Noname manuscript No.
(will be inserted by the editor)

Demand smoothing in shift design

Pieter Smet · Annelies Lejon · Greet Vanden Berghe

Received: date / Accepted: date

Abstract Shift design is an essential step in workforce planning in which staffing requirements must be obtained for a set of shifts which best cover forecasted demand given as a demand pattern. Existing models for this challenging optimization problem perform well when these demand patterns fluctuate around an average without any strong variability in demand. However, when demand is irregular, these models inevitably generate solutions with a significant amount of over- or understaffing or an excessive use of short shifts. The present paper explores a strategy which involves modifying the demand patterns such that the variable workload may be better matched using an acceptable number of shifts. Integer programming is employed to solve the resulting optimization problem. A computational study of the proposed model reveals interactions between different problem parameters which control the scope of demand modification and the type of the selected shifts. Moreover, the potential impact from an economic point-of-view is discussed and the time before profitability of the approach is evaluated. These insights enable operations management to better understand the trade-off between solution quality and different types of flexibility which may be realized in an organization.

Keywords Shift design · Irregular demand · Demand smoothing · Workforce planning · Integer programming

P. Smet
KU Leuven, Department of Computer Science, CODeS, Gebroeders De Smetstraat 1, 9000 Gent, Belgium
Tel.: +3292658704
E-mail: pieter.smet@kuleuven.be

A. Lejon
KU Leuven, Department of Computer Science, CODeS, Gebroeders De Smetstraat 1, 9000 Gent, Belgium

G. Vanden Berghe
KU Leuven, Department of Computer Science, CODeS, Gebroeders De Smetstraat 1, 9000 Gent, Belgium

1 Introduction

Workforce planning is a challenging task faced by many organizations. As personnel costs typically account for a significant proportion of an organization's operational expenses, it is vital to conduct this task as effectively as possible. The sequence of processes to arrive at an efficient workforce plan may vary depending on the area of application. For example, in sectors where employees are a scarce resource, it is common for them to participate in the construction of their own schedule. Practices such as self-rostering and preference-rostering try to find a balance between overall cost-efficiency and employee satisfaction (Asgeirsson, 2014; van der Veen et al., 2016). By contrast, this paper considers workforce planning as a centralized decision-making process consisting of sequential steps which may be grouped into three main processes: demand modeling, shift rostering and disruption handling (Defraeye and Van Nieuwenhuyse, 2016) Figure 1 provides an overview of these high-level processes with their detailed steps. Note that the strategic level of decision-making, primarily concerned with manpower planning, is also relevant in workforce planning (Llort et al., 2019). As discussed by Komarudin et al. (2013), decisions at this level have a significant influence upon the others, and vice versa.

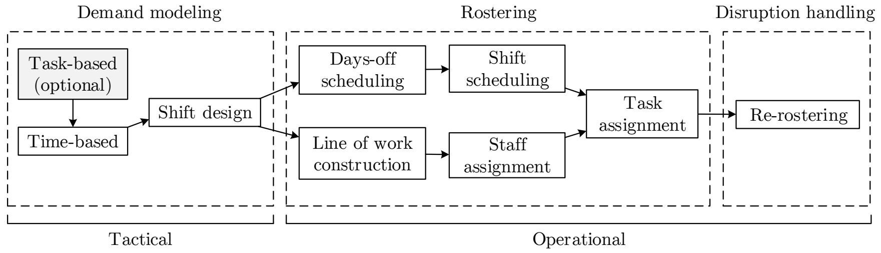


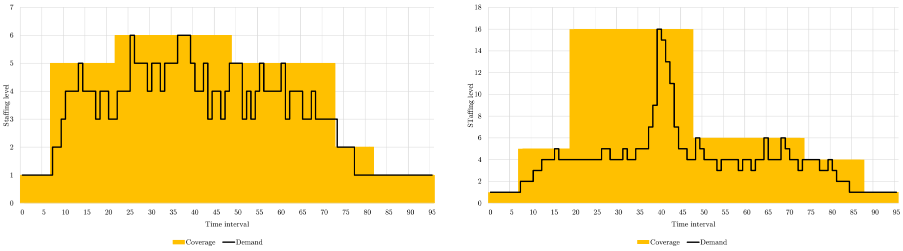
Fig. 1: Overview of centralized decision-making processes in workforce planning

At the operational level of decision-making, the rostering process assigns shifts and tasks to employees resulting in a schedule (also referred to as a roster). Several methodologies for constructing rosters have been established over the course of many years of research (Ernst et al., 2004). The two most common approaches are to either (i) construct complete individual work schedules which are then assigned to available employees or (ii) determine on which days employees are working and only then assign shifts. Optimization problems encountered during this process are typically subject to a variety of personal and organizational constraints such as, for example, skill requirements, contracts and individual preferences (Van den Bergh et al., 2013). The last step in the rostering process is to assign detailed tasks to employees, although this decision is often integrated into the construction of individual work schedules (Smet et al., 2016). The dynamic nature of workforce planning is subsequently accommodated via disruption handling. Disruptions caused by unforeseen events such as, for example, employee illness, are addressed in the re-rostering step (Gross et al., 2018; Maenhout and Vanhoucke, 2013).

At the tactical level, demand modeling determines the number of employees required on different dates and at different times based on forecasts of the expected workload or the actual tasks to be performed (Ernst et al., 2004). Time-based demand modeling generalizes task-based modeling and specifies *demand patterns*, which can be considered functions that map required staffing levels to time intervals (Meisen et al., 2016). These patterns may be derived from a task-based model of the demand which specifies all the actions which must be performed along with relevant timing information and additional task-specific metadata (Herbers, 2006). Shift design transforms the forecasted demand into staffing levels for shifts which may then be employed during the operational processes (Musliu et al., 2004).

Shift design is an NP-hard combinatorial optimization problem which constitutes an essential step in workforce planning (Di Gaspero et al., 2007). Depending on the specific application, objectives and constraints may vary, however, the core decision problem remains the same: given a large set of possible shifts, select a subset of shifts and define staffing requirements for these shifts such that deviation from demand is minimized. While this step is required in centralized workforce planning, in decentralized approaches such as self- or preference rostering, employees themselves may become responsible for choosing their own working hours.

Existing models for shift design consider demand fixed and have no means of accommodating variability in demand patterns which are common in environments characterized by highly variable workload such as, for example, warehouses (Boonstra-Hörwein et al., 2011), restaurants (Love and Hoey, 1990), hospitals (Litvak et al., 2005) or airports (Moore et al., 1996; Clausen, 2010). As a result, existing models produce solutions with high levels of under- or overstaffing which, in practice, translates into reduced quality of service or idle employees, both of which are undesirable for any organization (Gärtner and Kundi, 2005). Consider the examples shown in Figure 2 which visualize two different shift design scenarios. In Figure 2a, the demand fluctuations are relatively small. The amount of overstaffing in the solution is limited to 94 units, which, if one time interval lasts 15 minutes, corresponds to 23.5 hours of idle time distributed across the workforce. Using the same model, it is impossible to cover the irregular demand pattern shown in Figure 2b without incurring a significantly larger amount of overstaffing, in this example 363 units or 90.75 hours. The resulting coverage corresponds to a so-called *simple peak hour approximation* (Green and Kolesar, 1995). If organizations find this level of overstaffing to be excessive they may opt to focus on the average demand rather than on peak demand, however this will lead to understaffing during the peaks (Green et al., 2007). Including short two- or three-hour shifts would result in a better match of the irregular demand pattern, however, this is typically considered undesirable for employees (Burgess, 2007). Longer shifts allow for a more compressed working week, which has been shown to positively affect job satisfaction (Baltes et al., 1999). Moreover, hiring employees willing to work short shifts it not always possible while employing temporary workers brings it own set of challenges (Buzacott and Mandelbaum, 2008).



(a) Regular demand pattern

(b) Irregular demand pattern

Fig. 2: Examples of demand patterns and the resulting optimal shift coverage

The present paper investigates an alternative strategy in which demand patterns are slightly adjusted or *smoothed*, such that variability in demand may be better accommodated without using an excessive number of short shifts. The proposed approach enables quantifying the trade-off between cost reduction for the company (by reducing under- and overstaffing) and the cost required to modify the demand pattern. Despite the fact that demand requirements are often exogenous to the shift design phase and imposed by a third party or stemming from an internal process, it is often possible to adjust these patterns. For example, by arranging alternative delivery times of suppliers, hiring additional workers throughout the day for internal logistic processes or negotiating with airlines to adjust flight departure and arrival times. Operational restrictions concerning demand patterns have to be accounted for by way of constraints which limit the scope of modifications. The proposed decision support model is formulated as an integer programming problem and is analyzed in a computational study to reveal interactions between its most prominent parameters controlling modification flexibility and feasible shift selection.

The remainder of this paper is organized as follows. Section 2 surveys previous research on related shift design problems. The proposed model for addressing shift design with demand smoothing is presented in Section 3. Section 4 introduces an integer programming formulation for the defined optimization problem. Section 5 analyzes a series of computational experiments to generalize the interaction between different model parameters. The impact of demand smoothing in practice is discussed in an economic context in Section 6. Finally, conclusions and directions for future research are outlined in Section 7.

