Journal of Applied Operational Research (2011) **3**(3), 148–162

ISSN 1735-8523 (Print), ISSN 1927-0089 (Online)

Nurse rescheduling with shift preferences and minimal disruption

Alistair R. Clark [1,]* and Hannah Walker [2]

[1] *University of the West of England, Bristol, UK*
[2] *VEGA Group plc, Bristol, UK*

Abstract. Hospital wards must be staffed 24 hours a day by a limited number of nurses. With a well documented shortage of nurses in many countries, effective scheduling of nurse shifts is crucial. Recent research on nurse scheduling has focused on creating flexible schedules that are attractive to nurses, with the joint aims of improving the quality of care and increasing staff retention. To achieve this, some level of preference scheduling is desirable in a nurse scheduling model. Furthermore, if a nurse is unable to work their assigned shifts or nursing cover requirements change, then gaps will occur in the overall schedule that must be filled by rescheduling in a manner that disrupts a little as possible the existing schedule. Little research has been carried out on rescheduling nurses while considering their own preferences as well as minimising disruption. This paper will present some models for nurse scheduling and rescheduling that consider nurses' preferences, along with some preliminary computational results.

Keywords: nurse scheduling; rescheduling; preferences; disruption

Received November 2010. Accepted February 2011

Introduction

In a world where demand for round-the-clock services and higher output has lead to an increase in shift work in a wide variety of industries, the ability to devise a good staff schedule can be crucial to the success of an organisation. The task of scheduling staff is a complicated balancing act between an organisation's needs and the legal & contractual obligations to its employees. Moreover, shift work can have a significant impact on the health and lives of staff, which can in turn affect a person's productivity at work. All such factors should be considered when designing a staff shift schedule, leading to an extremely complex problem for which finding a good solution, within a reasonable time frame, can be difficult. Differences between industries, their goals and restrictions, often mean that specific models and algorithms must be developed for each of them (Ernst *et al.*; 2004). This together with the importance of finding the 'best' schedule has lead to much research covering industries including airlines, call centres and hospitals with a range of solution methods including mathematical programming, heuristic techniques, simulation and artificial intelligence being utilised.

An important area for research and the focus of this paper is the scheduling of nurses. Hospital wards must be staffed 24 hours a day 7 days a week by a limited number of staff, which in itself can be difficult. However, as well as the usual legal & cost constraints, patient care is an important factor. The effects of shift work on the individual are even more important, with patients' lives at risk if nurses are made to work undesirable schedules that may affect

* *Correspondence*: Alistair Clark, Bristol Institute of Technology, University of the West of England, Bristol BS16 1QY England, UK
Email: Alistair.Clark@uwe.ac.uk

their performance in some way. With shortages in qualified nurses reported regularly, good schedules are important to provide satisfactory patient care and potentially improve nurse retention.

Many solutions to the nurse scheduling problem have been proposed along with software that has been shown to provide better solutions and/or solutions in significantly quicker time frames than devising them by hand. Problems arise however, if a member of staff is unable to work their assigned shift and alterations have to be made to cover the shortfall. There are limited options for rectifying the problem with the main ones being the production of a completely new schedule, running a ward understaffed, employing temporary staff, or making changes to the schedule by hand. All of these options can be undesirable for a variety of different reasons including cost and the effect on the quality of patient care. If the need for a change in the schedule arises, then ideally regular staff should be used to fill the gaps in the schedule while keeping to a minimum the disruption for the rest of the staff. On top of this, a reschedule may be required very quickly and so should be produced quickly with very little user input and not compromising patient care.

Section 2 of this paper reviews the available literature on nurse scheduling & rescheduling, focusing on the importance of good nurse scheduling practices and ways to incorporate these into scheduling models. This is followed in section 3 by an explanation of the development of two different scheduling models, the first scheduling individual days, the second scheduling a week at a time using a set of viable shift patterns. Section 4 develops these scheduling models so they can be used for rescheduling following changes to the original schedule's data. In section 5 the two models are compared and conclusions drawn, with a summary of future research directions.

Literature review

Nurse shift scheduling

The variety and depth of manpower shift scheduling applications and techniques in general is not discussed in this paper. The reader is referred to Ernst *et al*. (2004) who provide an extensive review of research in shift scheduling covering many industries and the techniques employed. We will focus on nurse scheduling. Within this scope, the wide-ranging review by Burke *et al*. (2004) evaluates a variety of interdisciplinary approaches to nurse scheduling and discusses whether the literature adequately addresses the main research issues. We will review some of the more recent articles and those particularly relevant to preference scheduling.

While staff scheduling plays an important role in any company that relies on shift work, scheduling nurses has its own unique factors that make the problem both interesting and complex. As with all businesses, costs must be kept to a minimum and contractual obligations adhered to. However, in nursing there are other important factors, for example fatigue. A poorly constructed nurse schedule can lead to fatigued nurses which can in turn lead to mistakes. Rogers *et al*. (2004) reported a significant increase in mistakes made by nurses if shifts of more than 12 hours were worked, when more than 40 hours was worked in a one week period or, if extensive overtime was worked. There is evidence to suggest that the amount a person becomes fatigued when working shifts can be reduced by carefully planning the order in which they rotate through the shifts. The preferred rotating pattern has been shown to be morning, evening and then night shifts. The number of days a particular shift is worked before changing, the length of the break between changing shift and the distribution of days off have also been shown to have an impact on a person's health and productivity at work (Kostreva *et al*.; 2002). A well constructed schedule can help to reduce all these problems which can in turn help reduce absenteeism and the rescheduling problems this then causes.

Regular reports in the news about the shortage of qualified nurses, not just in England but countries all around the world has highlighted the importance for hospitals to retain their staff and good scheduling practices can help with this. McVicar (2003) found that shift working is an increasing source of distress for nurses. The UK's Royal College of Nursing has reported that "Nurses who are forced to work a rotating pattern of day and night shifts are more likely than other nurses to want to leave the National Health Service" (Royal College of Nursing; 2004) supporting the idea of more flexible schedules. Pryce *et al*. (2006) showed that allowing nurses to have some choice about the shifts they work can improve their perception of the shifts which in turn improves morale and has a positive effect on the nursing staff. This not only helps retain staff, but also assists in reducing the number of people calling in sick which can obviously have an impact on already stretched nursing staff. It can also help to reduce the number of temporary nurses employed to cover gaps in schedules, which can be expensive both in monetary terms and the continuity of patient care.

