

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

## Journal of Scheduling manuscript No. (will be inserted by the editor)

**TITLE:** Cyclic staff scheduling: optimization models for some real-life problems  
**AUTHORS:** Marta Rocha; Jose Fernando Oliveira; Maria Antónia Carraville;  
**SOURCE:** JOURNAL OF SCHEDULING, VOLUME: 16, ISSUE: 2, PAGES: 231-242, PUBLISHED: 2013  
**DOI:** <http://dx.doi.org/10.1007/s10951-012-0299-4>

# Cyclic staff scheduling: optimization models for some real-life problems

Marta Rocha · José Fernando Oliveira · Maria Antónia Carraville

Received: date / Accepted: date

**Abstract** In this work we propose a general integer programming model to address the staff scheduling problem, flexible enough to be easily adapted to a wide range of real-world problems. The model is applied with slight changes to two case studies: a glass plant and a continuous care unit, and also to a collection of benchmark instances available in the literature. The emphasis of our approach is on a novel formulation of sequence constraints and also on workload balance, which is tackled through cyclic scheduling. Models are solved using the CPLEX solver. Computational results indicate that optimal solutions can be achieved within a reasonable amount of time.

**Keywords** Staff scheduling · Rostering · Integer programming · Sequence constraints · Workload balance

## 1 Introduction

Workforce or staff scheduling is a complex and time-consuming problem to deal with in any organization. Which employee shall be doing which task? And when? The answer to these questions must satisfy several constraints concerning demand requirements, task skills specifications, legal or contractual obligations, employees' preferences, among others. The objective to achieve is, usually, to minimize costs or workforce size.

The staff scheduling problem is often decomposed into several sub-problems, which are then tackled independently or in combination. In this paper we address two sub-problems. The first one is the definition of the days off (and consequently the work days) for each employee in the whole planning horizon. The second is the definition of the shifts to assign to every employee in each of the working days. These two sub-problems are generally referred in literature as the days-off and the shift scheduling problems, respectively. When tackled together, they are classified as the tour scheduling problem (Morris and Showalter (1983)), which is easily identifiable with organizations with continuous operation, 24 hours a day, 365 days a year and it has been widely explored in literature (Alfares (2004)). Staff schedules, and tour schedules in particular, can still be classified as cyclic or acyclic (non-cyclic), depending on whether all employees have the same schedule, though lagged in time, or not.

This work studies the staff scheduling problem in a continuous 24 hour environment. The workforce considered is single skilled, the lengths and the starting times of the shifts are fixed and the minimum and maximum daily staff levels required for each shift are known in advance. We propose a mixed integer programming model to simultaneously assign shifts and days-off to each employee. A general model is first developed and then applied with minor adjustments to the problem of a glass manufacturing plant, to the problem of a continuous care unit and also to a collection of 20 benchmark instances. Two main concerns of this model are to ensure a balanced and equitable schedule between all employees, in terms of workload, and also to respect a predefined sequence of shifts and days-off, either following work rules or employees' preferences. The main

---

M. Rocha · J.F.Oliveira · M.A.Carraville  
 INESC TEC, Faculdade de Engenharia, Universidade do Porto, Rua Dr. Roberto Frias, s/n, 4200-465 Porto, Portugal  
 Tel.: +351 22 508 14 00  
 Fax: +351 22 508 14 40  
 E-mail: marta@fe.up.pt

contribution of this work is the way these two issues are tackled and formulated. The first concern is overcome, initially, through the leveling of the number of days that each team works in each shift, as imposed by the objective function defined for the general model and in a next phase, through the imposition of equal schedules to all employees but with a time lag, of a predefined number of days, between them. This feature gives a cyclic dimension to the schedule. The second condition is achieved through the formulation of an array of indices that, together with the definition of a maximum and a minimum number of consecutive days, enable the imposition of any desired pattern of working shifts and days-off. This formulation also makes it possible to control the periodicity of days-off, as well as the length of the tour or sub-period of the planning horizon. The definition of the planning period is not a very explored issue in the literature, since it is closely related to demand forecast periods and it is often an input parameter. But the initially set planning period may not be the one that gives the best solution and so, it is pertinent to study which is the "ideal" planning period for a specific instance. This experience was performed in the tests carried out during this work. The model developed in this work is, therefore, general and flexible, with several degrees of freedom and with the capacity of being easily applied to different staff scheduling problems but at the same time with a cyclic feature that ensures the equitableness and predictability of the schedule.

Staff scheduling is one of the major concerns of human resource management (Enz (2009)). Even nowadays, staff scheduling is still done manually in several activity sectors, absorbing time and resources that could be used more efficiently with automatic scheduling generators. It is not only a matter of reducing costs, but yet a matter of finding a solution that better fits cost minimization, compliance with work and legal rules, satisfaction of individual preferences of the employees, an evenly workload distribution between employees, among other constraints. In addition to this complexity in terms of constraints and objectives, staff scheduling problems usually have particular characteristics, becoming difficult to adapt models without significant reformulation (Ernst et al (2004b)). The evaluation or comparison of schedules/rosters is not, therefore, a simple exercise. Some authors have worked on this issue, mainly in the nurse rostering scope. De Causmaecker and Vanden Berghe (2010a, 2010b) propose a framework for the classification of personnel rostering problems in services that considers three categories: personnel environment, which includes different types of personnel constraints

and skills; work characteristics, which refers to coverage constraints and shift types; and optimization objective. It is a classification based on the problems' characteristics, more than on the models or solutions approaches. In an earlier work, Warner (1976) identifies an interesting set of indicators to measure schedules' performance in terms of: coverage, quality, stability, flexibility, fairness and cost. Coverage measures how close the solution fits the demand requirements. The quality of a roster indicates how well the schedule matches the employee's request or wish, while fairness is a measure of how the employee feels about his/her schedule when compared to the schedules of the other employees. Stability is related to predictability, and cost is a measure of resource consumption in developing the schedule. The impact that a schedule's attribute such as predictability, stability, quality or fairness, as defined above, can have on the productivity and engagement of an employee can be very significant (Glass and Knight (2010)). Short periods of rest and long periods of work, inadequate distribution between rest and work periods, non-standard working shifts are some of the stressful factors that can negatively affect the mental and physical health of employees, as explored by Totterdell (2005). The design of schedules shall then take these issues into consideration, when seeking simultaneously to optimize costs and to respect other kind of constraints.

This paper is organized as follows. In the next section, some literature review is pointed out. In section 3 we present the general model that was developed, describing the problem into consideration, and we emphasize some singular features that aim to make it a valid contribution to this field of research. Sections 4, 5 and 6 present the adaptation of the general model, firstly to a glass industry, secondly to a continuous care unit and lastly to the examples collected in the literature. The problems are introduced and the mathematical formulations are described. Section 7 reports the computational results.