2 Related work

Workforce planning, and personnel rostering in particular, has been the subject of many studies proposing different models and algorithms for a variety of applications (Van den Bergh et al., 2013; De Bruecker et al., 2015). Shift design, on the other hand, has received far less attention, even though it constitutes an essential step in workforce planning (Ernst et al., 2004). Studies on this problem typically consider three decisions: (i) shift selection (which shifts to use), (ii) staffing (how many employees to assign to each shift to meet the demand) and (iii) break scheduling (where to place the breaks in the selected shifts).

Early models for shift design addressed the first two decisions (shift selection and staffing) as a single decision process while considering break scheduling in a separate step to be solved afterward. This methodology makes the optimization problems easier to solve although there is no guarantee that an optimal solution with respect to shift selection and staffing will lead to a good solution when breaks have to be scheduled. These models typically have multiple objectives such as, for example, minimizing the deviation from the demand, minimizing the number of shifts used or balancing workload (Musliu et al., 2004; Di Gaspero et al., 2007). Lusby et al. (2016) include additional hard constraints on the maximum number of shifts and the number of available employees on each day. Even without considering breaks or skills, these optimization problems are already very challenging from a computational point of view and have lead to the development of a wide range of heuristics.

The problem of break scheduling when shift selection and staffing requirements are given has been studied by Widl and Musliu (2010). The objective function in their model minimizes deviation from the demand. A number of temporal constraints concerning break placement and length must be respected. For example, a break cannot be scheduled too close to the beginning or end of the shift and the time between two breaks must be sufficiently long.

Similar to the model proposed by Bonutti et al. (2017), the present paper addresses the three aforementioned decisions simultaneously considering simplified break scheduling constraints. Di Gaspero et al. (2010), meanwhile, present an integrated model for the three decisions including complex temporal break scheduling constraints, requiring a dedicated constraint programming model integrated in a heuristic search algorithm to solve the resulting problem. The same problem is considered by Akkermans et al. (2019) who use a two-phase integer programming approach in which shifts are initially chosen while approximating the impact of breaks. The second phase then iteratively schedules breaks for each shift. Hur et al. (2019) propose five different integer programming models for shift selection and staffing in which breaks may be scheduled in real-time while minimizing understaffing or maximizing the number of breaks which may be assigned. Additionally, a rolling horizon approach is proposed designed to accommodate short-term demand forecasts in order to efficiently manage fluctuations in demand.

As shown in Figure 1, shift design is typically considered a tactical long-term decision made based on workload forecasts rather than the actual workload at the time rosters are executed. Most studies focus on the deterministic variant of shift design which does not take into account any uncertainty. For applications in which workload forecasts are unreliable, these deterministic models will often result in sub-optimal solutions in practice when the realized workload differs significantly from the estimated workload. Uncertainty in shift design is discussed by van Hulst et al. (2017) who apply robust optimization to obtain a selection of shifts and staffing requirements for a practical case involving air navigation services. Their solutions are considered robust given that levels of under- and overstaffing will not change significantly when the actual workload differs from the forecasted workload. In a computational study the trade-off is shown which exists between the robustness and solution quality in terms of number of used shifts.

Table 1 compares the models and solution approaches described in the literature to those used in this paper. The proposed model considers the three main decisions in shift design while including demand smoothing and most of the common problem characteristics from the literature such as night shifts, skills and minimizing under- and overstaffing. The model introduced by Bonutti et al. (2017) lies closest to the one considered in the present paper with three notable differences. First, the present paper is the first to model demand smoothing in shift design. Second, only a single day is considered due to the increased computational complexity which arises from including demand smoothing. Third, the number of shifts is not minimized. Instead, constraints are included which limit the number of shifts, as proposed by Lusby et al. (2016).

Table 1: Overview of related work (WS: weighted sum objective function, LEX: lexicographic objective function)

	Decisions	Objectives		Multiple days	Night shifts	Breaks	Skills	Additional constraints	Solution approach
Musliu et al. (2004)	Shift selection + requirement definition	Under/overstaffing + nr of shifts + workload balancing	(WS)	Yes	Yes	No	No	-	Tabu search
Di Gaspero et al. (2007)	Shift selection + requirement definition	Under/overstaffing + nr of shifts	(WS)	Yes	Yes	No	No	-	Tabu search
Di Gaspero et al. (2007)[1]	Shift selection + requirement definition	Under/overstaffing + nr of shifts	(LEX)	No	Yes	No	No	-	Network flow
Lusby et al. (2016)	Shift selection + requirement definition	Under/overstaffing	(WS)	Yes	No	No	No	Max nr of shifts + max nr of employees	Benders decomposition heuristic
van Hulst et al. (2017)	Shift selection + requirement definition	Under/overstaffing + nr of shifts	(WS)	Yes	No	No	No	-	Robust optimization
Widl and Musliu (2010)	Break scheduling	Under/overstaffing	(WS)	No	Yes	Yes	No	Break constraints	Memetic algorithm
Di Gaspero et al. (2010)	Shift selection + requirement definition + break scheduling	Under/overstaffing + nr of shifts	(WS)	Yes	Yes	Yes	No	Break constraints	Local search with constraint programming
Akkermans et al. (2019)	Shift selection + requirement definition + break scheduling	Under/overstaffing + nr of shifts	(WS)	Yes	Yes	Yes	No	Break constraints	Integer programming
Hur et al. (2019)	Shift selection + requirement definition + break scheduling	Working hours or no. of breaks or understaffing		No	No	Yes	No	Break constraints + real-time break scheduling	Integer programming
Bonutti et al. (2017)	Shift selection + requirement definition + break scheduling	Under/overstaffing + nr of shifts + avg shift length	(WS)	Yes	Yes	Yes	Yes	-	Simulated annealing
This paper	Shift selection + requirement definition + break scheduling	Under/overstaffing	(WS)	No	Yes	Yes	Yes	Max nr of shifts + max nr of short shifts + demand smoothing	Integer programming

* Special case of the *min*-shift design problem

3 Problem definition

As in the canonical shift design problem, the aim in the present paper is to determine the set of shifts that best matches the demand requirements for a single day. The one-day planning horizon is discretized into a set of time intervals $T = \{1, \ldots, |T|\}$ based on a given time granularity Γ. Given the increasingly high- and multi-skilled environments encountered in practice, the model also includes a set of skills denoted by $K = \{1, ..., |K|\}$.

3.1 Shift characteristics

The set of all possible shifts is generated based on a set of shift templates. A template is defined by an earliest start time, latest start time, latest end time and minimum and maximum shift durations. A template may require one break of a specific length to be scheduled which cannot be placed too close to the beginning or end of the shift. Table 2 provides four examples of shift templates. The Day template does not require a break to be scheduled. Note that the combination of latest start time and maximum shift duration implies the latest end time, meaning that this parameter is not explicitly required.

Table 2: Examples of shift templates

Type	Earliest start	Latest start	Latest end	Minimum duration	Maximum duration	Break duration	Start offset	End offset
Early	6:00	8:00	-	7h	8h	1h	2h	2h
Day	9:00	11:30	-	6h	7h 30m	-	-	-
Day-short	9:00	11:30	-	3h 30m	4h	45m	1h	1h
Late	13:00	15:00	22:30	7h	8h	1h	1h	2h

Based on the set of shift templates, the set of all possible shifts $S = \{1, ..., |S|\}$ is generated. Here, a shift may be considered a realization of a shift template which respects all of the parameter ranges specified in the template. A shift is characterized by specific start and end times which lay within the bounds defined by the template. If a break is required, it should be scheduled appropriately, in accordance with the constraints imposed by the template. The subset of shifts which cover interval $t \in T$ is denoted by $S_t \subseteq S$. Table 3 shows a subset of the shifts derived from the Early and Day-short templates using a time granularity $\Gamma = 30$ minutes.

The proposed model distinguishes between short (shifts with a duration of four hours or less) and long shifts, thereby enabling appropriate shift configurations for companies whose workforce contains a mixture of part-time and full-time employees. The subset of short shifts is denoted by $S^{short} \subseteq S$ and long shifts by $S^{long} \subseteq S$ such that $S = S^{short} \cup S^{long}$. A parameter ϕ restricts the maximum number of short shifts in relation to the total number of selected shifts. For example, if a solution uses twelve shifts in total and $\phi = 0.25$, than

Table 3: Shifts derived from templates in Table 2 with $\Gamma = 30$ minutes

(a) Early template

Start	End	Break start	Break end
6:00	13:00	8:00	9:00
6:00	13:30	8:00	9:00
6:00	14:00	8:00	9:00
6:30	13:30	8:30	9:30
6:30	14:00	8:30	9:30
...	...	...	...
7:00	14:00	9:00	10:00
7:00	14:30	9:00	10:00
7:00	15:00	9:00	10:00

(b) Day-short template

Start	End	Break start	Break end
9:00	12:30	10:00	10:45
9:00	12:30	10:30	11:15
9:00	13:00	10:00	10:45
9:00	13:00	10:30	10:45
9:00	13:00	11:00	11:45
...	...	...	...
11:30	15:30	12:45	13:30
11:30	15:30	13:15	14:00
11:30	15:30	13:45	14:30

at most three out of the twelve shifts may have a duration of less than four hours. The total number of shifts selected from S is limited to at most η.