Many of these ideas for improving nurse scheduling have been used in recent research with a tendency to focus on flexibility and giving nurses greater control over their own schedules. This has been introduced into schedules in a number of ways with preference scheduling being adopted to some degree by many researchers (Warner; 1976; De Grano *et al*.; 2009). Preference scheduling has varying degrees of implementation, from simply asking nurses for a consensus about rules that should be implemented or avoided in the schedules, to allowing nurses to have almost complete control over which shifts they work. The preferences are then balanced with other cost factors such as using agency staff to fill the schedule.

One extreme form of preference scheduling is a complete self-scheduling system, where nurses design their own schedules entering onto a sign-up sheet the shifts they want to work. Here the manager's role is reduced to simply ensuring that all shifts are covered by the correct number of nurses. Bailyn *et al*. (2007) reported on a trial of a complete self-scheduling system for a group of nurses carried out over a 12 month period. The system received a positive reception from the nurses involved, with them perceiving they were getting more time with their families and able to provide better patient care. However, the results showed that over the 12 month trial period, an initial decrease in change requests gradually increased back to previous levels and the number of sick calls remained unchanged for the duration of the trial. The reasons for this were not determined by the research but problems in the way the nurses approached the task of choosing their preferred shifts were evident with nurses signing up for shifts that were already fully staffed, signing up for more shifts than they were contracted to work, and not fully understanding the implications of the combinations of shifts they signed up to. This meant that changes had to be made to people's preferences to ensure the right number of nurses were assigned to each shift, leading to a level of dissatisfaction amongst the nurses. These problems also meant that there was always a degree of rescheduling that had to be performed by the nurse manager which meant that although the amount of time the nurse manager spent sorting out schedules reduced, their annoyance with the task did not. Although the study showed some positive elements to a self scheduling system, it also showed a number of problems indicating a compromise between a completely managed schedule and a complete self scheduling system is probably needed.

Such a compromise was reached to an extent by Bard and Purnomo (2005a) and Bard and Purnomo (2005c) who considered preference scheduling. Virtually identical models were used in both instances, but the second paper had developed the model to consider the possibility of downgrading, where a higher qualified nurse can cover the role of a lower qualified nurse but not vice versa. The model used a defined list of violations such as undesirable shift patterns, working overtime, or not assigning a nurse their preferred shifts, and gave each violation a penalty. The aim of the model was to minimise these penalties and to try and spread them evenly amongst the nurses. This lead to an objective function that minimised the sum of the penalties for each nurse, and a maximum penalty being defined which must not be exceeded by any nurse's schedule. The model is initialised with sign-up schedules or, in their absence, an individualised template based on the nurse's contract. A limited number of candidate schedules are then produced in the form of columns. Each candidate schedule satisfies the constraints of the model, minimises the number of violations and is biased towards shifts where gaps have been identified. A limit was placed on the number of schedules produced purely to keep computational time down. One big advantage of this approach is that each nurse gets a 'good' schedule for the planning horizon because it is looked at as a whole for an individual nurse rather than looking at the schedule in segments. The model performed very well when tested on real and derived data with solutions being found in minutes in most cases with very few violations. The use of outside nurses was reduced via higher penalties.

Bard and Purnomo (2007) proposed another nurse scheduling model that was a preference cyclic schedule, "combining the principles of cyclic and preference scheduling in a single model". Due to the cyclical nature of this method it is not possible to accommodate individual preferences. Instead common preferences, such as shift patterns, are built into the model. There are disadvantages to cyclical scheduling: "cyclic schedules have become the bane of the profession because of the rigidity they impose" (Bard and Purnomo; 2005c). However, they are seen as extremely fair because each nurse will work the same schedules over a period of time. In addition, since the schedules only need changing when demand or preferences change, the schedules have to be recalculated less often and nurses know much further in advance what shifts they will be working. Bard and Purnomo's first approach to solving the model was to use an Integer Program and its associated Lagrangian relaxation. Both methods showed little success in the time to find a solution and "because Lagrangian relaxation algorithms only provide lower bounds, an efficient heuristic is needed to construct feasible solutions to the original problem" (Bard and Purnomo; 2007). A heuristic technique was developed that provided good results.

Azaiez and Sharif (2005) considered common preferences and built them into a goal programming model that attempted to balance hospital objectives with nurses' preferences and other soft constraints about ideal shift patterns found in the literature. This led to a large model for which Azaiez and Sharif felt a feasible solution would be difficult to find, so they split the constraints into two groups. The first group contained constraints that had to be satisfied, such as the number of nurses required for each shift. The second group contained the constraints relating to things like shift patterns that did not have to be satisfied, but the amount by which they were not satisfied should be kept to a minimum.

A completely different approach was developed by (Brucker *et al*.; 2005) who used a decomposition, construction and post processing approach to schedule nurses in an intensive care unit. Instead of solving the problem as a whole, this approach broke the problem down into three steps. The first step creates a cyclic schedule using sequences of shifts that were built to satisfy all the hard constraints and minimise the penalty associated with violating the soft constraints. Step two assigns nurses to these schedules making minor adjustments where necessary so that a basic schedule adhering to all the constraints is produced. These first two steps are done by hand and it is the final third step that is an automated neighbourhood search used to improve the schedule. The second and third steps mean that although a cyclic schedule is used to initialise the method, the final schedule is not necessarily cyclic. The authors say that this combines the advantages of cyclic patterns and non-cyclic schedules, which greatly reduce computational time and take care of individual constraints/requests". Computational time for the third step is usually under one minute which is more than adequate but having to do the first two steps manually could be as time consuming as the process already in place.

Aickelin and Dowsland (2004) broke the problem down and then used a genetic algorithm. Their model included the ability to substitute a more qualified nurse for a less qualified one, and aimed to produce schedules seen to be fair by the nurses. Fairness was achieved by meeting as many of the nurse's requests as possible and trying to evenly distribute unpopular shifts. Their algorithm was shown to perform well when compared to live data, striking a good balance of feasibility and quality, with 51 out of 52 of the trials producing an optimal or near optimal solution, and the other 1 achieving feasibility. However, an exact integer programming method proved better giving lower cost and higher feasibility results but had the disadvantage of being much slower than the genetic algorithm. The genetic algorithm also has the advantage that it can solve an unconstrained problem using the decoder to bias the search and that all the problem specific knowledge is held in the decoder so the algorithm can be quickly changed to adapt to changes in problem specification.

