, flex}				10, 367	4, 286	628.68		1.64

Table 18 Comparison of implicit BAP formulations for break regulation $\{S|V|T\}$

Instance	IBAP I				IBAP II				IBAP IIi		
	Comp (s)	Pre (s)	#Var	#Constr	Comp (s)	Pre (s)	#Var	#Constr	Time (s)	#Var	#Constr
{10, low, SD, fix}	1.35	0.02	2,803.71	2,727.43	1.08	0.03	2,730.86	2,721.43	2.89	790.29	2,823.43
{50, low, SD, fix}	1.74	0.07	3,411.86	3,204.86	1.63	0.4	3,237.43	3,195	5.26	12,573	2,040.29
{100, low, SD, fix}	1.97	0.09	3,433.29	3,211.71	1.26	0.9	3,252.86	3,201	25.26	23,994.86	3,973.14
{10, low, SD, flex}	0.67	0.05	3,485.71	3,185.43	0.57	0.04	3,200.86	3,178.86	0.58	2,699.14	898.86
{50, low, SD, flex}	1.76	0.18	3,641	3,214	1.46	0.03	3,283	3,203	1.56	5,982	1,511
{100, low, SD, flex}	13.34	0.58	3,555.43	3,234.29	10.73	1	3,345.43	3,222	3.26	10,418.57	2,314
{10, low, VD, fix}	1.47	0.02	2,821.86	2,735.43	1.4	0.4	2,748.86	2,728.43	0.14	2,013.43	756
{50, low, VD, fix}	2.18	0.17	3,462.14	3,271.14	1.86	0.4	3,386.57	3253	2,07	7,653.43	1,806
{100, low, VD, fix}	2.31	0.18	3,460.43	3,264.29	2.9	0.32	3,371.14	3,247	13.26	14,154.86	3,036
{10, low, VD, flex}	0.98	0.14	3,544.43	3,126.29	1.73	0.02	3,276.43	3,201	0.51	2,826	941.14
{50, low, VD, flex}	1.83	0.63	3,779.29	3,385.43	0.34	0.04	3,332.57	3,357	2.51	6,879.86	1,725.43
{100, low, VD, flex}	1.77	2.19	3,658.14	3,267.14	1.07	0.23	3,440.71	3,251	4.71	10,366.29	2,243.43
{200, high, SD, fix}	1.72	0.23	3,427	3,216.29	2.75	0.05	3,263.14	3,205	18.97	27,157.71	5,496
{250, high, SD, fix}	3.15	0.29	3,450.14	3,243.71	2.73	0.06	3,324.86	3229	19.5	33,659.14	6,726
{300, high, SD, fix}	2.55	0.2	3459.14	3,247.14	2	0.04	3,332.57	3232	20.96	40,160.57	7,956
{200, high, SD, flex}	2.02	0.44	3,516.29	3,318.57	1.8	0.08	3,605	3,301	10.16	25,488	5,571.43
{250, high, SD, flex}	1.55	0.59	3,569	3,329.71	0.23	0.27	3,595.14	3,298	11.92	26,324.14	5591.71
{300, high, SD, flex}	1.2	0.16	3,561,86	3,238.57	1.57	1.19	3,624.71	3,307	16.96	28,550.86	5,658.86
{200, high, VD, fix}	2.96	0.32	3,461.57	3,266.57	3.22	0.06	3,376.29	3,249	17.34	27,157.71	5,496
{250, high, VD, fix}	2.88	0.35	3,457	3,264.29	1.38	0.04	3,371.14	3,247	22.17	33,659.14	6,726
{300, high, VD, fix}	2.2	0.26	3,458.71	3,264.29	2.76	0.04	3,371.14	3,247	27.76	40,160.57	7,956
{200, high, VD, flex}	2.65	0.94	3,613.86	3,281.21	1.73	0.17	3,486.71	3,265	13.3	25,488	5,571.43
{250, high, VD, flex}	2.06	0.24	3,573.43	3,234	4.15	1.41	3,345.43	3,222	20.03	27,343.34	6,432.32
{300, high, VD, flex}	5.02	0.36	3,596.71	3,234.86	2.43	0.43	3,348.71	3,223	29.03	29.038.32	8,231.21

References

Addou, I., & Soumis, F. (2007). Bechtold-Jacobs generalized model for shift scheduling with extraordinary overlap. *Annals of Operations Research*, *155*(1), 177–205.