This model permits the inclusion of shifts which span two days such as, for example, night shifts, by modeling them as cyclic shifts. In this case, the final interval on a given day is assumed to be consecutive with respect to the day's first interval. Figure 3 illustrates an example of the intervals covered by different types of shifts, including a cyclic night shift.

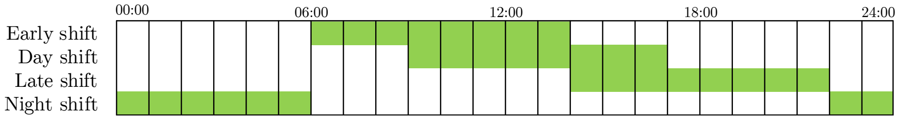


Fig. 3: Intervals covered by different shifts

3.2 Demand characteristics

A demand pattern is defined by the required number of employees r_{tk} in each interval $t \in T$ and for each skill $k \in K$. The problem environments of particular interest to this paper are characterized by highly irregular demand patterns. Specifically, the demand patterns contain intervals with peak demand, while the demand between such peaks fluctuates around some base level. Let P be the set of demand peaks. A peak $p \in P$ is defined by a set of intervals $\{t_p^{\text{start}}, ..., t_p^{\text{end}}\}$ in which the demand is considerably higher than the base level demand. A peak may consist of only a single interval such that $t_p^{\text{start}} = t_p^{\text{end}}$. Figure 4 provides an example of an irregular demand pattern and illustrates the various parameters associated with a peak. Note that the concept of a peak may be generalized to any set of intervals in which demand may be modified. Such a generalization reveals a larger field of application of the proposed model than demand patterns with peaks which are the main focus in this paper.

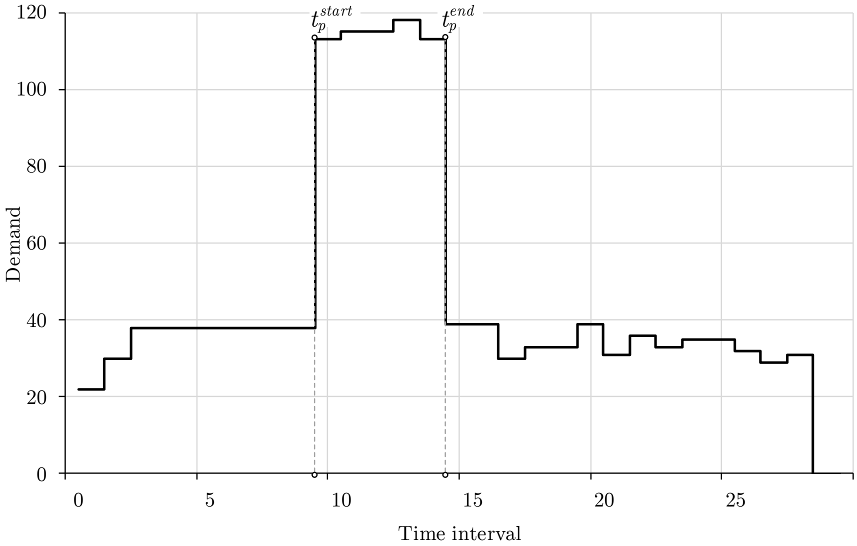


Fig. 4: Example of an irregular demand pattern with one demand peak

As mentioned before, existing models for shift design are unable to generate acceptable solutions when demand peaks such as those illustrated in Figure 4 occur. To address this issue, the proposed model permits slight adjustments to the demand patterns such that the demand may be better matched by the available shifts. Adjustments occur in the form of demand redistribution in which demand is decreased in some intervals and increased in others. This process is denoted as demand *smoothing* as it aims to reduce the variability in the demand patterns. Conservation of demand is essential in this process: demand cannot be lost, that is, the total amount of demand decrease should equal the total amount of demand increase. To account for operational limitations with regard to demand smoothing, two additional parameters are introduced to control the scope of the adjustments.

First, the parameter $\tau_p \in \mathbb{Z}_0^+$ is used to define in which intervals peak $p \in P$ may be modified. Practical restrictions may limit how far away from the peak demand can be adjusted. Therefore, the proposed model only allows modifications to peak p in the intervals $\{t_p^{\text{start}} - \tau_p, ..., t_p^{\text{end}} + \tau_p\}$. Second, the amount of redistributed demand is controlled by a parameter $\beta_p \in \{r \in \mathbb{R} \mid 0 \leq r \leq 1\}$, which is used to define the maximum quantity of demand of peak p which may be redistributed as $b_p = \beta_p \times \sum_{t=t_p^{\text{start}}}^{t_p^{\text{end}}} r_{tk_p}$, where k_p is the skill for which peak p exists.

Demand peaks are intuitively easier to match when the demand within the peak is reduced, while demand outside the peak is increased. This intuition is captured in the model by only allowing demand decreases for peak p in intervals $T_p^{\downarrow} = \{t_p^{\text{start}}, ..., t_p^{\text{end}}\}$ and only allowing demand increases in $T_p^{\uparrow} =$

$\{t_p^{\text{start}}-\tau_p, ..., t_p^{\text{start}}-1\} \cup \{t_p^{\text{end}}+1, ..., t_p^{\text{end}}+\tau_p\}$. Figure 5 highlights these subsets of intervals in an example with two demand peaks. Note that, depending on the specific application, the intervals which permit modifications may be defined based on other properties of the demand pattern, even some of which may be unrelated to demand peaks.

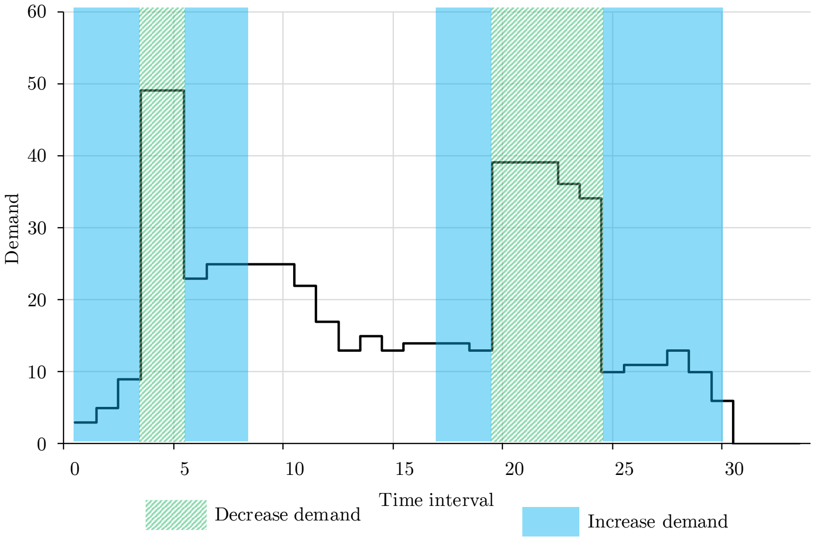


Fig. 5: Feasible ranges for demand modification of an irregular demand pattern containing two peaks

4 Integer programming formulation

An integer programming formulation is proposed to optimize shift selection, staffing, break scheduling and demand smoothing. For each $s \in S$, a binary variable y_s equals one if shift s is used in the solution and zero otherwise. For each $s \in S$ and $k \in K$, an integer variable x_{sk} equals the number of employees assigned to shift s to cover the demand of skill k. For each $t \in T$ and $k \in K$, two integer variables u_{tk} and o_{tk} are introduced which equal the amount of under- and overstaffing for skill k in interval t. Finally, for each $p \in P$ and each $t \in T_p^{\downarrow}$, the integer variable $m_{tp}^{\downarrow}$ equals the demand decrease of peak p in interval t. Similarly, for each $p \in P$ and $t \in T_p^{\uparrow}$, $m_{tp}^{\uparrow}$ equals the amount of demand increase of peak p in interval t. Let $\mathscr{C}^o$ and $\mathscr{C}^u$ be the weights associated with overstaffing and understaffing, respectively. For each skill $k \in K$, the set $P_k \subseteq P$ contains all peaks in the demand pattern for skill k. The shift design problem with demand smoothing may now be formulated as the following integer programming problem.