Dias *et al*. (2003) carried out research at a hospital in Brazil where the scheduling was being done by hand. The aim was to provide a complete scheduling system that was not just a model but also provided a user interface to produce efficient schedules quickly, minimising the amount of overtime and avoiding undesirable shift patterns. They used an intensive care unit as a representative solution and trialled two solution methods, a tabu search and genetic programming. The schedules that had been produced manually had an average of 20 violations whereas none of the best automatic schedules, using either method, had any violations and the worst automatic schedule had just 4 violations. Such a small difference between the best and worst solutions indicated a very robust algorithm. No improvement was made on the manual schedules where the amount of overtime was concerned, as the manual schedule had the minimum anyway. The algorithms did however help by spreading the overtime throughout the scheduling period better. Neither the genetic nor the tabu schedule was preferred when reviewed by the nurse manager and both needed very few manual changes, mostly due to factors beyond the control of the nurse manager such as late change requests. Since neither method could be shown to outperform the other, both methods were implemented in the hospital with a user interface that allowed the user to choose the solution method.

Wright *et al*. (2006) tackled nurse scheduling from a different perspective, investigating the effect mandatory staff patient ratios introduced in America might have. The research focused on monetary costs and the satisfaction of the staff patient ratio, with minimal consideration given to impacts of undesirable schedules on nurses. This is a little outside the scope of the current study but does provide some interesting conclusions. A similar approach to that used for call centre scheduling was used with the aim to minimise the number of times the ratio is likely to be violated rather than ensuring the ratio is never violated and risking overstaffing. Unlike the papers discussed so far, the model does not just assign nurses to shifts but also calculates the number of nurses required. This was approached in two different ways, the first considered the model in its entirety as a bicriteria non linear integer program solved with a heuristic method. The second method was a relaxed version which decomposes into two sub-problems, a workload model and a tour assignment model solved with an exact method. Tests based on actual data were carried out with the partial model showing that in some instances a small increase in cost could dramatically reduce the number of undesirable shift patterns. The insight gained here is that desirable scheduling policies can be achieved with little or no additional wage costs. This is an important conclusion given the evidence already presented about the effects of good scheduling policies.

The papers discussed here on nurse scheduling are just a fraction of the papers available on the subject, but have been chosen to highlight the diversity of solution methodologies and algorithms that have been developed in an attempt to solve the nurse scheduling problem. However, the objectives for all the papers are similar and tend to focus on not only costs but also the affect of the schedules on the nurses and their lives by looking at shift patterns and preferences. Where these factors were built into the schedule they received a positive response from the nurses with reports of nurse's perceptions improving greatly indicating a good reason to consider these factors when developing a scheduling model.

Nurse rescheduling

Nurse rescheduling aims to minimise changes to the original schedule while minimising costs. Rebuilding the schedule with the current staff is usually the cheaper option as there is no extra wage to pay, but altering the schedule will alter other nurses' schedules as well. This can lead to frustration amongst the nurses since they tend to organise their private lives in accordance with their expected duties, any change in the announced schedules may create personal inconveniences to some of them (Moz and Pato; 2007). This leads to a fine balancing act between costs and changing the schedule as little as possible.

Moz and Pato have produced a number of papers on the nurse rescheduling problem proposing various techniques for solving the problem using a hospital in Lisbon for their research. The aim is to change the original schedule as little as possible. The constraints are ultimately the same as in an original scheduling problem including hard constraints such as shift coverage, each nurse can only work one shift a day, infeasible sequences of shifts and a minimum number of days off. There are also soft constraints such as a maximum number of hours each nurse can work, a night shift should not be worked on consecutive nights, no leave of more than 3 consecutive days should be allowed and no evening shift should be worked for more than 3 consecutive days. Then there is, of course, the extra constraint that makes sure nurses are not assigned to tasks on the days they are absent. In all their research Moz and Pato do not consider the soft constraints but look at improving the solution method to gain better results in quicker times.

Their first study proposed an integer multicommodity flow model (Moz and Pato; 2003). The multicommodity flow model defines a service rendered by each nurse as a commodity and that such commodities share a network. Three sets of nodes are defined, 1 set is the 4 possible shifts, and the other sets are the origin and destination nodes, one node for each nurse where "the aim is to flow the commodities through the network at a minimum cost, while respecting the supply and demand on the nodes". Both a constructive heuristic solution method and an exact integer linear program were tried and compared using several test cases to trial the methods. As would be expected, the ILP generally finds the optimal solution but takes longer while the heuristic provides a feasible, non optimal solution much quicker. Although longer, the ILP did not take an unreasonable amount of time to find the optimal solution, usually less than ten minutes, so it was felt this would be the best method to use keeping the heuristic method as a backup in case the ILP was found to be too time consuming on some ill behaved cases. Although Moz and Pato discuss optimality they do not break it down and discuss the number of changes to the schedule directly.

Further research (Moz and Pato; 2004) developed the multicommodity flow formulations to more closely fit the rerostering context and taking in to account the option of overstaffing. The problem definition was unchanged from the earlier paper and the original disaggregated formulation was tested against a new aggregated formulation, where nodes corresponding to same shift tasks were aggregated into a single node. The aggregated method found optimal solutions for all instances within one minute and provided much better results than the disaggregated method from their previous paper which only found optimal solutions for 22 problems with a time limit of 5 hours for each problem. Even though the results obtained were good they felt that more needed to be done so that the soft constraints could be taken into account. This shows a significant improvement in the solution method but there is still no detailed discussion about how optimality translates to actual changes in the schedule.

Moz and Pato (2007) continued to try and improve the algorithm, looking at incorporating a genetic algorithm into their constructive heuristic which they had developed. Two versions were implemented and tested, the first one assigned tasks to nurses that had been ordered by some criteria, the second one randomly ordered the tasks and nurses in the initialisation step. The need to randomise the nurses as well as the tasks arose because more diversity was required when repeatedly running the algorithm. The quality of the solutions obtained was found to depend on the ordering of the tasks and the nurses so it was possible to search for better solutions by simply rerunning the heuristic using different orderings at the initialisation phase. This is basically a random search so a guided search was implemented by developing a genetic algorithm in the hope of improving the quality of the solution. The computational tests showed the genetic algorithm substantially improved the quality of the solution but was slower than the constructive heuristic. The computational time was still within acceptable limits though so, genetic algorithms were shown to be a good way forward for their research.

With the promising results from the genetic algorithm solution method, Moz and Pato (2007) introduced a second objective in to the fitness function. As well as trying to minimise the dissimilarity between the new and old schedules, they also wanted to minimise the amount of overtime which they describe as minimising the difference between the number of scheduled duties and the number of duties that should be performed during a particular period. A utopic strategy was introduced to try and induce diversity in to the population and compared the results with those achieved by using a basic bi-objective version without elitism and utopic strategy. The method with elitism and utopic strategy was shown to be better as smaller minima for the objectives were found. Moz and Pato comment on the fact the method is time consuming but the results show computational time was under 10 minutes in all cases. This is longer than other methods developed previously but still more than acceptable.