## 2 Literature review

A literature review on staff scheduling and rostering problems, presented by Ernst et al (2004a), shows that transportation, nurse scheduling or call-centers are within the most explored application areas. A comprehensive review work on nurse scheduling is presented by Burke et al (2004), while Gans et al (2003) review studies on telephone call centers. Several variations of the staff scheduling problem can be found in literature, considering: different employees' skills and availability (Loucks and Jacobs (1991)), different contract types (Brusco

and Johns (1996), Bard et al (2003)), employees with different productivity levels (Thompson and Goodale (2006)), staff mixed skills and weekend off requirements (Rong (2010)), hierarchical rules (Ulusam Seçkiner et al (2007)), work shifts flexibility in terms of starting-times (Ni and Abeledo (2007)) or in terms of breaks (Bechtold and Jacobs (1990), Rekik et al (2009)), among others.

Integer programming (IP) is among the most used formulations for modeling the tour scheduling problem (Alfares (2004), Ernst et al (2004a)). Blöchliger (2004) develops a tutorial for modeling staff scheduling problems using IP. Typical techniques are based on the general set-covering model developed by Dantzig (1954). Bard et al (2003) address the staff scheduling problem of a postal service company, which includes full and part-time staff as well as variable shift starting-times. Eitzen et al (2004) develop a set of three IP based methodologies (column generation, column subset and branch-and-price) for solving multiskilled workforce scheduling problems in dynamic environments, with variable workforce size and individual preferences. Emphasis is given to assuring equity between schedules of the employees with the same skills, which is achieved by means of a score levels assignment.

While coverage requirements are typically formulated as hard constraints, workload balance and sequence constraints are often treated as soft constraints, i.e., they can be violated, though at a defined cost added to the objective function. Goal programming or multi-objective techniques are used to incorporate these constraints into the IP scheduling models. Deviations from desired patterns of shifts, patterns of working and rest days, ratio between number of night and days shifts or other requirements are penalized in the objective function, which seeks the minimization of the sum of the weighted deviations (see, for example, the work of Topaloglu and Ozkarahan (2004), Azaiez (2005) or Burke et al (2010b)). With these formulations the user can analyze the impact of different weights to each of the goals. This sensitivity analysis can be very helpful in supporting the decision of choosing the most convenient from a set of feasible solutions. Sequence constraints have alternatively been formulated with network flow models (Moz and Pato (2004)). When an optimal solution is not mandatory, metaheuristics, such as Tabu Search (Burke et. al (1999, 2001)) and Genetic Algorithms (Aickelin and White (2004)), as well as Constraint Programming (Abdennadher and Schlenker (1999)) are alternative approaches that have been widely used to address consecutiveness and workload balance constraints. The combination of different techniques is

becoming popular, since it can take advantage of the best characteristics of the used methods. Examples of the so called hybrid approaches are described in Qu and He (2009), Valouxis and Housos (2000) or Sellmann et al (2000). A randomized greedy procedure is proposed in Carrasco (2010) to balance the workload in a long-term (annual) planning horizon. Employees' preferences are not considered in that case, what decreases the complexity of the problem. Hyperheuristics are a more recent technique that uses a high-level strategy to generate low-level heuristics, or to choose between existing ones, at each decision point, in the search for a better quality solution. It takes advantage of the best features of the selected heuristics and overcomes their limitations, having a more general application than any single heuristic, which is typically customized to a particular problem. A deep insight on this topic can be found in Burke et al (2010a). Examples of the application of hyperheuristics to nurse scheduling and to a home care scheduling problem are reported in Burke and Soubeiga (2003) and Misir et al (2010), respectively.

Cyclic scheduling problems have been tackled by some authors: Baker (1976) reviews mathematical programming formulations for the shift scheduling and the days-off problems; Balakrishnan and Wong (1990) propose a novel formulation based on a network model; a constraint programming algorithm is proposed by Laporte and Pesant (2004). Heuristics and metaheuristics based methods have also been used to solve the cyclic scheduling problem, as in the work of Mora and Musliu (2004) and Musliu (2006). In cyclic scheduling all employees have the same schedule but lagged in time. Enumeration of all feasible patterns (or segments) of working shifts/days and days off is often a common method in the construction of cyclical schedules to overcome sequence restrictions (Laporte (1999)). Laporte (1999) suggests the manual designing of cyclical schedules, arguing that integer programming formulations are too rigid to be applicable to real-world problems and defending therefore the creative break of some rules in order to achieve "workable solutions". In problems of non-cyclic nature, cyclical approaches are avoided because of their apparent inflexibility to deal with unexpected changes in schedules (absences, etc.), but they guarantee the balance and fairness of the schedule, in terms of workload distribution and days off.

### 3 General model

#### 3.1 Problem description

The general model was developed for the staff scheduling problem of an organization that works continuously, 24 hours a day. The day is divided in  $nS$  working shifts. The model considers a set of  $nT$  teams of homogeneous (single skilled and full-time) employees, that must be assigned to either a work or a break shift, in each of the  $nD$  planning period days. Daily shift demand levels must be satisfied, meaning that the model must guarantee a required number of teams working in each shift on each day. Work rules include a minimum and a maximum number of consecutive working days for each team, as well as a predefined sequence of working shifts to be respected. Each shift change must have a break or non-working day in between. The objective is to minimize and to level the number of days each team works in each shift, in order to balance the workload.

#### 3.2 Mathematical model

The following notation was defined:

##### Indices

$d \in \{1, \dots, nD\}$ , day;

$t \in \{1, \dots, nT\}$ , team;

$s \in \{1, \dots, nS\}$ , working shift;

$s' \in \{1, \dots, 2 \times nS\}$ , extended shift;

Extended shifts  $s' \in \{nS+1, \dots, 2 \times nS\}$  are non-working shifts that carry the information on the last working shift of the team;

$n(s')$  is the extended shift that follows the extended shift  $s'$  in a given sequence;

For example, considering 3 working shifts  $\{1, 2, 3\}$  and 3 non-working shifts  $\{4, 5, 6\}$  a possible sequence could be 1-4-2-5-3-6-1-4-..., defined as follows:

|         |   |   |   |   |   |   |
|---------|---|---|---|---|---|---|
| $s'$    | 1 | 2 | 3 | 4 | 5 | 6 |
| $n(s')$ | 4 | 5 | 6 | 2 | 3 | 1 |

The indices  $t$  and  $d$  should take values in a circular list. The list for index  $d$  should for instance be  $\{1, \dots, nD-1, nD, 1, \dots, nD-1, nD, \dots\}$ . For implementation purposes index  $d$  should be replaced by  $[(d-1) \bmod (nD)] + 1$  and index  $t$  should be replaced by  $[(t-1) \bmod (nT)] + 1$ .