$$min \sum_{t \in T} \sum_{k \in K} (\mathscr{C}^o o_{tk} + \mathscr{C}^u u_{tk}) \tag{1}$$

$$s.t. \sum_{s \in S_t} x_{sk} + u_{tk} - o_{tk} = r_{tk} + \sum_{p \in P_k : t \in T_p^{\uparrow}} m_{tp}^{\uparrow} - \sum_{p \in P_k : t \in T_p^{\downarrow}} m_{tp}^{\downarrow} \quad \forall t \in T, k \in K \tag{2}$$

$$x_{sk} \leq M_{sk} y_s \quad \forall s \in S, k \in K \tag{3}$$

$$\sum_{s \in S} y_s \leq \eta \tag{4}$$

$$\sum_{s \in S^{\text{short}}} y_s \leq \phi \sum_{s \in S} y_s \tag{5}$$

$$\sum_{t \in T_p^{\downarrow}} m_{tp}^{\downarrow} + \sum_{t \in T_p^{\uparrow}} m_{tp}^{\uparrow} \leq 2 b_p \quad \forall p \in P \tag{6}$$

$$\sum_{t \in T_p^{\downarrow}} m_{tp}^{\downarrow} - \sum_{t \in T_p^{\uparrow}} m_{tp}^{\uparrow} = 0 \quad \forall p \in P \tag{7}$$

$$y_s \in \{0, 1\} \quad \forall s \in S \tag{8}$$

$$x_{sk} \in \mathbb{Z}_0^+ \quad \forall s \in S, k \in K \tag{9}$$

$$u_{tk}, o_{tk} \in \mathbb{Z}_0^+ \quad \forall t \in T, k \in K \tag{10}$$

$$m_{tp}^{\downarrow} \in \mathbb{Z}_0^+ \quad \forall p \in P, t \in T_p^{\downarrow} \tag{11}$$

$$m_{tp}^{\uparrow} \in \mathbb{Z}_0^+ \quad \forall p \in P, t \in T_p^{\uparrow} \tag{12}$$

Objective function (1) minimizes a weighted sum of over- and understaffing. Constraints (2) require the selected shifts to meet the modified demand exactly except for an amount of under- or overstaffing which is penalized in the objective function. Constraints (3) link the x_{sk} and y_s variables using a big M formulation. An appropriate value for M_{sk} for shift s and skill k may be calculated as $M_{sk} = \max_{t \in T}\{r_{tk} \mid s \in S_t\}$. Constraints (4) limit the total number of selected shifts. Constraints (5) make sure at most a fraction ϕ of all selected shifts are short shifts. Constraints (6) limit the amount of modification for each peak. Conservation in the sense that the total demand remains unchanged no matter which modifications are made, is ensured by Constraints (7). Constraints (8)-(10) define the domain of the y_s, x_{sk}, u_{tk} and o_{tk} decision variables. Constraints (11) and (12) bound variables $m_{tp}^{\downarrow}$ and $m_{tp}^{\uparrow}$, as illustrated in Figure 5.

5 Computational study

Two sets of parameters in the proposed model control distinct types of flexibility it offers. First, flexibility with respect to demand smoothing is determined by τ_p, which controls the range in which demand may be modified, and β_p, which limits the amount of demand to be redistributed. Second, flexibility with respect to shift selection is controlled by η, which limits the total number of shifts, and ϕ, which restricts the relative number of short shifts that can be selected. The computational study discussed in this section analyzes each set of parameters to understand their effect on the problem's solution.

5.1 Data and experimental setup

The instances used in the experiments are derived from the dataset published by Bonutti et al. (2017). The existing demand patterns were adapted to fit a time granularity of $\Gamma = 30$ minutes. Two types of irregular demand patterns were generated based on each of the original 28 instances: one with a single peak for each skill with $t_p^{\text{start}} \in [10{:}00, 12{:}00]$, and one with two peaks for each skill in which the first peak has $t_{p_1}^{\text{start}} \in [8{:}00, 9{:}00]$ and the second peak has $t_{p_2}^{\text{start}} \in [18{:}00, 19{:}00]$. Peak duration was chosen to be between 30 minutes and one hour. Peak demand was set by multiplying the original demand in the peak's first interval by a value randomly sampled from the interval $[2, 3]$. All peak parameters were sampled from uniform distributions. Following this methodology, 196 instances of each type (single- and double-peak) were generated. For the experiments in this study, 20 instances of each type were randomly selected[1]. The weights for under- and overstaffing in objective function (1) were both set to one. Table 4 provides an overview of the selected instances' most important characteristics. In addition to the total number of shifts, the number of night shifts and short shifts is also reported. For the single-peak instances, the average and maximum demand for each skill is also shown.

Table 4: Overview of instance characteristics

Instance	Number of templates	Number of shifts	Number of skills	Number of night shifts	Number of short shifts	Maximum demand[1]	Average demand[1]
MS01 - day3	2	2016	2	633	144	12/12	3.4/2.3
MS02 - day4	2	2016	2	633	144	8/8	3.0/2.4
MS03 - day4	2	2016	2	633	144	16/16	3.6/2.1
MS05 - day6	3	285	2	0	0	57/57	16.1/10.5
MS06 - day2	3	285	3	0	0	112/112/112	22.7/14.4/10.3
MS07 - day5	3	285	2	0	0	114/114	19.5/13.1
MS11 - day2	3	285	2	0	0	66/66	12.2/8.0
MS12 - day1	3	285	3	0	0	67/67/67	10.1/5.0/3.2
MS12 - day2	3	285	3	0	0	60/60/60	11.2/6.1/4.0
MS13 - day0	4	1548	2	19	69	101/101	24.5/15.1
MS15 - day4	4	1548	2	19	69	90/90	15.9/11.5
MS16 - day3	4	1548	3	19	69	60/60/60	11.8/6.9/4.2
MS18 - day2	4	1548	3	19	69	76/76/76	13.8/8.6/5.4
MS19 - day1	4	1548	2	19	69	83/83	11.1/8.8
MS19 - day3	4	1548	2	19	69	72/72	13.3/8.5
MS21 - day0	10	5494	2	909	78	88/88	23.7/17.2
MS22 - day6	10	5494	3	909	78	40/40/40	13.1/8.8/5.9
MS23 - day6	10	5494	2	909	78	41/41	9.7/6.2
MS25 - day5	10	5494	2	909	78	103/103	18.3/12.6
MS28 - day3	10	5494	3	909	78	47/47/47	9.9/6.9/3.3

[1] For the single-peak instance, including peak demand.

[1] All instances and solutions employed in the computational study are publicly available at `http://people.cs.kuleuven.be/~pieter.smet/shiftdesign`.

All experiments were conducted on a Dell Poweredge T620, 2x Intel Xeon E5-2670 with 128GB RAM. Gurobi 7.5.2 was used as the integer programming solver with its default settings and configured to use one thread. As a stopping criterion, a time limit of two hours was imposed for each experiment.

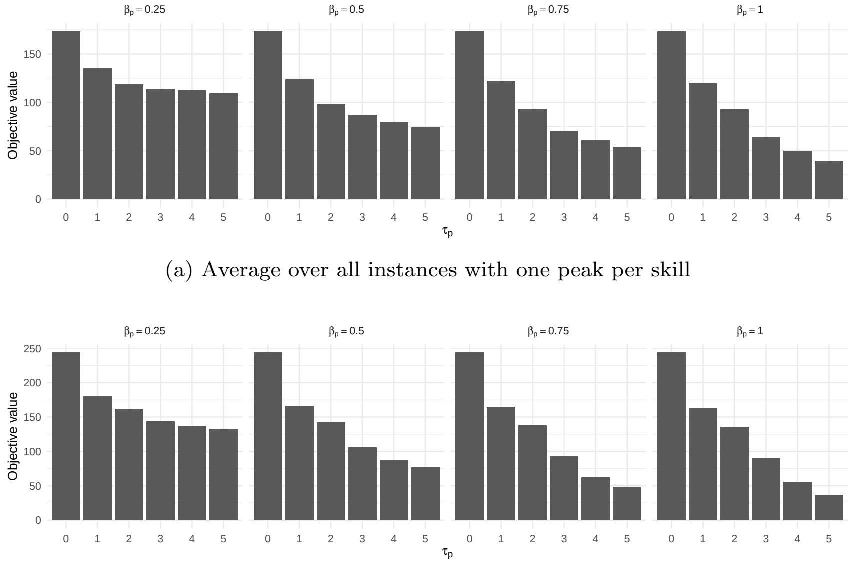
5.2 Demand smoothing flexibility

First, the impact of the parameters τ_p and β_p which control the scope of demand modification is analyzed. Parameter τ_p is varied from one to five while values of 0.25, 0.5, 0.75 and 1 are considered for β_p. Additionally, results are reported for $\tau_p = 0$ and $\beta_p = 0$, which corresponds to solving model (1) - (12) without allowing demand smoothing, that is, solving a standard shift design problem. In all experiments the maximum number of shifts is fixed to $\eta = 15$ and the maximum fraction of short shifts is fixed to $\phi = 0.3$.

Figure 6 shows the impact on the objective value, averaged over all instances, of varying τ_p and β_p for instances with one or two peaks per skill. Note that, for different values of β_p, the setting with $\tau_p = 0$ always results in the same average objective value. Indeed, demand smoothing cannot occur when there are no intervals defined in which to redistribute the demand. In general, given a value of β_p, increasing τ_p results in solutions with a lower objective value, that is, increasing the number of intervals in which demand may be redistributed leads to less over- and understaffing. Even when setting $\tau_p = 1$ and $\beta_p = 0.25$, which provides very little flexibility, over- and understaffing is reduced by 22.1% and 26.1% for the single and double peak instances, respectively. In a high-flexibility setting with $\tau_p = 5$ and $\beta_p = 1$, reductions of 77.1% and 85.0% are realized. The impact of allowing demand smoothing in more intervals around a peak is stronger for higher values of β_p. Comparing the results for instances with one peak per skill to those with two peaks per skill, the improvements in objective value are typically larger when two peaks are present. Organizations suffering from high variability in their workload would thus benefit more from enabling flexibility in their demand patterns.