Bard and Purnomo (2005b) looked at a slightly different rescheduling problem thy call reactive scheduling where a 24 hour period is reviewed at a time, (25). They start with a 6 week base schedule and then consider each shift a couple of hours prior to its start and make amendments where necessary. Nurses calling in sick, changes in the number of nurses required to cover a shift, and other unforeseen events can make it necessary to make small changes to the schedule so that wards are correctly staffed. Many options are available such as using agency staff, cancelling nurses due to work, and voluntary or mandatory overtime. Bard and Purnomo's paper concentrates on efficiently reallocating available resources such as floaters, on-call nurses and agency nurses in a way that minimises the cost of the changes and the differences between the new and original schedules. This is felt to be the best way of dealing with changes because there is evidence to suggest that if a nurse works overtime, voluntary or mandatory, they may ring in sick on a later shift causing further problems. In the first instance an integer program solution was tried, but it showed that for some problems of just 40 nurses, convergence to within 1% of the Linear Program lower bound failed after several hours. The time frames available for this particular problem mean that was not good enough. As it was important to find a feasible solution quickly in case an optimal solution could not be found, a column generation approach was taken, forming a set covering problem for the planning horizon which was solved using a heuristic technique. A set of columns were generated using a branch and price algorithm to create a limited size column. The set covering problem was then solved using two different heuristic techniques, a tabu search and a restricted set covering integer program. The restricted set covering integer program was found to be far superior so, the tabu search was discarded. Bard and Purnomo propose a method for generating 'good' columns for each nurse defined on set criteria which helps to refine the way cuts are performed in the branching process, enabling more good solutions to be found quicker. It was also shown that adjusting penalties can have an affect on the final result as does altering the maximum number of penalties allowed for each nurse. Using tests based on operational data to create 5 complex situations, the method was found to perform very well solving problems with up to 200 nurses efficiently.

The little research available on nurse rescheduling appears to have focused on the methodology and algorithms to solve the problem rather than how the shifts, and changes to the shifts, affect the nurses. The research of Moz & Pato, while identifying the need for considerations about shift patterns and nurses' preferences, does not take them into account in their models. With nurse scheduling focusing so much on giving nurses greater control over their own schedules, and the results showing this can greatly improve a nurse's perception, there seems to be a gap when it comes to nurse rescheduling.

This article starts filling this gap, providing a method for quickly and simply finding schedules in the first instance that can cope with simple nurse preferences and then be used if any rescheduling is required. The rescheduling will aim to minimise the number of changes to the schedule and continue to consider the preferences introduced at the initial scheduling stage.

Two models for nurse scheduling

The following representative problem is used, drawing on Burke *et al*. (2004). A single ward has 20 permanent nursing staff available to schedule over a planning horizon of 28 days (4 weeks). There are no floating nurses or on call staff. Each nurse is assigned to one of 3 contracts: Full-time, Part-time, Day-only. All nurses are the same level and experience. There are 3 shifts (Early, Late, Night) which must all be covered every day, plus an empty ”Off” shift that identifies a 24 hour period used when a nurse is not assigned to a shift or part of a shift.

The nurse scheduling and rescheduling problems can both have many objectives to consider, potentially including preferences, costs, and in the case of the rescheduling problem disruption. There are several exact methods that can cope with more than one objective including goal programming, weighted sums, and preemptive optimisation. This paper uses a combination of weighted sums and goal programming as it is a convenient way of combining several objectives into one, using weights to define the order of priority.

Two different models were developed for comparison purposes. The first model considers individual shifts, referred to as the individual days model, assigning individual shifts to each of the nurses and giving more flexibility over the schedules produced. The second model uses predefined weekly patterns that form a set covering problem, making the model less flexible but simplifies the model and reduces solution times. This is referred to as the patterns model. Both models have been developed to contain the same constraints and objectives to allow direct comparison. The models started as a basic assignment of the correct number of nurses to each shift and were then developed to include more complicated constraints drawing on the research previously reviewed. We only present the final models:

Individual days model

The individual days model assigns nurses to individual shifts. To formulate the model, the following indices are used:

s: Shift = Early, Late, Night, Off
d: Day of the week = 1,...,7
t: Week = 1,...,4
n: Nurse = 1,...,N
c: Type of contract = FullTime, PartTime, DayOnly

where the input data required by the model are:

r_{sdt} Number of nurses required on shift s on day d in week t
ho_s Number of hours in shift s
Ho_c Maximum contracted hours for contract c
$c(n)$ Type of contract of nurse n
g_{sdt} Cost of a shortage of one nurse on shift s of day d in week t
h_{sdt} Cost of a surplus of one nurse on shift s of day d in week t

and the decisions variables are:

x_{nsdt} Assigns nurse n to shift s on day d in week t (= 1, otherwise = 0).
S^+_{sdt} Shortage of nurses on shift s of day d in week t, ≥ 0.
S^-_{sdt} Surplus of nurses on shift s of day d in week t, ≥ 0.

The individual days model is:

$$Minimise \sum_{s,d,t} (g_{sdt}\, S^+_{sdt} + h_{sdt}\, S^-_{sdt}) \tag{1}$$

such that

$$\sum_{n} x_{nsdt} + S^+_{sdt} + S^-_{sdt} = r_{sdt} \qquad \forall\ s,d,t \tag{2}$$

$$\sum_{s} x_{nsdt} = 1 \qquad \forall\ n,d,t \tag{3}$$

$$\sum_{s,d} x_{nsdt} \leq 5 \qquad \forall\ n,t \tag{4}$$

$$\sum_{d} x_{n,\mathrm{Night},d,t} \leq 2 \qquad \forall\ n,t \mid c(n) \neq \mathrm{DayOnly} \tag{5}$$

$$\sum_{d} x_{n,\mathrm{Night},d,t} = 0 \qquad \forall\ n,d,t \mid c(n) = \mathrm{DayOnly} \tag{6}$$

$$x_{n,\mathrm{Night},d,t} + x_{n,\mathrm{Early},d+1,t} + x_{n,\mathrm{Late},d+1,t} = 1 \qquad \forall\ n,d<7,t \tag{7}$$

$$x_{n,\mathrm{Night},7,t} + x_{n,\mathrm{Early},1,t+1} + x_{n,\mathrm{Late},1,t+1} = 1 \qquad \forall\ n,t<4 \tag{8}$$

