European Journal of Operational Research 249 (2016) 67–75

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Discrete Optimization

Polynomially solvable personnel rostering problems

Pieter Smet*, Peter Brucker[1], Patrick De Causmaecker, Greet Vanden Berghe

KU Leuven, Department of Computer Science, CODeS & iMinds-ITEC, Gebroeders De Smetstraat 1, 9000 Gent, Belgium

ARTICLE INFO

Article history:
Received 8 September 2014
Accepted 17 August 2015
Available online 22 August 2015

Keywords:
Personnel rostering
Polynomial time algorithms
Assignment
Networks

ABSTRACT

Personnel rostering is a personnel scheduling problem in which shifts are assigned to employees, subject to complex organisational and contractual time-related constraints. Academic advances in this domain mainly focus on solving specific variants of this problem using intricate exact or (meta)heuristic algorithms, while little attention has been devoted to studying the underlying structure of the problems. The general assumption is that these problems, even in their most simplified form, are NP-hard. However, such claims are rarely supported with a proof for the problem under study. The present paper refutes this assumption by presenting minimum cost network flow formulations for several personnel rostering problems. Additionally, these problems are situated among the existing academic literature to obtain insights into what makes personnel rostering hard.

1. Introduction

Research on personnel rostering, and personnel scheduling in general, has focused mostly on solving some problem at hand. As a result, a large part of the academic literature details algorithms tailored to one specific problem. Typically, general complexity claims are made, thereby referring to NP-complete problems that *resemble* the problem under discussion. However, in many cases there is no certainty that these complexity claims hold for this particular rostering problem. Theoretical studies on models and complexity of personnel rostering are lacking in the present literature.

There are only a few authors who have formally determined the hardness of a personnel rostering problem. Osogami and Imai (2000) and Brunner, Bard, and Köhler (2013) prove that rostering problems with constraints on the number of assignments of particular shifts, and with constraints on consecutive days worked and days-off are hard. Lau (1996b) describes a shift assignment problem closely related to rostering, and proves its NP-completeness. For restricted variants of the problem, Lau (1996a; 1996b) provide polynomial time algorithms, which are discussed in detail in Section 4.2 of the present paper.

To the best of our knowledge, Brucker, Qu, and Burke (2011) are the only authors to systematically study personnel scheduling from a theoretical point of view. Based on a general mathematical model, four polynomially solvable cases have been identified, two of which are closely related to rostering. The first problem, P_{Thm1}, considers different shifts which require a constant number of employees on different days. The employees are assumed to be available on all days. Without any further restrictions on the assignment of shifts to employees, the problem can be solved as a series of transshipment problems. The second problem, P_{Thm2}, assumes one type of shift, and the availability of employees given by one interval, i.e. employee availability is assumed to be contiguous. There are no other restrictions. A reformulation models the problem as a minimum cost network flow problem.

The present paper identifies new personnel rostering problems that can also be solved in polynomial time. Table 1 compares the two polynomially solvable rostering problems studied by Brucker et al. (2011) with the problems discussed in this work.

In the light of the new contributions, complexity results from the academic literature are revisited to obtain insights into what it is that makes personnel rostering hard. This work provides an update on the current results, and further establishes the foundations for theoretical studies on personnel rostering models.

Even though all results are discussed in terms of *shifts* and *days*, the ideas can be directly transferred to the domain of *tasks* and *periods*. This observation underpins the idea that the presented results have a potential impact not only in different rostering application areas, such as logistics and health care, but also in personnel scheduling in general.

The remainder of this paper is organised as follows. Section 2 introduces basic definitions of concepts in personnel rostering. Section 3 investigates problems with restrictions on the number of

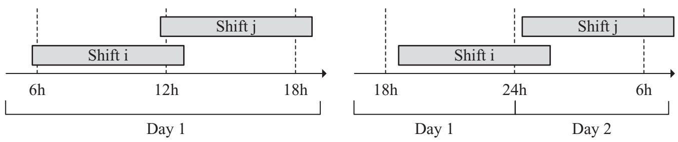
* Corresponding author. Tel.: +3292658704.
E-mail address: pieter.smet@cs.kuleuven.be (P. Smet).

[1] Peter Brucker sadly passed away on July 24, 2013. His coauthors dedicate their contribution in this paper to his memory.

http://dx.doi.org/10.1016/j.ejor.2015.08.025

Table 1
Comparison of characteristics of polynomially solvable rostering problems from Brucker et al. (2011).

	Shifts		Demand		Employee availability			Time-related constraints
	Single	Multiple	Stable	Varying	Full	Contiguous	Varying	
P_{Thm1}	✓	✓	✓		✓			
P_{Thm2}	✓		✓	✓		✓		
This paper	✓	✓	✓	✓	✓	✓	✓	✓



(a) In-day overlap (b) Next-day overlap

Fig. 1. Types of overlap between shifts.

assignments to each employee. Several polynomially solvable cases are identified by formulating them as minimum cost network flow problems. Based on these results, an efficient approach to a known problem from the literature is presented, and the complexity of commonly used benchmark datasets for nurse rostering is discussed. Sections 4 and 5 consider problems with constraints on consecutive assignments. Again, polynomially solvable cases are presented, and linked with results from the literature. For all results, the practical implications are discussed. Finally, Section 6 concludes the paper and identifies areas for future research.