The change in objective value is clearly related to the number of modifications that have been made to the demand. Figure 7 shows the total number of modifications $\sum_{t \in T_p^{\downarrow}} m_{tp}^{\downarrow} + \sum_{t \in T_p^{\uparrow}} m_{tp}^{\uparrow}$ relative to the modification *budget*[2], that is, the maximum allowed number of modifications $\sum_{p \in P} 2 \times \beta_p \times \sum_{t=t_p^{\text{start}}}^{t_p^{\text{end}}} r_{tk_p}$. The results show that increasing the value of τ_p allows the available budget to be better utilized. A usage of 100% is never realized, even when combining the smallest budget $\beta_p = 0.25$ with the largest redistribution range $\tau_p = 5$, indicating that for the considered instances, setting $\beta_p = 0.25$ is already too flexible considering the number of modifications which actually contribute

[2] Budget is not expressed as a monetary cost, but rather as a parameter without unit. Section 6 presents an analysis of the monetary costs associated with modifications.



(a) Average over all instances with one peak per skill

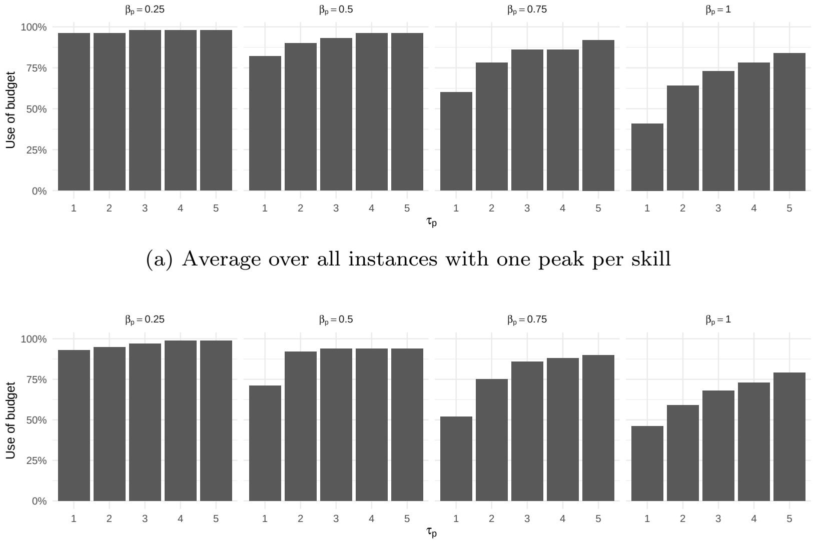
(b) Average over all instances with two peaks per skill

Fig. 6: Objective values for different values of β_p and τ_p with $\eta = 15$ and $\phi = 0.3$

towards improving over- and understaffing. If in practice, costs would be associated with setting the modification budget, these results demonstrate that it is only useful to provide a large budget if the range in which demand may be modified is sufficiently large.

For instances with short shifts, Figure 8 shows the number of short shifts used $\sum_{s \in S^{\text{short}}} y_s$ as a fraction of the maximum number that could be used $\left\lfloor \phi \sum_{s \in S} y_s \right\rfloor$. Recall that in these experiments the maximum number of shifts η was fixed to 15, which was also the total number of shifts used in all solutions. The maximum fraction of short shifts was fixed to $\phi = 0.3$. On average, the solutions include fewer short shifts than the maximum allowed. For sufficiently large values of β_p and τ_p, considerably fewer short shifts are used compared to the setting in which no demand flexibility is permitted. These results confirm the initial hypothesis that by allowing demand smoothing, fewer short shifts are required, thereby potentially improving employee satisfaction. While this benefit may be harder to quantify than over- and understaffing, organizations should also take this aspect into account when assessing the trade-off between demand flexibility and solution quality.

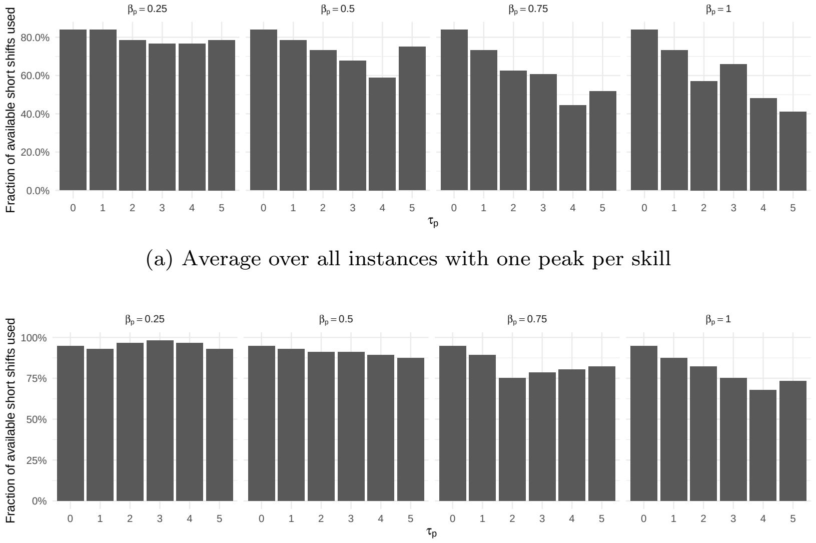
In addition to affecting solution quality, β_p and τ_p also impact computational hardness of the integer programming problems. Table 5 shows the number of instances solved to optimality (out of 20), the average computation time for these instances and the average optimality gap for the instances which were not solved to optimality. Even without allowing modifications to the de-



(a) Average over all instances with one peak per skill

(b) Average over all instances with two peaks per skill

Fig. 7: Use of modification budget for different values of β_p and τ_p with $\eta = 15$ and $\phi = 0.3$



(a) Average over all instances with one peak per skill

(b) Average over all instances with two peaks per skill

Fig. 8: Fraction of short shifts used for different values of β_p and τ_p with $\eta = 15$ and $\phi = 0.3$

mand, optimality gaps of 18.2% and 7.1% are reported for instances with one and two peaks, respectively. Allowing demand smoothing negatively affects the performance of the integer programming solver. Both the computation times and optimality gaps are significantly higher when $\beta_p > 0$ and $\tau_p > 0$. For low values of τ_p, computation time and optimality gap are unaffected by increasing β_p. Similarly, for $\beta_p = 0.25$, increasing the number of intervals in which modifications are allowed has no significant impact. However, the average computation time for instances with two peaks per skill is decreased by 71.6% when increasing τ_p from four to five if $\beta_p = 1$.

Table 5: Average computation time in seconds, number of instances solved to optimality and average gap to lower bound, averaged over all instances for varying values of β_p and τ_p with $\eta = 15$ and $\phi = 0.3$

One peak per skill																		
β_p	τ_p																	
	0			1			2			3			4			5		
	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap
0	10	349.4	18.2%	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0.25	-	-	-	10	479.7	39.9%	9	1062.5	34.4%	9	989.5	32.4%	9	567.6	33.9%	10	434.9	35.5%
0.50	-	-	-	9	1036.8	45.6%	8	472.8	38.8%	16	192.6	39.6%	8	202.4	34.8%	9	767.7	37.4%
0.75	-	-	-	10	483.6	48.5%	8	946.7	44.1%	8	981.2	39.0%	7	98.4	36.2%	9	224.6	39.4%
1.00	-	-	-	10	361.0	46.7%	9	817.1	46.5%	9	184.5	39.9%	8	319.6	39.6%	10	264.4	38.3%

Two peaks per skill																		
β_p	τ_p																	
	0			1			2			3			4			5		
	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap
0	11	738.7	7.1%	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0.25	-	-	-	9	787.5	34.5%	9	245.0	33.4%	11	306.4	25.4%	10	332.2	31.4%	10	377.9	22.8%
0.50	-	-	-	8	560.6	42.9%	8	293.0	50.9%	18	808.1	40.6%	11	1116.4	42.2%	12	413.6	30.5%
0.75	-	-	-	8	201.0	42.7%	7	922.9	55.2%	9	593.1	48.5%	10	1068.2	62.6%	16	376.9	50.5%
1.00	-	-	-	8	576.8	43.2%	7	124.5	54.2%	10	922.2	52.8%	18	829.2	50.5%	19	235.9	100.0%

Table 6 provides details on the computational results for each instance with $\tau_p = 3$ and $\beta_p = 0.5$. Results are reported for the two irregular variants of each instance as well as for the variant without demand peaks. There is a subset of instances, MS05 to MS12, which require significantly less computation time compared to the other instances. From Table 4 it is clear that these problems are small in the sense that they have fewer shifts to chose from compared to the other instances, even though the number of skills is higher or comparable to some of the other harder instances. This trend may be observed for both instances with regular and irregular demand. Note that while the relative gaps appear large, the absolute gaps are small and would, in practice, correspond to solutions in which only few intervals suffer from a limited amount of over- or understaffing.