$$x_{n,\mathrm{Late},d,t} + x_{n,\mathrm{Early},d+1,t} \leq 1 \qquad \forall\ \ n,d<7,t \tag{9}$$

$$x_{n,\mathrm{Late},7,t} + x_{n,\mathrm{Early},1,t+1} \leq 1 \qquad \forall\ \ n,t<4 \tag{10}$$

$$\sum_{s,d} ho_s\, x_{nsdt} \leq Ho_{c(n)} \qquad \forall\, n,t \tag{11}$$

Constraint (2) ensures that every shift has the correct number of nurses assigned to it, calculating any shortage or surplus of nurses. The objective function (1) minimises the costs of any shortage or surplus of nurses, the former typically being more important than the latter whose cost is thus much smaller. Constraints (3) and (4) oblige each nurse to work no more than shift per day and to have at least 2 days off every week. Constraints (5) and (6) prohibit non-Day-Only nurses from working more than 2 nights a week and ensure that Day-Only ones do not work nights. Constraints (7) and (8) ensure that if a nurse works a night shift then they cannot work an early or late the following day. Constraints (9) and (10) help ensure the quality of the schedule by preventing a nurse from being assigned an early shift if they worked a late shift the previous day. The number of hours a nurse can work in a week are restricted by constraint (11). Nurses of different skills and experience levels, the use of agency nurses, and scheduled absences, can be straightforwardly incorporated into the model. This paper uses a relatively simple model in these respects in order to focus on rescheduling.

Computational testing

The model was tested on PC with a 2GHz Pentium M processor and 1.5GB of ram. The modelling software was AMPL (Fourer *et al.*; 2003). For optimisation, two solvers were used: the commercial heavy-weight solver CPLEX v11.0 (Ilog; 2007), and the free-to-use MINTO solver (Linderoth and Ralphs; 2005) available over the internet on the NEOS Server (Fourer and Goux; 2001). A time limit of 10 minute was imposed for both solvers.

The model was tested on an instance representative of "normal" conditions, i.e., 2 or 3 nurses per shift are required. The solution was adequate as it assigned nurses to shifts minimising the overstaffing and understaffing of shifts. Legal constraints were also satisfied, ensuring that nurses did not work more than their contracted hours, had adequate time off and was flexible in scheduling nurses with different contracts. However, as was seen in the literature review, building nurses' preferences into the model can greatly impact on the perception of a schedule and therefore the happiness, morale and quality of a nurse's work.

To try and improve the quality of the schedules produced, the model was adapted to include some of the constraints indicated in the literature as generally preferred by nurses. These include ensuring that each nurse gets at least one complete weekend off in every 4 weeks, that a nurse works no more than 5 days in a row, and minimising the number of times a nurse has an isolated day off. Allowing the nurses one weekend off in four and not allowing nurses to work more than 5 consecutive days were introduced as constraints. To minimise the number of isolated days off, a new deficiency variable was created to be included in the objective function and calculated via additional constraints.

As a result, the model became complex with many more constraints, but still incomplete. The CPLEX solver took just 4 seconds to obtain a optimal zero-valued solution (i.e., there were no nursing shortages or surpluses) while the MINTO solver on the NEOS server only managed to provide a non-zero-valued solution at the end of the 10-minute time limit. In both solutions, the use of individual nurses was sometimes uneven, even though the constraints specified at the end of section 3.1 were satisfied. Although the model still lacked constraints to evenly spread the isolated days off among the nurses, no nurse had more than 1 isolated day off. Thus the schedule looked fair and as such would be better perceived by nurses. However, fairly distributing isolated days off really needs additional constraints, adding to the size and complexity of the model.

Under tighter conditions of requiring cover by an extra nurse per shift, CPLEX used all its 10-minute quota of computing time, but did provide a non-zero-valued solution within 3% of optimality, while MINTO was unable to even find a feasible solution within 10 minutes.

So long as the computational time remains reasonable and model complexity is not an issue for the user, we could continue adding constraints to improve the model and its perceived fairness, such as ensuring that night shifts and worked weekends are evenly distributed among nurses, that the day a night shift ends should be considered as a working day when calculating weekends off, working 5 consecutive days and isolated days off. One could minimise the distance from the schedule indicated on a sign-up sheet of each nurse's preferred shifts. An approach similar to that of the rescheduling discussed below could take into account individual preferences, improving the schedule's perceived fairness. Fortnightly shift schedules could be used, instead of weekly ones, so that a nurse may work more hours than contracted in one week, but balanced out with fewer hours in the following week. The sequence through which shifts are rotated and the gap between changes in types of shift could be considered to improve the quality of the schedule.

Talking to nurses directly would probably identify further fairness and preference constraints, eventually producing a cumbersome and inflexible model that would be difficult to maintain. Furthermore, it may not be possible to formulate certain constraints with the integer linear programming paradigm. The next subsection considers an alternative modelling formulation takes the complexity away from linear algebraic constraints and instead embeds it in the decision variables.

Patterns model

The patterns model selects from an input set of predefined shift patterns to identify shifts for each nurse. Only acceptable shifts patterns are allowed in the set which, however, may not be complete, i.e., it does not necessarily contain all acceptable shifts patterns. In other words, an optimal solution to the patterns model is not guaranteed to be a mathematically optimal schedule, but should be nearly optimal if we take care to include only good quality shifts patterns in the set. This approach is not original -many authors have used it on a variety of combinatorial problems, including manpower scheduling, often under the name of column generation (Barnhart *et al*.; 1998), classically applied to cutting-stock problems (Gilmore and Gomory; 1961).

Column generation can be used to include shift patterns that will eventually result in a mathematically optimal solution, but the concern of this paper is work with a relatively limited set of shift patterns that are attractive to the nurses in terms of their personal preferences. Column generation may be used to suggest good shift patterns whose inclusion may nevertheless be vetoed by the human scheduler taking into account preference and other criteria. To formulate the patterns model, an extra index is used for the shift patterns:

p: Pattern over a week of 7 days = $1,...,P$

The data needed is now:

r_{sdt} Number of nurses required on shift s on day d in week t
co_{sdp} = 1 if pattern p covers shift s on day d, otherwise 0

and the decisions variables are now:

x_{npt} Assigns nurse n to shift pattern p in week t (= 1, otherwise = 0).
S^{+}_{sdt} Shortage of nurses on shift s of day d in week t, ≥ 0.
S^{-}_{sdt} Surplus of nurses on shift s of day d in week t, ≥ 0.