##### Parameters

|             |   |
|-------------|---|
| $nT$        | number of teams;                            |
| $nS$        | number of shifts;                           |
| $nD$        | number of days in the planning cycle;       |
| $demands_s$ | daily demand for each working shift $s$ ;   |
| $maxD$      | maximum number of consecutive working days; |
| $minD$      | minimum number of consecutive working days; |

##### Decision variables

$$x_{ts'd} = \begin{cases} 1 & \text{if team } t \text{ is assigned to shift } s' \text{ on day } d \\ 0 & \text{otherwise} \end{cases}$$

##### Decision variables (auxiliary)

$$b_{tdm} = \begin{cases} 1 & \text{if team } t \text{ works at least } minD \text{ consecutive days,} \\ & \text{starting on day } d+m-1 \\ 0 & \text{otherwise} \end{cases}$$

##### Objective function

$$\min \max_{ts} \sum_d x_{tsd} \quad (1)$$

##### Linearized objective function

$$\min Z \quad (2)$$

##### Constraints

$$\forall_{ts} \sum_d x_{tsd} - Z \leq 0 \quad (3)$$

$$\forall_{td} \sum_{s'} x_{ts'd} = 1 \quad (4)$$

$$\forall_{sd} \sum_t x_{tsd} \geq demands_s \quad (5)$$

$$\forall_{td} \sum_{q=0}^{maxD} \sum_s x_{ts(d+q)} \leq maxD \quad (6)$$

$$\forall_{td} \sum_{m=1}^{minD} b_{tdm} - \sum_s x_{ts(d+minD-1)} \geq 0 \quad (7)$$

$$\forall_{td} \forall_{m=1}^{minD} \sum_s \sum_{q=m}^{m+minD-1} x_{ts(d+q-1)} - minD \times b_{tdm} \geq 0 \quad (8)$$

$$\forall_{ts'd} x_{ts'd} - x_{ts'(d+1)} - x_{tn(s')(d+1)} \leq 0 \quad (9)$$

$$\forall_{ts'd} x_{ts'd}, b_{tdm} \in \{0, 1\} \quad (10)$$

The objective function seeks the minimization of the maximum number of days that a team works in each shift. It levels the working days of each team, leading to a solution in which each team works the same number of days in each shift. The linearization of (1) results in the linear objective function expressed in (2), where  $Z$  represents the maximum number of days that a team works in each shift, and also in Equations 3.

Equations 4 state that each day every team has exactly one shift assigned, either a working shift or a break shift.

Equations 5 are coverage constraints, making sure that each shift daily requirements are fulfilled.

Equations 6 ensure that no team works more than  $maxD$  consecutive days. For each day  $d$  a window of length  $maxD + 1$  is opened and at least one of the corresponding  $x_{tsd}$  must be 0, independently of the working shift  $s$ .

Equations 7 and 8 guarantee that each team works at least  $minD$  consecutive working days. The second term on the left-hand-side of equation 7 sums-up the working days  $x_{ts(d+minD-1)}$  within a window of width  $minD$ , starting at  $d$ . If all  $x_{ts(d+minD-1)}$  are zero no constraint is imposed to the variables  $b_{tdm}$ . However, if at least one  $x_{ts(d+minD-1)} = 1$  then at least one of the variables  $b_{tdm}$  must be equal to 1. When the variable  $b_{tdm}$  equals zero, the corresponding equation 8 is fulfilled. However if equation 7 imposes that a variable  $b_{tdm}$  equals one, then the first term on the left-hand-side of equation 8 has to sum-up at least  $minD$ , i.e. the team has to work at least  $minD$  consecutive days. The meaning of  $m$  is that if a team works one day within a window of width  $minD$ , then it has to work at least  $minD$  consecutive days, starting at  $m = 1$  or  $m = 2$  or ... or at  $m = minD$ . Figure 1 illustrates this process.

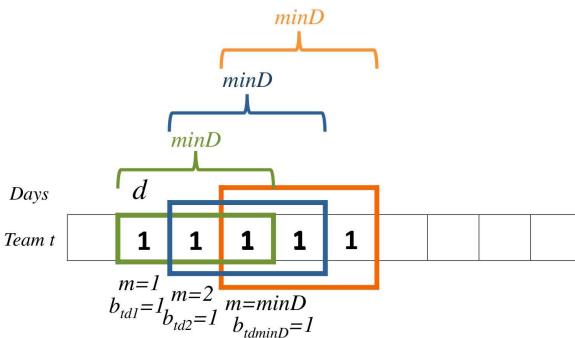


Fig. 1 Illustration of Equations 7 and 8

Equations 9 ensure that the required shift sequence is followed. The basic sequencing requirement is defined over the working shifts that follow the sequence: 1, 2, 3, 1 ..., but, as there are breaks between the working shifts, the breaks must carry the memory of the last working shift. This is obtained through the "extended shift"  $s'$ . For instance, if a team has an extended shift  $s' = 4$  assigned, it means that the team is having a breaking shift after a working shift 1. If the same shift is assigned on days  $d$  and  $d + 1$ , then the corresponding constraint 9 is satisfied independently of the value of  $x_{tn(s')(d+1)}$ . However if the shift ends, i.e. a different shift is assigned on days  $d$  and  $d + 1$ , then the next possible shift is imposed by the vector of indices  $n(s')$  and the equation 9.

### 3.3 Special features

Emphasis must be given to the wide scope and flexibility introduced with the formulation of the sequence shift restriction (constraint 9). Any desired sequence pattern of working shifts and days-off can be imposed through the proper definition of the vector of indices  $n(s')$ .

The limits on the maximum and minimum number of consecutive days for each shift enable the distinction between the length of the working and rest periods, but also between the work shifts' length itself. Some activities have work rules that impose different maximum allowable numbers of consecutive working shifts, for instance night vs day shifts. But those parameters, together with the shift sequence constraints, also allow to control the periodicity of days-off, as well as the length of the tour or sub-period or sub-cycle of the planning horizon. It is possible to impose a schedule with sub-cycles of equal length (if it is a divisor of the planning horizon) or give the model flexibility to construct sub-cycles with different lengths.