Table 6: Computational results for all three instance classes with $\tau_p = 3$, $\beta_p = 0.5$, $\eta = 15$ and $\phi = 0.3$

	No peaks			One peak per skill			Two peaks per skill		
	Time (s)	UB	Gap	Time (s)	UB	Gap	Time (s)	UB	Gap
MS01 - day3	1806.8	1	0.0%	537.7	4	0.0%	5969.5	0	0.0%
MS02 - day4	443.9	1	0.0%	985.4	0	0.0%	1090.0	0	0.0%
MS03 - day4	394.1	0	0.0%	16.9	0	0.0%	232.0	0	0.0%
MS05 - day6	46.8	48	0.0%	7200.0	65	3.1%	0.0	122	0.0%
MS06 - day2	59.5	190	0.0%	1.9	341	0.0%	0.3	476	0.0%
MS07 - day5	0.0	143	0.0%	0.8	234	0.0%	0.5	388	0.0%
MS11 - day2	0.0	90	0.0%	0.0	113	0.0%	0.2	229	0.0%
MS12 - day1	0.0	113	0.0%	0.3	183	0.0%	0.6	209	0.0%
MS12 - day2	0.3	89	0.0%	0.3	145	0.0%	0.5	214	0.0%
MS13 - day0	7200.0	91	48.4%	7200.0	68	30.9%	7200.0	48	2.1%
MS15 - day4	7200.0	91	37.4%	7200.0	86	11.6%	7200.0	61	6.6%
MS16 - day3	7200.0	104	35.6%	7200.0	79	15.2%	7200.0	93	15.1%
MS18 - day2	7200.0	132	31.8%	7200.0	114	14.9%	7200.0	100	8.0%
MS19 - day1	7200.0	30	76.7%	7200.0	36	16.7%	7200.0	11	36.4%
MS19 - day3	7200.0	96	33.3%	7200.0	74	13.5%	7200.0	76	15.8%
MS21 - day0	7200.0	55	85.5%	7200.0	32	75.0%	7200.0	9	11.1%
MS22 - day6	7200.0	91	80.2%	7200.3	69	75.4%	7200.1	12	100.0%
MS23 - day6	7200.0	25	32.0%	7200.0	26	50.0%	7200.0	2	100.0%
MS25 - day5	7200.0	33	81.8%	7200.0	30	80.0%	7200.0	42	52.4%
MS28 - day3	7200.0	53	90.6%	7200.0	41	87.8%	7200.0	22	100.0%
Average	4097.6	73.8	31.7%	4397.2	87.0	23.7%	4324.7	105.7	22.4%

As an example, Figure 9 shows a solution for instance MS21 - day 0 with $\tau_p = 3$ and $\beta_p = 0.5$. Both the original and modified demand patterns for one of the two skills in this problem instance are shown. In addition, the coverage realized by the selected shifts, staffing and breaks is also shown. The peak is clearly smoothed out in the modified pattern, within the limits imposed by the constraints, allowing the available shifts to closely match the demand.

5.3 Shift selection flexibility

The second set of parameters analyzed in the computational study affects shift selection: the maximum number of shifts η and the maximum fraction of

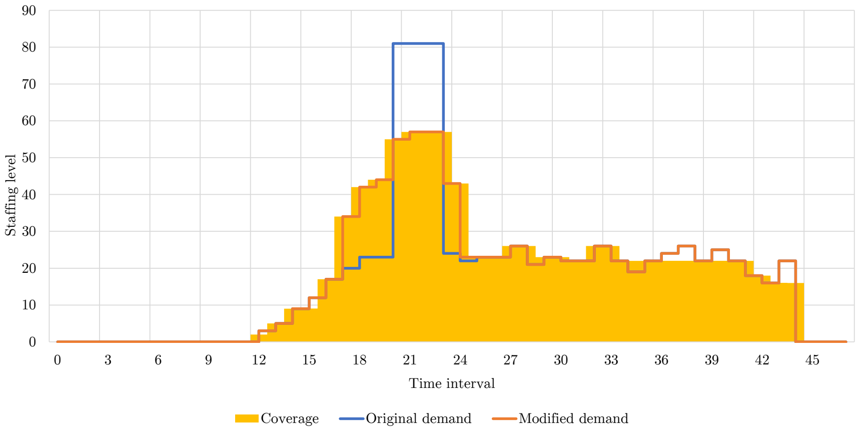


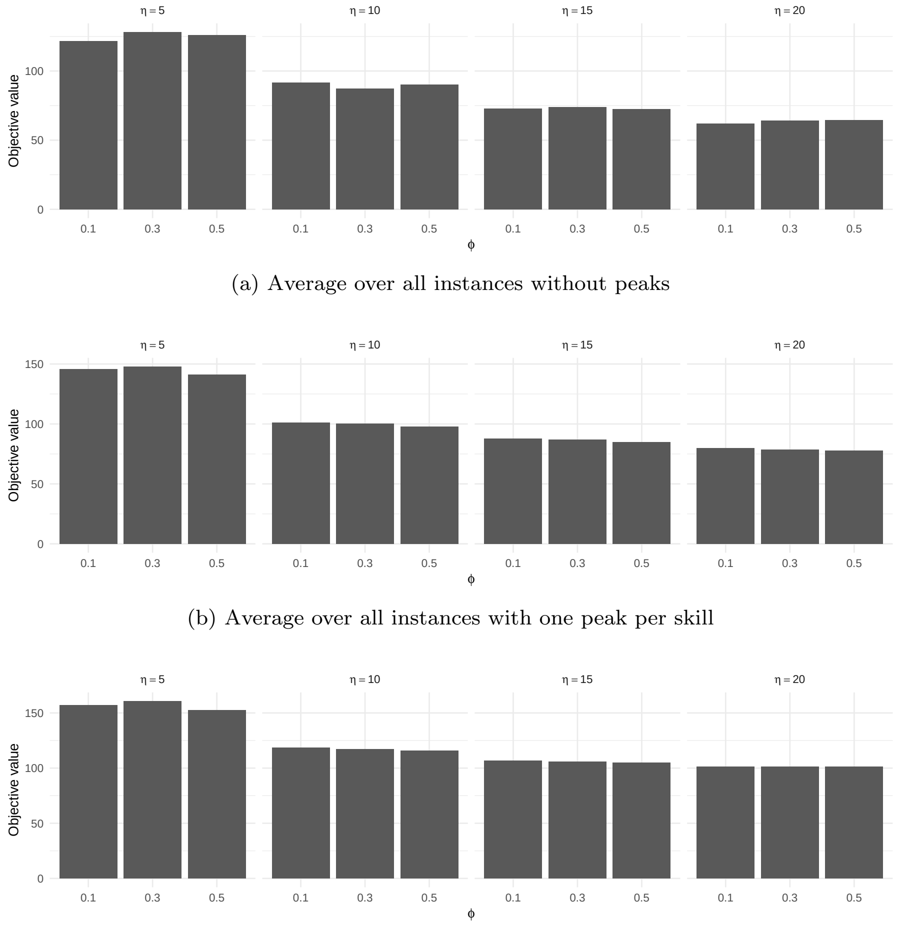
Fig. 9: Coverage, original and modified demand patterns for skill 1 in instance MS21 - day 0 with $\tau_p = 3$, $\beta_p = 0.5$, $\eta = 15$ and $\phi = 0.3$

selected short shifts ϕ. For $\beta_p = 0.5$ and $\tau_p = 3$, ϕ is varied between 0.1 and 0.5 while η is varied from 5 to 20. These ranges were chosen to represent scenarios in which both the number of short shifts and shifts in total are not excessively large as, in practice, having too many different shifts could negatively affect subsequent steps in workforce planning (Burke and Curtois, 2014).

Figure 10 shows the impact of the shift selection parameters on solution quality. The bars represent the objective value, averaged over all instances. These results demonstrate that the main parameter allowing a decrease in over- and understaffing is the maximum number of shifts η. For the considered instances, there is also a diminishing returns-effect: the decrease in objective value becomes smaller for increasingly large values of η. The largest improvement in solution quality is realized by increasing the maximum number of shifts from five to ten. Selecting more shifts could, however, potentially increase the complexity of constructing the actual rosters in later steps of the workforce planning process. The impact of allowing a larger number of short shifts is far less pronounced. An organization could thus better focus their efforts in efficiently managing a large number of shifts rather than re-structuring their workforce to allow more short shifts to be assigned. These effects are observed for instances with regular as well as with irregular demand patterns.

Figure 11 shows the number of short shifts used as a fraction of the maximum allowed for different values of η and ϕ. In general, increasing ϕ leads to fewer short shifts being used. On the other hand, larger values of η typically result in more short shifts. For most settings, however, the number of short shifts always remains well below the maximum allowed, thereby explaining the limited impact of ϕ on the objective value in Figure 10.