2. Personnel rostering problems

This section introduces common concepts in personnel rostering problems, which will be used throughout the paper.

Employees have to be assigned to shifts in a way that satisfies a variety of constraints. These problems are characterised by a set of employees $E = \{1, \ldots, e\}$, a scheduling period of days $T = \{1, \ldots, t\}$ and a set of shifts $S = \{1, \ldots, s\}$.

A *shift* is a fixed time interval which denotes a working period. Each shift is characterised by a unique type which classifies the shifts in various ways, e.g., by time interval (morning, late), by required qualifications (senior, junior), or by a combination of these (morning-senior, late-junior). A shift is considered to occur on the day where its time interval starts. The number of employees required for each shift can vary from day to day, and is typically more than one employee.

An *assignment* is the allocation of an employee to a shift on a day. A *roster* is an $e \times t$ matrix which contains in each cell either an assignment or a day-off. If the cardinality of S is one, the single shift represents a day-on, and the solution is referred to as a *day-off roster*. This work considers non-cyclic rosters, in contrast to cyclic rosters in which all employees have the same assignments, but lagged in time (Rocha, Oliveira, & Carravilla, 2013).

Two shifts are *in-day overlapping* if their time intervals overlap when considering the shifts on the same day. An ordered set of two shifts is *next-day overlapping* if an employee cannot be assigned to these shifts on consecutive days without overlap of their time intervals. Fig. 1 visualises these concepts. This distinction is important since several models for personnel rostering problems assume that at most one shift can be assigned per day, thereby automatically eliminating in-day overlap, but not necessarily next-day overlap.

Domain constraints define the possible assignments for each employee on each day. For each employee i and day j, a set of shifts $\bar{S}_{ij}$ is defined, consisting of the shifts that can be feasibly assigned. In practice, these constraints can be used to model restrictions such as 'part-time employees can only work 4 hour or 6 hour shifts' or 'an employee does not want to work late shifts on Wednesday'. This concept can also be used to model employee skills by only including shifts in $\bar{S}_{ij}$ for which employee i is qualified.

The *demand* d_{jk} (or coverage requirement) is the required number of employees on day j, shift k. Demand is *stable* if the same number of employees is required on each day and shift, i.e. $\forall j \in T, k \in S : d_{jk} = d$. Furthermore, if on each day and shift only one employee is required, $\forall j \in T, k \in S : d_{jk} = 1$, there is *unit demand*. In contrast, demand is *varying* if d_{jk} can be any non-negative value.

Demand can be expressed as an *exact*, *ranged*, *minimum* or *maximum* requirement. In the case of exact demand, the specified value is exactly the number of employees to be assigned. A ranged definition requires that the number of assigned employees should be within a specified interval. When such an interval has no upper (lower) limit, the requirement is defined as a minimum (maximum).

In addition to the coverage requirements, personnel rostering problems are typically also subject to a variety of contractual time-related constraints, which can be categorised as *counters*, *series* or *successions*. Counters restrict the number of times a specific roster item (e.g. assignments or days-off) can occur within a certain period. Series restrict consecutive occurrences of specific roster items (Smet, Bilgin, De Causmaecker, & Vanden Berghe, 2014). Similar to the coverage requirements, these different types of constraints can be expressed as either ranged, minimum, maximum or exact. Finally, successions denote a special type of series, which restrict occurrences of specific roster items on two consecutive days.

3. Counter constraints

This section presents results for problems with counter constraints. More specifically, constraints on the number of days worked and on the number of shifts worked of each type are discussed. The literature survey of Van den Bergh, Beliën, De Bruecker, Demeulemeester, and De Boeck (2013) illustrates the importance of these two constraints as they were included in 85 and 47 recent academic studies, respectively.

Each subtitle describes the problem discussed. The first two elements describe the number of shifts and type of demand. The last element states the objective, if any. All other elements describe constraints of the problem. The type of definition (exact, range, minimum or maximum) for each constraint is mentioned between parentheses.

3.1. Single shift, varying demand (minimum), number of days worked (exact), feasibility

The *number of days worked* constraint limits the number of assignments per employee in the scheduling period. In practice, this constraint is used to model different contract types. For example, a full time employee will be required to work 20 days in a monthly scheduling period, whereas a part time employee should only work 15 days.

Consider the set of employees E to be homogeneous, i.e. each employee has to work exactly a days. The problem can be formulated as the following integer linear program.

$$x_{ij} = \begin{cases} 1 & \text{if employee } i \text{ works on day } j \\ 0 & \text{otherwise} \end{cases}$$