## 4 Homogeneous workforce model

### 4.1 Problem description

In the base case for the glass industry, the workforce was distributed in 4 teams of employees. However, during the summer vacation period, they had to hire additional teams of temporary staff to keep the production running. This temporary staff had to be trained beforehand during some weeks, but the company couldn't avoid having a higher number of quality problems during the summer period. Besides that, the schedules for

the 4 teams of fixed staff were very inconvenient in this period. For all these reasons they wanted to make some studies with a higher number of teams of fixed staff.

The facility works 365 days per year, 24 hours a day, in three different eight-hour shifts: M - morning (from 5:00 a.m. to 1:00 p.m.), A - afternoon (from 1:00 p.m. to 9:00 p.m.) and N - night (from 9:00 p.m. to 5:00 a.m.). All teams are homogeneous in terms of skills and contract types, everyone works full-time. Shift daily demand is fixed and known in advance: exactly 1 team for each shift. A mandatory shifts sequence must be respected (M-N-A) and there must be a rest shift (B) between each working shift change (M-B-N-B-A-B). There is a predefined minimum and maximum number of allowed consecutive working days.

#### 4.2 Mathematical model

In order to adapt the general model to this new problem, the following adjustments were made.

##### Indices

$n(s')$  is the extended shift that follows the extended shift  $s'$  in a given sequence;

Considering the 3 working shifts {M,A,N} as {1, 2, 3} and the 3 non-working shifts {4, 5, 6} the sequence M-B-N-B-A-B is now defined as follows:

|         |   |   |   |   |   |   |
|---------|---|---|---|---|---|---|
| $s'$    | 1 | 2 | 3 | 4 | 5 | 6 |
| $n(s')$ | 4 | 5 | 6 | 3 | 1 | 2 |

##### Parameters

$\delta D$  offset between the working cycles of the teams (in number of days).

##### Constraints

$$\forall_{sd} \sum_t x_{tsd} = 1 \quad (11)$$

$$\forall_{t=1}^{nT-1} \forall_{s'd} x_{ts'd} - x_{(t+1)s'(d+\delta D)} = 0 \quad (12)$$

Equations 11 make sure that the daily requirements are fulfilled.

Equations 12 impose that all teams have the same schedule, but with a time lag of  $\delta D$  days between them. With this additional constraint the objective function could be replaced by a constant number but the computational experiments showed that keeping the objective function lowers the computational time.

## 5 Heterogeneous workforce model

### 5.1 Problem description

The problem of the continuous care unit brought to light the challenge to adapt the general model to a service environment. This organization works continuously, around the clock, 24h a day, in a multi-shift working scheme: M - morning (from 8:00 a.m. to 2:00 p.m.), A - afternoon (from 2:00 p.m. to 8:00 p.m.), N - night (from 8:00 p.m. to 0:00 a.m.) and D - after-night (from 0:00 a.m. to 8:00 a.m.). The demand is now different for each shift and the maximum and minimum number of consecutive working (or rest) days is now indexed to each shift. Again, no daily meal breaks are considered, as well as weekends-off restrictions. Employees' preferred sequence of shifts and breaks (B) must be assured (M-A-N-D-B) and preference is given to a balanced schedule between employees. The workforce, still single skilled, is now heterogeneous, combining a fixed workforce of 49 full-time, permanent, employees with a variable pool of part-time workers. The objective of the model is now to minimize the part-time requirements, assuming that full-time contracted hours must be as fully assigned as possible.

### 5.2 Mathematical model

In order to adapt the general model to this new problem, the following adjustments are proposed.

##### Indices

$n(s')$  is the extended shift that follows the extended shift  $s'$  in a given sequence;

Considering 4 working shifts {1, 2, 3, 4} and 1 non-working shift {5} the defined sequence M-A-N-D-B is now:

|         |   |   |   |   |   |
|---------|---|---|---|---|---|
| $s'$    | 1 | 2 | 3 | 4 | 5 |
| $n(s')$ | 2 | 3 | 4 | 5 | 1 |

$maxD_{s'}$  maximum of consecutive working (to shifts 1 to 4) and rest (shift 5) days for each shift;

$minD_{s'}$  minimum of consecutive working (to shifts 1 to 4) and rest (shift 5) days for each shift;

$ptCost_s$  hourly cost of a part-time employee working in shift s;

$h_s$  number of working hours of shift s.

##### Parameters

$\delta D$  offset between the working cycles of the teams (in number of days).

6 *Objective function* Minimization of the cost with part-  
 7 time work.

8  $\min \sum_d \sum_s ptCost_s \times h_s \times (demand_s - \sum_t x_{tsd})$

9 *Constraints*

10  $\forall_{sd} \sum_t x_{tsd} \leq demand_s \quad (13)$

11 Equations 13 state that each day, the number of full-  
 12 time employees assigned to every working shift is less  
 13 than or equal to the demand. The difference between  
 14 the assigned and the demanded work will be assured by  
 15 part-time workers.

16 The constraints concerning the minimum and the  
 17 maximum number of consecutive days must also be  
 18 adjusted in order to consider the indexed parameters  
 19  $maxD'_s$  and  $minD'_s$ .

20 **6 Benchmark instances**

21 **6.1 Problems description**

22 In order to evaluate the flexibility and wide ranging  
 23 scope of the proposed formulation and to compare re-  
 24 sults with other approaches, experiences were carried  
 25 out on a collection of 20 rotating workforce scheduling  
 26 problems presented in Musliu (2006), and available in  
 27 <http://www.dbai.tuwien.ac.at/staff/musliu/benchmarks>.  
 28 In these instances, the number of employees vary from  
 29 7 to 163. The number of standard shifts is either 2 (day  
 30 and afternoon) or 3 (day, afternoon and night). The  
 31 length of working and days-off blocks is now limited  
 32 by a minimum and a maximum number of consecutive  
 33 days, as well as the length of each sequence of days as-  
 34 signed to the same shift. In the previous case studies  
 35 only this last situation is considered. The main differ-  
 36 ence from these problems to the previous ones is the ex-  
 37 istence of a set of forbidden shift sequences, instead of  
 38 a predefined sequence to follow. Not allowed sequences  
 39 imposed are of two types: (N-D, N-A and A-D) or (N-  
 40 B-N, A-B-D, N-B-A and N-B-D). Naturally, the second  
 41 type is considered only if single day-off is allowed. Oth-  
 42 erwise, only the first type is taken into account. Prob-  
 43 lems have either both types of not allowed sequences or  
 44 only type 1.

45 **6.2 Mathematical model**

46 The adaptation of the general model to these problems  
 47 imposes constraints on the length of working and days-  
 48 off blocks and especially adjustments in the sequence

49 constraints. Following the same perspective used in the  
 50 previous case studies, it is possible to define allowable  
 51 sequences for each given type of not allowed sequences.