Alfares, H. K. (2007). Operator staffing and scheduling for an IT-help call centre. *European Journal of Industrial Engineering*, *1*(4), 414–430.

Ásgeirsson, E. I. (2012). Bridging the gap between self schedules and feasible schedules in staff scheduling. *Annals of Operations Research*, *218*, 1–19.

Ashford, N., Stanton, H., & Moore, C. (1997). *Airport operations*. New York: McGraw Hill.

Avramidis, A. N., Chan, W., Gendreau, M., ECuyer, P., Pisacane, O., et al. (2010). Optimizing daily agent scheduling in a multiskill call center. *European Journal of Operational Research*, *200*(3), 822–832.

Aykin, T. (1996). Optimal shift scheduling with multiple break windows. *Management Science*, *42*(4), 591–602.

Bard, J., & Wan, L. (2006). The task assignment problem for unrestricted movement between workstation groups. *Journal of Scheduling*, *9*(4), 315–341.

Bard, J. F. (2004). Selecting the appropriate input data set when configuring a permanent workforce. *Computers and Industrial Engineering*, *47*(4), 371–389.

Bard, J. F., & Wan, L. (2008). Workforce design with movement restrictions between workstation groups. *Manufacturing and Service Operations Management*, *10*(1), 24–42.

Bard, J. F., Morton, D. P., & Wang, Y. M. (2007). Workforce planning at usps mail processing and distribution centers using stochastic optimization. *Annals of Operations Research*, *155*(1), 51–78.

Bechtold, S., & Jacobs, L. (1990). Implicit modeling of flexible break assignments in optimal shift scheduling. *Management Science*, *36*, 1339–1351.

Beer, A., Musliu, N., Schafhauser, W., & Slany, W. (2010). An ai-based break-scheduling system for supervisory personnel. *Intelligent Systems, IEEE*, *25*(2), 60–73.

Brunner, J. O., & Stolletz, R. (2014). Stabilized branch and price with dynamic parameter updating for discontinuous tour scheduling. *Computers and Operations Research*, *44*, 137–145.

Brunner, J. O., Bard, J. F., & Kolisch, R. (2009). Flexible shift scheduling of physicians. *Health Care Management Science*, *12*(3), 285–305.

Brunner, J. O., Bard, J. F., & Kolisch, R. (2010). Midterm scheduling of physicians with flexible shifts using branch and price. *IIE Transaction*, *43*(2), 84–109.

Brusco, M. J., & Jacobs, L. W. (1998). Personnel tour scheduling when starting-time restrictions are present. *Management Science*, *44*(4), 534–547.

Brusco, M. J., & Jacobs, L. W. (2000). Optimal models for meal-break and start-time flexibility in continuous tour scheduling. *Management Science*, *46*(12), 1630–1641.

Çecik, T., & Günlük, O. (2004). Reformulating linear programs with transportation constraints—with applications to workforce scheduling. *Naval Research Logistics*, *51*(2), 275–296.

Çecik, T., Günlük, O., & Luss, H. (2001). An integer programming model for the weekly tour scheduling problem. *Naval Research Logistics*, *48*(7), 607–624.

Chu, S. (2007). Generating, scheduling and rostering of shift crew-duties: Applications at the Hong Kong International Airport. *European Journal of Operational Research*, *177*, 1764–1778.

Côté, M.-C., Gendron, B., & Rousseau, L.-M. (2007). *Modeling the regular constraint with integer programming. Integration of AI and OR techniques in constraint programming for combinatorial optimization problems* (pp. 29–43). Berlin: Springer.

Dantzig, G. B. (1954). Letter to the editor-a comment on Edie's "traffic delays at toll booths". *Journal of the Operations Research Society of America*, *2*(3), 339–341.

Dowling, D., Krishnamoorthy, M., Mackenzie, H., & Sier, D. (1997). Staff rostering at a large international airport. *Annals of Operations Research*, *72*, 125–147.