The shift patterns model is:

$$Minimise \sum_{s,d,t} (g_{sdt}\, S^{+}_{sdt} + h_{sdt}\, S^{-}_{sdt}) \tag{12}$$

such that

$$\sum_{n,p} co_{sdp}\, x_{npt} + S^{+}_{sdt} + S^{-}_{sdt} = r_{sdt} \qquad \forall\ s,d,t \tag{13}$$

$$\sum_{p} x_{npt} = 1 \qquad \forall\ n,t \tag{14}$$

The objective function (12) is the same as for the individual days model and minimises any shortage and surplus of nurses. Constraint (13) is the equivalent of (2), ensuring that every shift has the correct number of nurses assigned to it, calculating any shortage or surplus of nurses. Constraint (14) makes sure that only one shift pattern per week is assigned to each nurse.

If the model were to (trivially) cover 2 only days, then it would be possible to use total enumeration to specify all 4 × 4 = 16 patterns. However, 2 of the patterns are (1) a night shift followed by an early shift, and (2) a night shift followed by a late shift. These combinations are not allowed so the two patterns are removed from the model leaving 14 acceptable patterns. Thus the exclusion of prohibited patterns avoids the need for constraints such as (3) to (7). Since many constraints of the individual days model are now embedded in the set of acceptable patterns, the patterns model is easier to understand, maintain and update when required. Careful selection and limiting of the patterns to be used by the model could lead to better solutions and quicker solution times. Before we can compare the performance of the two models, more constraints are required.

As the patterns are one week long, we still need week-linking constraints similar to (8). For example, if one pattern ends with a night shift, the pattern that follows it must not start with an early or a late shift. Contract types and associated constraints also need adding to the model.

The following new data is needed:

E: Patterns which have the weekend off
F: Patterns not suitable for day only staff
G: Patterns not suitable for part time staff
j_{pq} If pattern q can follow pattern p then = 1 , otherwise = 0

The following constraints are needed in addition to (13) and (14):

$$x_{npt} + x_{nq,t+1} \geq 1 + j_{pq} \qquad \forall\ n,p,q,t < 4 \tag{15}$$

$$\sum_{p \in E,\, t} x_{npt} \geq 1 \qquad \forall\ n \tag{16}$$

$$x_{npt} = 0 \qquad \forall\ n|c(n) = \text{DayOnly}, p \in F, t \tag{17}$$

$$x_{npt} = 0 \qquad \forall\ n|c(n) = \text{PartTime}, p \in G, t \tag{18}$$

Constraint (15) checks whether 2 patterns can be adjacent and ensures that no nurse works more than 5 consecutive days, that there are no isolated days off and checks whether the last shift of the first pattern can be preceded by the first shift of the second pattern. Each nurse gets at least one weekend off in 4 due to constraint (16). Constraints (17) and (18) make sure that prohibited patterns are not assigned to nurses on non full-time contracts (the variables affected simply being prefixed at zero before optimisation). For example (17) ensures day only staff do not get assigned patterns that have a night shift and (18) ensures part time staff only get assigned patterns that do not exceed their maximum contracted hours.

As with the individual days model, the night shift does not get recognised as being a shift on the day it finishes. This means the same problems with weekends off, isolated days off and working 5 consecutive days that occur in the individual days model occur in the patterns model. It is worth noting, however, that rectifying these problems in the patterns model is much more straightforward requiring only a few simple changes to the data. In the individual days model, the model itself would need changing and new constraints would need to be added.

Making available 26 pre-selected shift patterns p and using the same normal instance as in section 3.2, an optimal zero-valued solution is found in 0.23 seconds with CPLEX and 1.62 seconds with MINTO on the NEOS server. Examining the normal CPLEX solution, it appears that the limited nature of the set patterns available may be causing some nurses to be given whole weeks off while other nurses are working all 5 shifts in a week. It may be possible to overcome this by introducing a larger set of carefully selected or generated patterns. Assigning a penalty to each pattern would allow the patterns to be distributed more fairly among the nurses by minimising the maximum penalty of assigned patterns or keeping them within limits.

Under tighter conditions of requiring cover by an extra nurse per shift, CPLEX took 33 seconds to an optimal non-zero-valued solution while MINTO used all its 10-minute quota to find a solution within 17% of optimality. Table 1 shows the results obtained when scaling up the instance size under tight conditions. Observe that a quality solver such as CPLEX can give near-optimal solutions for large realistically-sized instances, while MINTO on the NEOS server clearly struggles and would require alternative solution methods or decomposition into smaller instances.

Table 1. Results from Patterns Model scaling up under tight conditions.

Instance		*CPLEX*		*NEOS*	
Weeks	*Nurses*	*CPU Time*	*Gap*	*CPU Time*	*Gap*
4	30	174 seconds	Optimal	10 minutes	7.6%
6	20	229 seconds	Optimal	10 minutes	6.3%
8	20	10 minutes	0.01%	10 minutes	15%
4	40	10 minutes	3%	10 minutes	Infeasible

Comparison of the individual days and patterns models

For a given problem instance, it appears from initial testing that the patterns model takes much less time to solve than the individual days model. The individual days model represents in theory all feasible schedules. It does so at the expense of a large set of complex constraints that may require an expert modelling person to modify if the staffing environment of rules change. The individual days model can also take a long time to solve to optimality, or not at all, depending on the solver used and the actual instance.

In contrast, the shift patterns model is much simpler as most constraints are satisfied via prefiltered shift patterns. The set of patterns would be very large if it were complete. However, a near-optimal solution may very well be possible to obtain from a reduced set constituting preferred schedules. The risk is that a shift pattern that is part of an optimal solution may be missing. If the staffing constraints change, then a model end-user can modify the set of permissable/preferred shift patterns without the need for an expert modelling person. This user input and autonomy, along with the simpler set of algebraic constraints, is likely to create trust and confidence in the model output.

Models for nurse rescheduling

When considering rescheduling and the disruption it causes, two alternative approaches can be used. The first approach simply minimises the number of changes, reasoning that fewer changes cause less disruption. The second approach recognises that different types of change cause different amounts of disruption; for example, being assigned a night shift on a day that was originally free would be more disruptive than a simple change from a late shift to an early shift on the same day. For this approach, penalties can be assigned to each kind of change. The model's objective function then includes the minimisation the sum of these penalties.

Nurses' preferences are better modelled by the second approach as it takes into account the impact of any changes on the individual nurses. As well as minimising the impact of the disruptions caused by the rescheduling, the nurses' preferences that were supported in the initial scheduling model should continue to be satisfied.