52 Consider the 3 working shifts {D,A,N} as {1,2,3}  
 53 and the 3 non-working shifts as {4,5,6}. An allowable  
 54 sequence would be: D-B-A-B-N-B, i.e., 1-4-2-5-3-6. But,  
 55 in opposition to the previous case studies, this is not the  
 56 only sequence allowed for the present problems.

57 For the first type of not allowed sequences: N-D, N-  
 58 A and A-D, the allowed sequences can be defined as  
 59 follows:

**Table 1** First type of allowable sequences

| $s'$    | 1 | 2 | 3 | 4 | 5 | 6 |
|---------|---|---|---|---|---|---|
| 4       | 4 | 5 | 6 | 1 | 1 | 1 |
| 2       | 2 | 3 |   | 2 | 2 | 2 |
| $n(s')$ | 3 |   |   | 3 | 3 | 3 |
| 5       |   |   |   | 5 | 4 | 4 |
| 6       |   |   |   | 6 | 6 | 5 |

60  $n(s')$  is the extended shift that can follow the ex-  
 61 tended shift  $s'$  in a given sequence.

62 From Table 1, it is possible to realize that only the  
 63 shifts that follow the shifts 2 and 3 must be imposed.  
 64 All the remaining shifts can be followed by any of the  
 65 other shifts, inclusive by themselves. The sequence con-  
 66 straints in the general mathematical model are replaced  
 67 by the following constraints:

68  $\forall_{td} x_{t2d} - x_{t2(d+1)} - x_{t5(d+1)} - x_{t3(d+1)} \leq 0 \quad (14)$

69  $\forall_{td} x_{t3d} - x_{t3(d+1)} - x_{t6(d+1)} \leq 0 \quad (15)$

70 Considering now both types of not allowable se-  
 71 quences: N-D, N-A, A-D, N-B-N, A-B-D, N-B-A and  
 72 N-B-D, the allowed sequences to consider are now de-  
 73 fined as follows:

**Table 2** First and second types of allowable sequences

| $s'$    | 1 | 2 | 3 | 4 | 5 | 6 |
|---------|---|---|---|---|---|---|
| 4       | 4 | 6 | 4 | 1 | 1 | 2 |
| 2       | 2 | 3 |   | 2 | 2 | 3 |
| $n(s')$ | 3 |   |   | 3 | 3 | 4 |
| 5       |   |   |   | 5 | 4 | 5 |
| 6       |   |   |   | 6 | 6 | 6 |

Note that, in this case, shift 3 (N) can only be followed by another shift 3 or by a minimum of two consecutive days-off (B). To simplify the model and in order to satisfy this limitation, shift 4 is now defined as the non-working shift that follows shift 3 and, in the model's parameterization, the minimum number of consecutive days ( $minD'_s$ ) indexed to shift 4 is set to 2. The sequence constraints in the general mathematical model are now restricted to the constraints on shifts 2, 3 and 6, defined as follows:

$$\forall_{td} x_{t2d} - x_{t2(d+1)} - x_{t6(d+1)} - x_{t3(d+1)} \leq 0 \quad (16)$$

$$\forall_{td} x_{t3d} - x_{t3(d+1)} - x_{t4(d+1)} \leq 0 \quad (17)$$

$$\forall_{td} x_{t6d} - x_{t6(d+1)} - x_{t2(d+1)} - x_{t3(d+1)} - x_{t4(d+1)} \leq 0 \quad (18)$$

For the problems with only two shifts {1,2}, the only forbidden sequence is A-D, and so the allowed sequences are now defined as follows:

**Table 3** Allowable sequences for nS=2

| $s'$ | 1 | 2 | 3 | 4 |
|------|---|---|---|---|
| 2    | 2 | 4 | 2 | 1 |
| 3    | 3 | 3 | 4 | 3 |
| 4    |   | 4 | 1 | 2 |

The sequence constraints in the general mathematical model are replaced by the following constraints:

$$\forall_{td} x_{t2d} - x_{t2(d+1)} - x_{t4(d+1)} - x_{t3(d+1)} \leq 0 \quad (19)$$

## 7 Results

Models were coded in OPL Studio version 6.3 and solved using the CPLEX 12.1.0 solver on a server machine powered by 2 Intel® Xeon® processors of 2.4 GHz and 1.39 GHz, and with 2 GB RAM.

### 7.1 Homogeneous workforce model

Tests were performed considering 5 teams and 3 working shifts. The maximum ( $maxD$ ) and minimum ( $minD$ ) number of consecutive working days were set to 4 and 2, respectively. A set of different planning periods and offsets combinations were tested. The experience showed that this problem is extremely tight in terms of relationships between parameters. There is no place for large

parameters' variations and variables are strongly connected, what makes the space of feasible solutions very limited. The only combinations that always guarantee the existence of a feasible solution are those that verify the following condition:

$$\delta D = \frac{nD}{nT}$$

The offset parameter ( $\delta D$ ) must be equal to the ratio between the number of days of the planning period ( $nD$ ) and the number of teams ( $nT$ ). Putting it simple, what this means is that the offset parameter must be the one that divides the planning period in a number of sub-periods (or sub-cycles) equal to the number of teams. Table 4 reports the computational time results and model size, in terms of number of decision variables and constraints, for a set of different planning periods.

**Table 4** Model size and computational times

| $nD$ | $\delta D$ | No. Decision Variables | No. Constraints | Time (sec.) |
|------|------------|------------------------|-----------------|-------------|
| 30   | 6          | 900                    | 2640            | 0.44        |
| 35   | 7          | 1050                   | 3080            | 0.44        |
| 60   | 12         | 1800                   | 5280            | 2.64        |
| 70   | 14         | 2100                   | 6160            | 7.78        |
| 90   | 18         | 2700                   | 7920            | 80.94       |
| 180  | 36         | 5400                   | 15840           | 184.02      |

As an example, the solution found for a 35 days (5 weeks) planning period in 0.44 seconds is shown in Figure 2. This schedule can either be repeated as many times as the company wants, or can be integrated with schedules of other length, in order to build a longer (yearly) plan. The summer holidays special period was also taken into account in a subsequent phase. Considering a planning period of 16 days, with only 4 working teams (the fifth is on holidays), a holidays schedule was developed, that could be easily integrated in the 5 working teams' schedule with some calendar adjustments. The offset parameter gives a cyclic dimension to the schedule, since every team/employee is allocated to exactly the same sequence of working shifts and days-off, but with a time lag of  $\delta D$  days between each other. This feature allows an evenly distribution of workload among employees. In this case, any unexpected adjustment to the schedule will be done within each team of employees, having little or no impact in the long-term schedule.