Table 7 details the impact of the shift selection parameters on the solver's performance. For each type of instance, the following values are reported: the



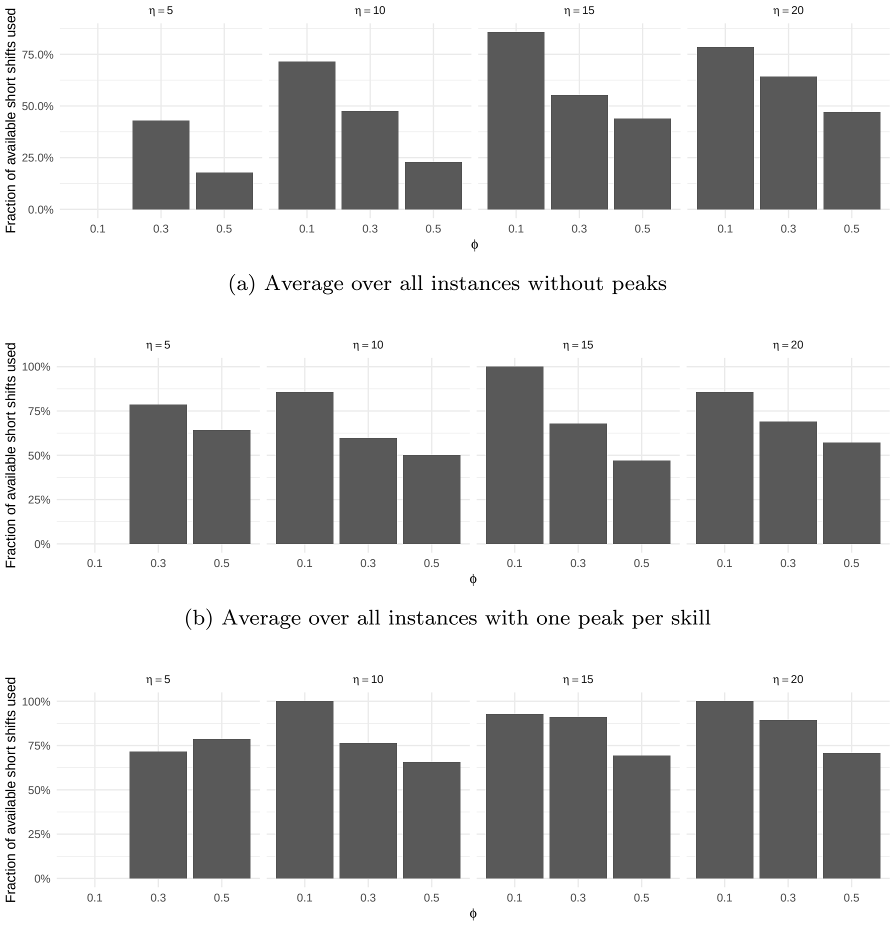
(a) Average over all instances without peaks

(b) Average over all instances with one peak per skill

(c) Average over all instances with two peaks per skill

Fig. 10: Objective values for varying values of η and ϕ with $\beta_p = 0.5$ and $\tau_p = 3$

number of instances solved to optimality (out of 20), the average computation time for these instances and the average optimality gap of the instances which were not solved to optimality. The results indicate that the main driver of the solver's performance is the maximum number of shifts η. Allowing more shifts to be selected increases the number of instances solved to optimality and generally reduces the average computation time for instances where there is no demand peak or only one demand peak per skill. On the other hand, allowing more short shifts to be selected has a less pronounced effect on the solver's performance. In most experiments, the solver's performance was unaffected by changing the value of ϕ. However, for the instances with two peaks per skill,



(a) Average over all instances without peaks

(b) Average over all instances with one peak per skill

(c) Average over all instances with two peaks per skill

Fig. 11: Fraction of short shifts used for varying values of η and ϕ with $\beta_p = 0.5$ and $\tau_p = 3$

increasing ϕ led to more optimal solutions in less computation time when $\eta \geq 15$.

6 Economic impact

As demonstrated in previously published studies, increasing flexibility for employees significantly impacts an organization's labor costs (Erhard, 2019). The computational results reported in Section 5 showed that by increasing flexibility through parameters β_p, τ_p, η and ϕ, significant reductions in objective value can be realized. This section discusses this effect in an economic con-

Table 7: Average computation time, upper bound and gap to lower bound, averaged over all instances for varying values of η and ϕ with $\beta_p = 0.5$ and $\tau_p = 3$

No peaks									
η	ϕ								
	0.1			0.3			0.5		
	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap
5	6	100.4	46.9%	6	100.4	51.9%	6	100.8	50.1%
10	6	155.0	64.3%	6	155.0	61.6%	6	155.6	62.6%
15	9	294.2	54.8%	9	305.7	57.5%	9	972.0	54.4%
20	9	1.3	40.5%	9	1.1	46.7%	9	1.1	45.5%
One peak per skill									
η	ϕ								
	0.1			0.3			0.5		
	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap
5	7	633.6	50.0%	6	233.3	50.1%	7	1073.2	49.8%
10	6	528.4	55.3%	6	522.5	56.6%	6	530.1	53.0%
15	8	147.0	38.9%	8	192.9	39.6%	8	475.9	36.8%
20	12	428.6	30.9%	11	215.5	23.6%	10	18.4	19.3%
Two peaks per skill									
η	ϕ								
	0.1			0.3			0.5		
	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap	No. opt	Time (s)	Gap
5	6	12.7	51.6%	6	12.7	61.6%	6	12.7	62.4%
10	6	1.8	64.1%	6	1.8	66.2%	6	1.8	66.0%
15	8	974.4	39.7%	9	810.4	40.6%	9	390.7	40.5%
20	14	449.4	24.3%	16	672.4	33.5%	16	137.2	51.0%

text to highlight the practical impact of demand smoothing in shift design. The primary result of demand smoothing in terms of an organization's operational expenses is that the required number of employees is reduced by better matching the available shifts to the demand. The number of full-time equivalents (FTEs) is used in this section to estimate the monetary savings achieved through demand smoothing. Based on these savings, an organization can calculate an upper bound on the cost per unit of modification in the demand patterns.

To evaluate the impact of demand smoothing on the required number of FTEs, model (1) - (12) is modified such that understaffing is no longer permitted. Overstaffing is still minimized in the objective function. The updated model was applied to the single-and double-peak variants of instance MS12 (day 1) with $\tau_p = 3$, $\beta_p = 0.5$, $\eta = 15$ and $\phi = 0.3$. Four metrics were calculated from the obtained optimal solutions, as reported in Table 8. First, the total number of assigned FTEs is shown. Using the average Belgian monthly gross salary of €3875 , the total labor cost associated with these FTEs is estimated (Statbel, 2016). The total cost saving from applying demand smoothing is then calculated as the difference between FTE cost without and with smoothing. Note that since the monthly gross salary is used, this corresponds to the savings an organization may realize per month. Finally, the total number of modifications made to the demand patterns is shown.

Using these values, Figure 12 shows the number of months before demand smoothing becomes profitable as a function of the cost required for modifying one unit of demand. In order to arrive at these results, it is assumed that a one-off modification cost is required to alter the demand patterns. Examples of such alterations include upgrading a machine or renegotiating the delivery times of a third-party logistics provider. When considering one peak per skill, if the organization requires profitability after one year then the maximum cost of one unit of modification may be €18,395. However, if the time is restricted to only six months then the maximum cost is €9,197. Clearly, there is a linear relationship between the modification cost and the time before profitability. The results show that the degree of variability in demand has little impact on this length of time. As costs increase, it will require only slightly more time before demand smoothing becomes profitable. However, it is worth noting that this upper bound only takes into account quantifiable criteria. Side-effects of demand smoothing which impact employee well-being, such as fewer short shifts, are not included in this evaluation. The analysis also does not include the additional expenses associated with part-time employees assigned to the short shifts, which would further increase the upper bound on profitability.

Table 8: Impact of demand smoothing on required number of FTEs

	No demand smoothing	Smoothing with $\tau_p = 3$, $\beta_p = 0.5$, $\eta = 15$, $\phi = 0.3$
One peak per skill		
No. of FTEs assigned	132	96
Total FTE cost per month	€511,500	€372,000
Monthly savings from demand smoothing	-	€139,500
No. of modifications	-	91
Two peaks per skill		
No. of FTEs assigned	144	96
Total FTE cost per month	€558,000	€372,000
Monthly savings from demand smoothing	-	€186,000
No. of modifications	-	93

7 Conclusions

State-of-the-art models for shift design assume that demand may be covered by the available shift types. However, many practical applications have highly irregular demand patterns caused by variability in workload. Applying existing models from the academic literature to such problems leads to high levels of under- and overstaffing, which severely impacts quality of service, or ex-

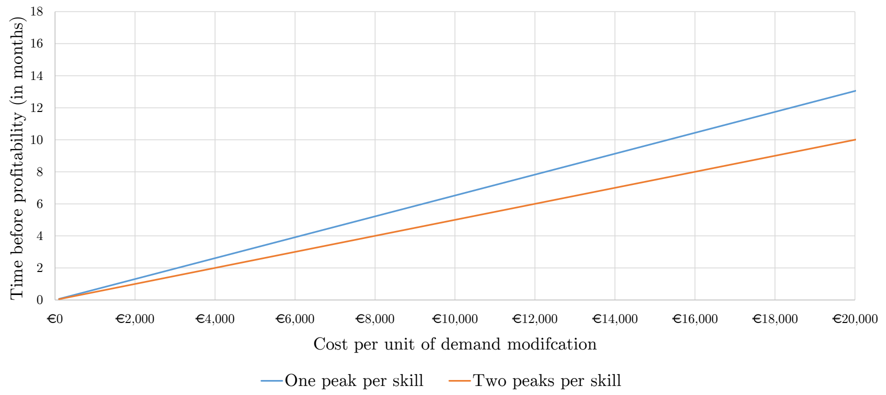


Fig. 12: Time before profitability of demand smoothing as a function of cost per demand modification

cessive use of short shifts, which typically violates various organizational and legislative regulations and may conflict with employee preferences.