Individual days model rescheduling

For the individual days rescheduling model, human judgement was used to assign penalties to each change in an individual shift. For example, it was judged that working a shift on a previously assigned day off was the worst kind of change, while being given a day off that previously had to be worked would be the least disruptive kind of change. The following new data were introduced to represent the new schedule:

d_{nsr} Nurse n's own disruption penalty of a change from shift s to shift r.
w_n Weighting for nurse n's disruption penalties relative to the 1.0 weighting attached to the nurse shortage term $\sum_{s,d,t} S^+_{sdt}$
a_{nsdt} If nurse n is assigned to shift s on day d of week t in the original schedule then = 1, otherwise = 0
ch_{dnt} If the change to day d of week t was requested (and previously agreed) by nurse n then = 1, otherwise = 0

The rescheduling objective function minimises a weighted sum of the original function (1) and the disruption penalties:

$$Minimise \sum_{s,d,t} (g_{sdt}\, S^+_{sdt} + h_{sdt}\, S^-_{sdt}) + \sum_{n,d,t,s,r} w_n d_{nsr} a_{nsdt}(1 - ch_{dnt}) x_{nrdt} \qquad (19)$$

Observe in objective function (19) that the disruption penalty $w_n d_{nsr}$ can be a product of a nurse's own subjective assessments d_{nsr} factored up or down by management's own evaluation w_n of the nurse's assessments relative to the importance of nursing cover shortages. Furthermore, this disruption penalty is not just multiplied by the original schedule's binary assignment parameter a_{nsdt} and the binary rescheduling variable x_{nrdt}, but also by the parameter ch_{dnt} that records whether the nurse requested the change. Thus previously-agreed changes requested by a nurse (and so pre-fixed in the model) are not included in the disruption calculations. The constraints remain the same as in the initial scheduling model so that rescheduling adheres to the same restrictions. Thus any preferences included in the original model are not suddenly disregarded.

Patterns model rescheduling

Rescheduling with the patterns model makes changes to a week's shift pattern. Thus numerous shift changes can occur because one pattern has been swapped for another. Rather than simply adding the penalties associated with each individual shift change caused by the change in pattern, the change in the pattern was considered as a whole. For example, if the newly assigned pattern has more days off than the previously assigned pattern, it could be argued that this was a better pattern even if there were several individual shift changes. Similarly to the individual days rescheduling model, the following new data were introduced to represent the new schedule:

d_{npq} Nurse n's own disruption penalty of a change from pattern p to pattern q.
w_n Weighting for nurse n's disruption penalties relative to the 1.0 weighting attached to the nurse shortage term $\sum_{s,d,t} S^+_{sdt}$
a_{npt} If nurse n is assigned to pattern p in week t in the original schedule then = 1, otherwise = 0
ch_{nt} If the change to week t was requested (and previously agreed) by nurse n then = 1, otherwise = 0

The rescheduling objective function minimises a weighted sum of the original function (1) and the disruption penalties:

$$Minimise \sum_{s,d,t} (g_{sdt}\, S^+_{sdt} + h_{sdt}\, S^-_{sdt}) + \sum_{n,t,p,q} w_n d_{npq} a_{npt}(1 - ch_{nt}) x_{nqt} \qquad (20)$$

Again, the constraints remain unchanged from the initial scheduling patterns model so that the reschedule adheres to the same restrictions.

Fairness in the rescheduling models

Given small perturbations to the instance of section 3 under normal conditions, both rescheduling models produce optimal results in well less than a second using the CPLEX solver. The two models impact in different ways on the perceived fairness of shift rescheduling. The individual days model has greater flexibility with the solutions indicating that it was able to assign smaller shift changes to a larger group of nurses whereas, the patterns model assigned one big change to a single nurse. The set of patterns needs investigating in more depth so that a better understanding can be gained about how the selected patterns affect the solution and whether a more informed choice of patterns can help reduce some of the problems.

The number of changes required to accommodate the requested changes are relatively small, but there are no constraints or goals to try and spread these evenly amongst the nurses. This means that while some nurses get no changes, others are required to change a number of shifts. This could potentially cause problems to do with fairness and perception of the nurses who have more changes. For example, the rescheduling in the patterns model resulted in a simple swap of patterns between two nurses. Nurse 12 that previously had week 4 off is now working two shifts, one of which is a night shift, and nurse 12 is the only person to have any changes at all. Nurse 12 would probably see the changes as very unfair, leading to ill feeling and negating all the positive work done to try and improve the schedules. Such problems could be reduced by (i) negotiating a future week off for nurse 12 and fixing this in a future run of the model, (ii) providing a (maybe larger) set of more informed pattern choices, (iii) including explicit fairness constraints in the model, or (iv) using judgement to balance the human fix of (i) with potential over-modelling from (ii) and (iii).

Urgent mid-week rescheduling

Following surprise changes to nurse availability or other data upon which a schedule is based, rescheduling frequently needs to be carried out with immediate effect. Consider the patterns model. Suppose we are halfway through week 1 of the schedule. Let W^1 be the set of shift-day pairs (s, d) remaining in week 1. Let P^1 be the set of shift patterns that match what has occurred in week 1 so far and can occur until the end of the week (i.e., taking the disturbance into account). Then the patterns rescheduling model becomes:

$$Minimise \sum_{s,d,t} (g_{sdt}\, S^+_{sdt} + h_{sdt}\, S^-_{sdt}) + \sum_{n} \sum_{p,q \in P^1} w_n d_{npq} a_{np1}(1 - ch_{n1}) x_{nq1} + \sum_{n} \sum_{t \geq 2} \sum_{p,q} w_n d_{npq} a_{npt}(1 - ch_{nt}) x_{nqt} \quad (21)$$

such that

$$\sum_{n,p} co_{sdp}\, x_{np1} + S^+_{sd1} + S^-_{sd1} = r_{sd1} \qquad \forall\ (s,d) \in W^1 \quad (22)$$

$$\sum_{n,p} co_{sdp}\, x_{npt} + S^+_{sdt} + S^-_{sdt} = r_{sdt} \qquad \forall\ s,d,t \geq 2 \quad (23)$$

$$\sum_{p \in P^1} x_{np1} = 1 \qquad \forall\ n \quad (24)$$

$$\sum_{p} x_{npt} = 1 \qquad \forall\ n,t \geq 2 \quad (25)$$

plus adaptations of the week-linking constraints (15) and (16) and others.

Thus the patterns model is readily adapted to urgent rescheduling.

Conclusions and further research

Two models have been produced that support nurse scheduling and adapted for rescheduling while considering and preserving nurses' preferences. Many ideas from the literature were used to try and improve the quality of the schedules as perceived by the nurses themselves, rather than considering monetary costs. The patterns model in particular provided good solutions in reasonable computing time, although each has its own advantages. A good baseline has been set so that work could now be carried out with a hospital to allow informed decisions to be made that will enable the models to develop further.