| Days | T1 | T2 | T3 | T4 | T5 |
|------|----|----|----|----|----|
| 1    | M  | A  | N  | B  | B  |
| 2    | B  | A  | N  | M  | B  |
| 3    | B  | A  | N  | M  | B  |
| 4    | N  | A  | B  | M  | B  |
| 5    | N  | B  | B  | M  | A  |
| 6    | N  | M  | B  | B  | A  |
| 7    | N  | M  | B  | B  | A  |
| 8    | B  | M  | A  | N  | B  |
| 9    | B  | B  | A  | N  | M  |
| 10   | B  | B  | A  | N  | M  |
| 11   | B  | N  | A  | B  | M  |
| 12   | A  | N  | B  | B  | M  |
| 13   | A  | N  | M  | B  | B  |
| 14   | A  | N  | M  | B  | B  |
| 15   | B  | B  | M  | A  | N  |
| 16   | M  | B  | B  | A  | N  |
| 17   | M  | B  | B  | A  | N  |
| 18   | M  | B  | N  | A  | B  |
| 19   | M  | A  | N  | B  | B  |
| 20   | B  | A  | N  | M  | B  |
| 21   | B  | A  | N  | M  | B  |
| 22   | N  | B  | B  | M  | A  |
| 23   | N  | M  | B  | B  | A  |
| 24   | N  | M  | B  | B  | A  |
| 25   | B  | M  | B  | N  | A  |
| 26   | B  | M  | A  | N  | B  |
| 27   | B  | B  | A  | N  | M  |
| 28   | B  | B  | A  | N  | M  |
| 29   | A  | N  | B  | B  | M  |
| 30   | A  | N  | M  | B  | B  |
| 31   | A  | N  | M  | B  | B  |
| 32   | A  | B  | M  | B  | N  |
| 33   | B  | B  | M  | A  | N  |
| 34   | M  | B  | B  | A  | N  |
| 35   | M  | B  | B  | A  | N  |

Fig. 2 Case study 1 example - schedule for  $nD=35$  days and  $\delta D=7$

## 7.2 Heterogeneous workforce model

In the second case study, the number of employees ( $nT$ ) is 49 and the working shifts ( $nS$ ) are now 4. The daily shift demand ( $demand_s$ ) is 20 M, 17 A, 11 N and 11 D. Tests were conducted for planning periods of 25, 28 and 30 days, considering different combinations of the parameters  $minD_s$ ,  $maxD_s$  and  $ptCost_s$ . The choice of these parameters' values took into account the desired length of the sub-periods, the assurance of the minimum of one day-off every 7 days, and also that the working hours assigned to each employee should fall below 160h in a 30-day period. The reasoning made in case study 1, concerning the value of the offset parameter, does not make sense in this problem, since the planning horizon

is now shorter than the number of employees. Therefore, the offset was set to 1, as it achieved satisfactory results.

The problem size is now larger than in the previous case study, with the number of decision variables varying between 12250 for  $nD=25$  and 14700 for  $nD=30$  and the number of constraints reaching the maximum of 41310 for  $nD=30$  and  $maxD_s=minD_s=[2\ 1\ 1\ 1]$ . Table 5 reports the computational results for a set of different values of input parameters. The column "solution pattern" illustrates the schedule for one employee for the whole planning horizon considered. It is noticeable the ability to control the solution pattern with the variation of input parameters. For a 25 days horizon, for instance, it is possible to get a balanced solution, with equal length sub-periods, with a reduction of  $maxD_s$ , forcing the model to assign exactly 1 day for each shift. In this solution the number of sub-periods increases (from 4 to 5), meaning that the number of breaks or days-off of full-time employees will also increase, as well as the requirements for part-time service (higher/poorer solution value). The results of the tuning of  $maxD_s$  and  $minD_s$  can also be checked in for a 30 days planning horizon. In this case, fixing to 2 the minimum number of consecutive days of shift M, results in a balanced solution but with the same number of sub-periods and, therefore, with the same solution value. The influence of the  $ptCost_s$  parameter can be verified in the 28 days planning horizon case. The increase in the cost of a part-time night and after-night shifts, from 1 to 3 units, results in a higher number of sub-periods and therefore, in a higher number of days off of full time employees and higher part-time needs, leading to a worse solution. The same happens in the  $nD=30$  case, where a raise in the cost of the part-time night and after-night shifts from 1 to 1000 achieves a solution with 2 more sub-periods, which means more part-time requirements and a consequently a worse solution value. In terms of execution times, all runs fall bellow 6.1 seconds. Figure 3 shows the schedule for  $nD=28$  days,  $maxD_s=[2\ 2\ 1\ 1]$ ,  $minD_s=[1\ 1\ 1\ 1]$  and  $ptCost_s=[1\ 1\ 1\ 1]$ .

## 7.3 Benchmarking instances

Tests were performed for the 20 instances and results are shown in Table 6. The offset considered for all the examples is 7 days, so the planning periods are defined by  $7 \times nT$ . The optimization model found the optimal solution for 12 problems, with the number of teams varying from 7 to 29. For the remaining problems, signed with "-", the optimization model did not

**Table 5** Model computational parameters and results

| $nD$ | $maxD_s$<br>[M A N D B] | $minD_s$<br>[M A N D B] | $ptCost_s$<br>[M A N D] | Solution Pattern<br>(sub-periods)   | PT Req.<br>(hours) |
|------|-------------------------|-------------------------|-------------------------|-------------------------------------|--------------------|
| 25   | [2 2 1 1 1]             | [1 1 1 1 1]             | [1 1 1 1]               | MMAANDB-MMAANDB-MMANDB-MANDB        | 2676               |
| 25   | [1 1 1 1 1]             | [1 1 1 1 1]             | [1 1 1 1]               | MANDB-MANDB-MANDB-MANDB-MANDB       | 2970               |
| 28   | [3 3 1 1 1]             | [1 1 1 1 1]             | [1 1 1 1]               | MMMAANDB-MMAANDB-MMAAANDB-MANDB     | 2856               |
| 28   | [2 2 1 1 1]             | [1 1 1 1 1]             | [1 1 1 1]               | MMAANDB-MMAANDB-MMAANDB-MMAANDB     | 2856               |
| 28   | [2 2 1 1 1]             | [1 1 1 1 1]             | [1 1 3 3]               | MAANDB-MANDB-MAANDB-MANDB-MAANDB    | 3150               |
| 28   | [2 2 1 1 1]             | [1 1 1 1 1]             | [1 5 1000 1000]         | MAANDB-MANDB-MAANDB-MAANDB-MANDB    | 3150               |
| 30   | [3 3 1 1 1]             | [1 1 1 1 1]             | [1 1 1 1]               | MMMANDB-MMMAANDB-MMMANDB-MMAAANDB   | 2976               |
| 30   | [2 2 1 1 1]             | [1 1 1 1 1]             | [1 1 1 1]               | MANDB-MANDB-MAANDB-MMAANDB-MMAANDB  | 3270               |
| 30   | [2 1 1 1 1]             | [2 1 1 1 1]             | [1 1 1 1]               | MMANDB-MMANDB-MMANDB-MMANDB-MMANDB  | 3270               |
| 30   | [1 1 1 1 1]             | [1 1 1 1 1]             | [1 1 1 1]               | MANDB-MANDB-MANDB-MANDB-MANDB-MANDB | 3564               |
| 30   | [3 3 1 1 1]             | [1 1 1 1 1]             | [1 1 1000 1000]         | MANDB-MANDB-MANDB-MANDB-MANDB-MANDB | 3564               |