This paper introduces a model for the shift design problem in which variable demand is accommodated by allowing minor modifications to the demand patterns through smoothing. The model includes a set of constraints related to operational restrictions on the permitted demand modifications and shift selection. An integer programming formulation for this new shift design problem was presented.

The trade-off between solution quality and flexibility, as controlled by the model's main parameters, was analyzed in a computational study which led to the following main insights. First, increasing the permitted amount of demand redistribution is only useful when the range in which modifications are allowed is sufficiently large. While this effect may be intuitive, the amount of quality improvement achieved by only slightly relaxing the demand constraints was impressive. Reductions in over- and understaffing of up to 26% were reported in the computational study. Second, fewer short shifts are required when the amount of demand smoothing is increased. Third, reduction of over- and understaffing is primarily driven by the maximum number of shifts in total, and not by the maximum number of short shifts. This last effect is somewhat counter-intuitive and may improve employees' employment satisfaction due to the possibility of eliminating short shifts whenever appropriate. An analysis of the computational results showed that significant cost savings may be realized through demand smoothing by reducing the number of required FTEs.

The model introduced in this paper considered the shift design problem for a single day. A natural extension of the model would be to include multiple days such that shift selection, staffing and break scheduling decisions are made for a longer scheduling period. Given the computational hardness of the single-day problem, effective exact approaches or heuristics may be required. The relevance and ubiquitous nature of irregular demand patterns warrant further investigation from a theoretical point-of-view as well, specifically on how to accommodate irregular workload in the context of shift design. For example, is it possible to identify worst cases for demand peaks, and what would their impact on a solution be? Finally, this paper considers a deterministic shift design problem. As this is typically a long-term decision problem in which some parameters may vary from day to day, a stochastic model in which shift selection, staffing and break scheduling are addressed for a typical day also represents a relevant direction for future research.

Acknowledgements The authors are grateful to the anonymous referees for their valuable suggestions and comments on earlier versions of the paper. Editorial consultation provided by Luke Connolly (KU Leuven). This research was supported by the Strategic Basic Research project Data-driven logistics (S007318N), funded by the Research Foundation Flanders (FWO).

References

Akkermans A, Post G, Uetz M (2019) Solving the shift and break design problem using integer linear programming. Annals of Operations Research DOI 10.1007/s10479-019-03487-6

Asgeirsson EI (2014) Bridging the gap between self schedules and feasible schedules in staff scheduling. Annals of Operations Research 218(1):51–69

Baltes BB, Briggs TE, Huff JW, Wright JA, Neuman GA (1999) Flexible and compressed workweek schedules: A meta-analysis of their effects on work-related criteria. Journal of Applied Psychology 84(4):496

Bonutti A, Ceschia S, De Cesco F, Musliu N, Schaerf A (2017) Modeling and solving a real-life multi-skill shift design problem. Annals of Operations Research 252(2):365–382

Boonstra-Hörwein K, Punzengruber D, Gärtner J (2011) Reducing understaffing and shift work with Temporal Profile Optimization (TPO). Applied ergonomics 42(2):233–237

Burgess PA (2007) Optimal shift duration and sequence: recommended approach for short-term emergency response activations for public health and emergency management. American journal of public health 97(Supplement 1):S88–S92

Burke EK, Curtois T (2014) New approaches to nurse rostering benchmark instances. European Journal of Operational Research 237(1):71–81

Buzacott JA, Mandelbaum M (2008) Flexibility in manufacturing and services: achievements, insights and challenges. Flexible Services and Manufacturing Journal 20(1):13

Clausen T (2010) Airport Ground Staff Scheduling. PhD thesis, Technical University of Denmark

De Bruecker P, Van den Bergh J, Beliën J, Demeulemeester E (2015) Workforce planning incorporating skills: State of the art. European Journal of Operational Research 243(1):1 – 16

Defraeye M, Van Nieuwenhuyse I (2016) Staffing and scheduling under non-stationary demand for service: A literature review. Omega 58:4 – 25

Di Gaspero L, Gärtner J, Kortsarz G, Musliu N, Schaerf A, Slany W (2007) The minimum shift design problem. Annals of Operations Research 155(1):79–105

Di Gaspero L, Gärtner J, Musliu N, Schaerf A, Schafhauser W, Slany W (2010) A hybrid LS-CP solver for the shifts and breaks design problem. In: Hybrid Metaheuristics: 7th International Workshop, HM 2010, Vienna, Austria, October 1-2, 2010. Proceedings, Springer Berlin Heidelberg, pp 46–61

Erhard M (2019) Flexible staffing of physicians with column generation. Flexible Services and Manufacturing Journal

Ernst A, Jiang H, Krishnamoorthy M, Sier D (2004) Staff scheduling and rostering: A review of applications, methods and models. European Journal of Operational Research 153(1):3 – 27

Gärtner J, Kundi M (2005) The length of shifts in the service sector–its relation to scheduling working days. Ergonomia: An International Journal of Ergonomics and Human Factors

Green LV, Kolesar PJ (1995) On the accuracy of the simple peak hour approximation for markovian queues. Management Science 41(8):1353–1370

Green LV, Kolesar PJ, Whitt W (2007) Coping with time-varying demand when setting staffing requirements for a service system. Production and Operations Management 16(1):13–39

Gross CN, Fügener A, Brunner JO (2018) Online rescheduling of physicians in hospitals. Flexible Services and Manufacturing Journal 30(1):296–328

Herbers J (2006) Representing labor demands in airport ground staff scheduling. In: Operations Research Proceedings 2005, Springer, pp 15–20

van Hulst D, den Hertog D, Nuijten W (2017) Robust shift generation in workforce planning. Computational Management Science 14(1):115–134

Hur Y, Bard JF, Frey M, Kiermaier F (2019) An investigation of shift and break flexibility with real-time break assignments using a rolling horizon approach. Flexible Services and Manufacturing Journal 31(1):174–211

Komarudin, Guerry MA, De Feyter T, Vanden Berghe G (2013) The roster quality staffing problem a methodology for improving the roster quality by modifying the personnel structure. European Journal of Operational Research 230(3):551 – 562

Litvak E, Buerhaus PI, Davidoff F, Long MC, McManus ML, Berwick DM (2005) Managing unnecessary variability in patient demand to reduce nursing stress and improve patient safety. The Joint Commission Journal on Quality and Patient Safety 31(6):330 – 338

Llort N, Lusa A, Martínez-Costa C, Mateo M (2019) A decision support system and a mathematical model for strategic workforce planning in consultancies. Flexible Services and Manufacturing Journal 31(2):497–523

Love RR, Hoey JM (1990) Management science improves fast-food operations. Interfaces 20(2):21–29

Lusby RM, Range TM, Larsen J (2016) A benders decomposition-based matheuristic for the cardinality constrained shift design problem. European Journal of Operational Research 254(2):385 – 397

Maenhout B, Vanhoucke M (2013) Reconstructing nurse schedules: Computational insights in the problem size parameters. Omega 41(5):903 – 918

Meisen P, Meisen T, Recchioni M, Schilberg D, Jeschke S (2016) Modeling and processing of time interval data for data-driven decision support. In: Automation, Communication and Cybernetics in Science and Engineering 2015/2016, Springer, pp 923–940

Moore CA, Stanton HM, Ashford N (1996) Airport Operations. McGraw-Hill Education

Musliu N, Schaerf A, Slany W (2004) Local search for shift design. European Journal of Operational Research 153(1):51 – 64

Smet P, Ernst AT, Vanden Berghe G (2016) Heuristic decomposition approaches for an integrated task scheduling and personnel rostering problem. Computers & Operations Research 76:60–72

Statbel (2016) The Belgian statistical office. `https://statbel.fgov.be`, accessed: 2019-07-10

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

van der Veen E, Hurink JL, Schutten J, Uijland ST (2016) A flexible iterative improvement heuristic to support creation of feasible shift rosters in self-rostering. Annals of operations research 239(1):189–206

Widl M, Musliu N (2010) An improved memetic algorithm for break scheduling. In: Hybrid Metaheuristics: 7th International Workshop, HM 2010, Vienna, Austria, October 1-2, 2010. Proceedings, Springer Berlin Heidelberg, pp 133–147