Directions for further research and improvements include the following. If there is frequent rescheduling then maybe penalties should be based on how far into the future a change is. For example, with a 4-week planning horizon, a change at the end of the 4th week may be less problematic than the same change occurring within the next few days. In fact, detailed scheduling of later weeks could be a waste of effort if this part of the schedule is never actually implemented. It would be better to use the model on a rolling horizon with detailed scheduling over a stable short-term, and approximate mid-to-long-term scheduling using a smaller pattern set to enable planning. In other words, there is scope to develop an effective and efficient joint planning-&-scheduling model, as in manufacturing production (Clark; 2003, 2005).

Individual preferences could be considered via a sign-up sheet that could be used as the "original" schedule and the reschedule model run to complete the schedule but minimising the changes from the sign up sheet. Rather than nurses saying which individual shifts they want to work, they could state which pattern they want to work in a particular week. Such preferences could be incorporated in the automatic generation of patterns every time the model is run, or at set intervals, to produce a set of patterns that provide the desired level of flexibility. The set produced needs to be limited to keep computational times down for the scheduling model.

Finally, noting that the rescheduling takes the original schedule into account, but not vice-versa, future research could develop a model that considers the dependency between the two stages' decisions. This is a much more complex challenge involving robust stochastic modelling, but one that would certainly improve the resulting schedules.

Acknowledgments— The authors wish to thank the reviewers whose valuable feedback resulted in a better paper.

References

Aickelin, U. and Dowsland, K. A. (2004). An indirect genetic algorithm for a nurse-scheduling problem, Computers and Operations Research 31(5): 761–778.

Azaiez, M. N. and Sharif, S. S. A. (2005). A 0-1 goal programming model for nurse scheduling, Computers and Operations Research 32(3): 491–507.

Bailyn L, Collins R and Song Y (2007). Self-scheduling for hospital nurses: an attempt and its difficulties, Journal of Nursing Management 15(1): 72–77.

Bard, J. F. and Purnomo, H. W. (2005a). A column generation -based approach to solve the preference scheduling problem for nurses with down-grading, Socio-Economic Planning Sciences 39(3): 193–213.

Bard, J. F. and Purnomo, H. W. (2005b). Hospital-wide reactive scheduling of nurses with preference consideration, IIE Transactions 37: 589–608.

Bard, J. F. and Purnomo, H. W. (2005c). Preference scheduling for nurses using column generation, European Journal of Operational Research 127(2): 510–534.

Bard, J. F. and Purnomo, H. W. (2007). Cyclic preference scheduling of nurses using a lagrangian-based heuristic, Journal of Scheduling 10(1): 5–23.

Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W. P. and Vance, P. H. (1998). Branch-and-price: Column generation for solving huge integer programs, Opera¬tions Research 46(3): 316–329.

Brucker, P., Qu, R., Burke, E. and Post, G. (2005). A decomposition, construction and post-processing approach for a specific nurse rostering problem, Proc of The 2nd Multidisciplinary Conference on Scheduling: Theory and Applications, NY, USA, pp. 397–406.

Burke, E. K., Causmaecker, P. D., Berghe, G. V. and Landeghem, H. V. (2004). The state of the art of nurse rostering, Journal of Scheduling 7(6): 441–499.

Clark, A. R. (2003). Optimization approximations for capacity constrained material requirements planning, International Journal of Production Economics 84(2): 115–131.

Clark, A. R. (2005). Rolling horizon heuristics for production and setup planning with backlogs and error-prone demand forecasts, Production Planning and Control 16(1): 81– 97.

De Grano, M. L., Medeiros, D. and Eitel, D. (2009). Accommodating individual preferences in nurse scheduling via auctions and optimization, Health Care Management Science 12: 228–242.

Dias, T. M., Ferber, D., de Souza, C. and Moura, A. V. (2003). Constructing nurse schedules at large hospitals, International Transactions in Operational Research 10(3): 245–265.

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

Fourer R, Gay DM and Kernighan BW (2003). AMPL - A Modeling Language for Mathematical Programming, second edn, Duxbury Press / Brooks-Cole Publishing Company, USA. http://www.ampl.com/.

Fourer, R. and Goux, J.-P. (2001). Optimization as an internet resource, Interfaces 31(2): 130–150.

Gilmore, P. C. and Gomory, R. E. (1961). A linear programming approach to the cutting-stock problem, Operations Research 9(6): 849–859.

Ilog (2007). CPLEX 11.0 User's Guide, ILOG S.A.

Kostreva, M., McNelis, E. and Clemens, E. (2002). Using a circadian rhythms model to evaluate shift schedules, Ergonomics 45(11): 739–763.

Linderoth, J. T. and Ralphs, T. K. (2005). Noncommercial software for mixed-integer linear programming, in Karloff JK (ed.), Integer Programming: Theory and Practice, Vol. 3 of Operations Research Series, CRC Press, London UK and Boca Raton FL, pp. 253–303.

McVicar, A. (2003). Workplace stress in nursing: a literature review, Journal of Advanced Nursing 44: 633–642.

Moz, M. and Pato, M. V. (2003). An integer multicommodity flow model applied to the rerostering of nurse schedules, Annals of Operations Research 199: 285–301.

Moz, M. and Pato, M. V. (2004). Solving the problem of rerostering nurse schedules with hard constraints: New multicommodity flow models, Annals of Operations Research 128(1-4): 179–197.

Moz, M. and Pato, M. V. (2007). A genetic algorithm approach to a nurse rerostering problem, Computers and Operations Research 34(3): 667–691.

Pryce, J., Albertsen, K. and Nielsen, K. (2006). Evaluation of an open-rota system in a danish psychiatric hospital: a mechanism for improving job satisfaction and work-life balance, Journal of Nursing Management 14: 282–288.

Rogers, A. E., Hwang, W.-T., Scott, L. D., Aiken, L. H. and Dinges, D. F. (2004). The working hours of hospital staff nurses and patient safety, Health Affairs 23: 202–212.

Royal College of Nursing (2004). NHS needs 25,000 more nurses http://news.bbc.co.uk/1/low/health/3466461.stm

Warner DM (1976). Scheduling nursing personnel according to nursing preference: A mathematical programming approach, Operations Research 24: 842–856.

Wright, P. D., Bretthauer, K. M. and Cˆe, M. J. (2006). Re-examining the nurse scheduling problem: Stang ratios and nursing shortages, Decision Sciences 37(1): 39–70.