reach any feasible solution within 20000 seconds of running time. These situations are all instances with more than 27 teams. The model is able to reach an optimal solution for problem 16, for example, while for problem 7 it does not find any feasible solution. Although both problems have the same number of teams, 29, the shift daily demand, the limits on the length of shift sequences, working-day and days-off blocks differ from one problem to another. It is evident that the combination of these parameters has a decisive influence on the complexity of each problem and consequently on the model's performance. In Musliu (2006), the best resolution times for these instances are achieved by a tabu search based heuristic (MC-T). In the same work, Musliu compares the resolution times of MC-T with the ones achieved by a commercial software, First Class Scheduler (FCS). The resolution times achieved by Musliu with FCS are also reported in the last column of Table 6, where “-” signs the problems to which FCS did not find any solution within 1000 seconds of running time. The problem explored by Musliu seeks always the satisfaction of all the constraints, what means that when a solution is found that satisfies all the constraints, that solution is optimal. When comparing resolution times, the heuristic based method outperforms the optimization approach in all instances. Nevertheless, when comparing the resolution times obtained by the MIP model and FCS, we can conclude that the former has better times for 7 of these instances than the latter. The comparison of the values of the objective functions proves that the solutions obtained by the MC-T method are all optimal, in spite of the different pattern composition of the schedules.

**Table 6** Computational times for the benchmarking instances using the MIP model, MC-T and FCS

| Ex. | $nD$ | $nT$ | $nS$ | Time(sec.) |        |        |
|-----|------|------|------|------------|--------|--------|
|     |      |      |      | MIP        | MC-T   | FCS    |
| 1   | 63   | 9    | 3    | 7.94       | 0.07   | 0.90   |
| 2   | 63   | 9    | 3    | 2.90       | 0.07   | 0.40   |
| 3   | 119  | 17   | 3    | 907.40     | 0.42   | 1.90   |
| 4   | 91   | 13   | 3    | 1.59       | 0.11   | 1.70   |
| 5   | 77   | 11   | 3    | 2.47       | 0.43   | 3.50   |
| 6   | 49   | 7    | 3    | 1.23       | 0.08   | 2.00   |
| 7   | 203  | 29   | 3    | -          | 52.79  | 16.10  |
| 8   | 112  | 16   | 3    | 7.60       | 0.74   | 124.00 |
| 9   | 329  | 47   | 3    | -          | 15.96  | -      |
| 10  | 189  | 27   | 3    | -          | 0.60   | 9.50   |
| 11  | 210  | 30   | 3    | -          | 13.15  | 367.00 |
| 12  | 140  | 20   | 2    | 310.00     | 1.17   | -      |
| 13  | 49   | 7    | 3    | 255.95     | 0.87   | -      |
| 14  | 91   | 13   | 3    | 73.14      | 0.76   | 0.54   |
| 15  | 448  | 64   | 3    | -          | 159.04 | -      |
| 16  | 203  | 29   | 3    | 1923.00    | 0.54   | 2.44   |
| 17  | 231  | 33   | 2    | 29.64      | 2.16   | -      |
| 18  | 371  | 53   | 3    | -          | 6.83   | 2.57   |
| 19  | 840  | 120  | 3    | -          | 75.83  | -      |
| 20  | 1141 | 163  | 3    | -          | 71.38  | -      |

## 8 Conclusions

Staff scheduling is, typically, an intensive time-consuming and cost-significant task. It represents, therefore, one of the main management concerns for any organization, either from manufacture or service sectors. In this work, the staff scheduling problem was applied to two real-life case studies: a glass industry and a continuous care unit. Despite their apparent dissimilarity, the fact is that these two problems do have common features. A general mixed integer problem was first formulated and then adapted introducing only slight changes. The same model was adjusted to solve a collection of examples of the rotating workforce scheduling problem available

**Fig. 3** Case study 2 example - schedule for  $nD=28$  days and  $\delta D=1$

in the literature. The novelty of the proposed formulation is both the flexible and wide-ranging way in which the sequence conditions are tackled, and the cyclic approach of an initial non-cyclic problem, to ensure the equity and predictability of the schedule. The proposed automatic scheduling approach will not only lead to cost reductions, by an efficient use of its resources, but also to a productivity increase by improving the motivational and the satisfaction levels of its employees, contributing also to the corporate social responsibility. Usually, real-life instances dimension is not compatible with exact methods because of the high number of variables. In this work, however, real-world size instances were solved to optimality. Although low solution times are not a must in this case, computational results revealed that it is possible to reach optimal solutions in an adequate amount of time.

**Acknowledgements** This work is partially funded by the ERDF - European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT - Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within project FCOMP - 01-0124-FEDER-022701.