Ernst, A. T., Jiang, H., Krishnamoorthy, M., & Sier, D. (2004). Staff scheduling and rostering: A review of applications, methods and models. *European Journal of Operational Research*, *153*, 3–27.

Garey, M., & Johnson, D. (1979). *Computers and intractability: A guide to the theory of NP-completeness*. New York: Freeman.

Kiermaier, F., Frey, M., & Bard, J. (2016). Flexible cyclic rostering in the service industry. *IIE Transactions*, *42*(12), 1139–1155.

Kuo, Y.-H., Leung, J., & Yano, C. (2014). Scheduling of multi-skilled staff across multiple locations. *Production and Operations Management*, *23*(4), 626–644.

Lau, H. C. (1996). On the complexity of manpower shift scheduling. *Computers and Operations Research*, *23*(1), 93–102.

Lim, G. J., Mobasher, A., Bard, J. F., & Najjarbashi, A. (2016). Nurse scheduling with lunch break assignments in operating suites. *Operations Research for Health Care*, *10*, 35–48.

Love, R. R., & Hoey, J. M. (1990). Management science improves fast-food operations. *Interfaces*, *20*(2), 21–29.

Lusby, R., Dohn, A., Range, T. M., & Larsen, J. (2011). A column generation-based heuristic for rostering with work patterns. *Journal of the Operational Research Society*, *63*(2), 261–277.

Mac-Vicar, M., Ferrer, J. C., Muñoz, J. C., & Henao, C. A. (2017). Real-time recovering strategies on personnel scheduling in the retail industry. *Computers and Industrial Engineering*, *113*(2), 589–601.

Mirrazavi, S., & Beringer, H. (2007). A web-based workforce management system for Sainsburys Supermarkets Ltd. *Annals of Operations Research*, *155*(1), 437–457.

Nemhauser, G., & Wolsey, A. (1988). *Integer and combinatorial optimization*. New York: Wiley.

Ni, H., & Abeledo, H. (2007). A branch-and-price approach for large-scale employee tour scheduling problems. *Annals of Operations Research*, *155*(1), 167–176.

Rekik, M., Cordeau, J.-F., & Soumis, F. (2004). Using Benders' decomposition to implicitly model tour scheduling. *Annals of Operations Research*, *128*, 111–133.

Rekik, M., Cordeau, J.-F., & Soumis, F. (2010). Implicit shift scheduling with multiple breaks and work stretch duration restrictions. *Journal of Scheduling*, *13*, 49–75.

Restrepo, M. I., Gendron, B., & Rousseau, L.-M. (2016). Branch-and-price for personalized multiactivity tour scheduling. *INFORMS Journal on Computing*, *28*(2), 334–350.

Rong, A. (2010). Monthly tour scheduling models with mixed skills considering weekend off requirements. *Computers & Industrial Engineering*, *59*(2), 334–343.

Stolletz, R. (2010). Operational workforce planning for check-in counters at airports. *Transportation Research Part E: Logistics and Transportation Review*, *46*(3), 414–425.

Sungur, B., Özgüven, C., & Kariper, Y. (2017). Shift scheduling with break windows, ideal break periods, and ideal waiting times. *Flexible Services and Manufacturing Journal*, *29*(2), 203–222.

Thompson, G. M., & Pullman, M. E. (2007). Scheduling workforce relief breaks in advance versus in real-time. *European Journal of Operational Research*, *181*(1), 139–155.

Tien, J. M., & Kamiyama, A. (1982). On manpower scheduling algorithms. *SIAM Review*, *24*, 275–287.

Topaloglu, S., & Ozkarahan, I. (2004). An implicit goal programming model for the tour scheduling problem considering the employee work preferences. *Annals of Operations Research*, *128*(1–4), 135–158.

Van den Bergh, J., Belién, J., De Bruecker, P., Demeulemeester, E., & De Boeck, L. (2013). Personnel scheduling: A literature review. *European Journal of Operational Research*, *226*(3), 367–385.

Widl, N., & Musliu, M. (2014). The break scheduling problem: Complexity results and practical algorithms. *Memetic Computing*, *6*(2), 97–112.