## References

- Abdennadher S, Schlenker H (1999) Nurse scheduling using constraint logic programming. In: Proceedings of the 11th Conference on Innovative Applications of Artificial Intelligence, pp 838–843

Aickelin U, White P (2004) Building Better Nurse Scheduling Algorithms. *Annals of Operations Research* 128(1-4):159–177

Alfares H (2004) Survey, categorization, and comparison of recent tour scheduling literature. *Annals of Operations Research* 127(1):145–175

Azaiez M (2005) A 0-1 goal programming model for nurse scheduling. *Computers & Operations Research* 32(3):491–507

Baker KR (1976) Workforce Allocation in Cyclical Scheduling Problems: A Survey. *Operational Research Quarterly* (1970-1977) 27(1):155

Balakrishnan N, Wong RT (1990) A network model for the rotating workforce scheduling problem. *Networks* 20:25–42

Bard JF, Binici C, DeSilva AH (2003) Staff scheduling at the United States Postal Service. *Computers & Operations Research* 30(5):745–771

Bechtold S, Jacobs L (1990) Implicit modeling of flexible break assignments in optimal shift scheduling. *Management Science* 36(11):1339–1351

- Blöchliger I (2004) Modeling staff scheduling problems. A tutorial. *European Journal of Operational Research* 158(3):533–542
- Brusco M, Johns T (1996) A sequential integer programming method for discontinuous labor tour scheduling. *European Journal of Operational Research* 95(3):537–548
- Burke E, Soubeiga E (2003) Scheduling nurses using a tabu-search hyperheuristic. In: Proceedings of the 1st Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA 2003), Nottingham, UK, pp 180–197
- Burke E, Causmaecker PD, Berghe GV (1999) A Hybrid Tabu Search Algorithm for the Nurse Rostering Problem. In: et Al BM (ed) Lecture Notes in Artificial Intelligence, vol 1585, Springer, pp 187–194
- Burke E, Cowling P, De Causmaecker P, Berghe G (2001) A memetic approach to the nurse rostering problem. *Applied intelligence* 15(3):199–214
- Burke E, De Causmaecker P, Berghe GV, Van Landeghem H (2004) The State of the Art of Nurse Rostering. *Journal of Scheduling* 7(6):441–499
- Burke E, Hyde M, Kendall G, Ochoa G, Ozcan E, Woodward JR (2010a) A classification of hyper-heuristic approaches. In: Gendreau M, Potvin JY (eds) *Handbook of Metaheuristics*, International Series in Operations Research & Management Science, vol 146, Springer US, pp 449–468
- Burke E, Li J, Qu R (2010b) A hybrid model of integer programming and variable neighbourhood search for highly-constrained nurse rostering problems. *European Journal of Operational Research* 203(2):484–493
- Carrasco RC (2010) Long-term staff scheduling with regular temporal distribution. *Computer methods and programs in biomedicine* 100(2):191–9
- Dantzig GB (1954) Letter to the editor - a comment on edie's "traffic delays at toll booths". *Operations Research* 2(3):339–341
- De Causmaecker P, Vanden Berghe G (2010a) A categorisation of nurse rostering problems. *Journal of Scheduling* 14(1):3–16
- De Causmaecker P, Vanden Berghe G (2010b) Towards a reference model for timetabling and rostering. *Annals of Operations Research* pp 1–10
- Eitzen G, Panton D, Mills G (2004) Multi-Skilled Workforce Optimisation. *Annals of Operations Research* 127(1-4):359–372
- Enz CA (2009) Key issues of concern in the lodging industry: What worries managers. Cornell Hospitality Report 4, The Center for Hospitality Research, School of Hotel Administration, Cornell University
- Ernst AT, Jiang H, Krishnamoorthy M, Owens B, Sier D (2004a) An annotated bibliography of personnel scheduling and rostering. *Ann Oper Res* 127(1-4):21–144
- Ernst AT, Jiang H, Krishnamoorthy M, Sier D (2004b) Staff scheduling and rostering: a review of applications, methods and models. *Eur J Oper Res* 153(1):3–27
- Gans N, Koole G, Mandelbaum A (2003) Telephone call centers: Tutorial, review, and research prospects. *Manufacturing and service operations management* 5(2):79–141
- Glass CA, Knight RA (2010) The nurse rostering problem: A critical appraisal of the problem structure. *European Journal of Operational Research* 202(2):379–389
- Laporte G, Pesant G (2004) A general multi-shift scheduling system. *Journal of the Operational Research Society* 55(11):1208–1217
- Laporte G (1999) The art and science of designing rotating schedules. *The Journal of the Operational Research Society* 50(10):1011–1017
- Loucks J, Jacobs F (1991) Tour scheduling and task assignment of a heterogeneous work force: A heuristic approach. *Decision Sciences* 22(4):719–738
- Misir M, Verbeeck K, De Causmaecker P, Berghe G (2010) Hyper-heuristics with a dynamic heuristic set for the home care scheduling problem. In: Evolutionary Computation (CEC), 2010 IEEE Congress on, IEEE, pp 1–8
- Mora M, Musliu N (2004) Genetic algorithm for rotating workforce scheduling problem. In: Computational Cybernetics, 2004. ICCC 2004. Second IEEE International Conference on, IEEE, pp 121–126
- Morris JG, Showalter MJ (1983) Simple Approaches to Shift, Days-Off and Tour Scheduling Problems. *Management Science* 29(8):942–950
- Moz M, Pato MV (2004) Solving the problem of rerostering nurse schedules with hard constraints: new multicommodity flow models. *Annals of Operations Research* (1976):179–197
- Musliu N (2006) Heuristic Methods for Automatic Rotating Workforce Scheduling. *International Journal of Computational Intelligence Research* 2(4):309–326
- 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
- Qu R, He F (2009) A Hybrid Constraint Programming Approach for Nurse Rostering Problems. In: Allen, T and Ellis, R and Petridis, M (ed) *Applications and Innovations in Intelligent Systems XVI*, Springer-Verlag London Ltd, pp 211–224, 28th SGAI International Conference on Innovative Techniques and Applications

- plications of Artificial Intelligence, Cambridge, England
- Rekik M, Cordeau JF, Soumis F (2009) Implicit shift scheduling with multiple breaks and work stretch duration restrictions. *Journal of Scheduling* 13(1):49–75
- Rong A (2010) Monthly tour scheduling models with mixed skills considering weekend off requirements. *Computers & Industrial Engineering* 59(2):334–343
- Sellmann M, Zervoudakis K, Stamatopoulos P, Fahle T (2000) Integrating Direct CP Search and CP-based Column Generation for the Airline Crew Assignment Problem. In: *Transportation Science*, 24, pp 163–170
- Thompson GM, Goodale JC (2006) Variable employee productivity in workforce scheduling. *European Journal of Operational Research* 170(2):376 – 390
- 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
- Totterdell P (2005) Work schedules. *Handbook of work stress* p 53
- Ulusam Seçkiner S, Gökçen H, KURT M (2007) An integer programming model for hierarchical workforce scheduling problem. *European Journal of Operational Research* 183(2):694–699
- Valouxis C, Housos E (2000) Hybrid optimization techniques for the workshift and rest assignment of nursing personnel. *Artificial intelligence in medicine* 20(2):155–75
- Warner DM (1976) Scheduling nursing personnel according to nursing preference: a mathematical programming approach. *Operations Research* 24(5)