Ann Oper Res (2007) 155: 279–288
DOI 10.1007/s10479-007-0202-4

Solving the multi-objective nurse scheduling problem with a weighted cost function

D. Parr · J.M. Thompson

Published online: 20 July 2007

Abstract The primary objective of the nurse scheduling problem is to ensure there are sufficient nurses on each shift. There are also a number of secondary objectives designed to make the schedule more pleasant. Neighbourhood search implementations use a weighted cost function with the weights dependent on the importance of each objective. Setting the weights on binding constraints so they are satisfied but still allow the search to find good solutions is difficult. This paper compares two methods for overcoming this problem, SAWing and Noising with simulated annealing and demonstrates that Noising produces better schedules.

Keywords Nurse scheduling · Meta-heuristic · Simulated annealing · SAWing · Noising

1 Introduction

The task of producing nurse work rosters is becoming increasingly important in the UK. Many hospitals are suffering from recruitment difficulties meaning that they need to use their available staff as efficiently as possible. Some hospitals are also attempting to recruit staff by promising family friendly work schedules. However this can have a knock-on effect on other staff and to maintain morale it is essential that all staff are given work schedules that give them the opportunity for a healthy social life. As the task of producing the work roster increases in difficulty, senior nurses are moving from traditional manual scheduling methods and looking for alternatives. Some hospitals have turned to self-rostering, where the nurses on each ward produce their own schedule but this can cause difficulties between staff members and still requires a senior nurse to oversee the whole process. An alternative is to switch to automated scheduling. This paper investigates the effectiveness of three meta-heuristic techniques based on local search in producing suitable schedules.

Our choice of techniques is motivated by two characteristics of the nurse scheduling problem. Firstly, the problem of finding a feasible solution may be a difficult problem in its

D. Parr · J.M. Thompson (✉)
School of Mathematics, Cardiff University, Senghennydd Road, P.O. Box 926, Cardiff, CF24 4YH, UK
e-mail: ThompsonJM1@cardiff.ac.uk

own right. Even if a schedule satisfying all the binding constraints can be found relatively easily it is often not possible to define a simple neighbourhood structure that will allow the space of feasible solutions to be searched effectively. Thus it is common to include infeasible solutions in the search space and to penalise them by an appropriate term in the objective function. Secondly the problem usually contains a number of soft constraints and objectives. The relative importance of these may vary from hospital to hospital, or even between different units in the same hospital, and the usual way of dealing with this is via a series of suitably weighted terms in the objective. However, as pointed out by Abramson (1991) and by Thompson and Dowsland (1996) it can be difficult to set the weights for the binding constraints at an appropriate level to allow sufficient flexibility to seek out good solutions whilst at the same time ensuring that a feasible solution is obtained. Similarly Wright (1996) points out that the weighting given to hierarchies of constraints/objectives in the evaluation function should not only reflect their coefficients in the objective, but also the relative difficulty of satisfying them.

For this reason we have selected two lesser used neighbourhood search approaches that alter the weights dynamically during the search. The first, the SAWing method was originally suggested as an intelligent means of escaping local optima by adjusting the weights to give priority to those terms that were proving more difficult to satisfy. The second, the noising method, was suggested as a simple way of mimicking the main features of simulated annealing. Our reason for choosing the noising method is that it achieves this by randomly adjusting the weights in the evaluation function. Both methods are evaluated on eleven data sets obtained from a number of different Welsh hospitals, and the results are compared with random descent and a simulated annealing implementation based on the same solution space and neighbourhood structure.

The remainder of the paper is organised as follows. The first section discusses the results of a survey that was conducted in order to ascertain an exact problem definition, and the extent to which nurse scheduling is a problem in the South Wales area. The following section describes previous solution methods and section three establishes the local search framework that will be the basis for our subsequent experiments. Section four describes the solution algorithms proposed here and the following section gives the results of our experiments. Section six draws conclusions and makes some general observations.

2 Nurse scheduling survey

In order to discover more about the nature of the nurse scheduling problem, questionnaires were sent to 36 hospitals in the South Wales area of the UK. These varied from large general hospitals to specialist units. Replies were received from 19 different hospitals and 31 distinct wards. The main points of interest in the replies were as follows:

- Average ward size is about 28 nurses, though numbers varied between 6 (a day hospital) and 95 (Intensive Care Unit).
- The schedule was produced manually in all cases. Nearly all replies indicated that a computerised solution package would be desirable. In one case, self-rostering was used.
- The average time taken to produce a monthly schedule is about 4 hours though in some cases, an entire days work is required. As a senior nurse nearly always completes this task, this is time that could be more usefully spent in actual patient care.
- The exact nature of the scheduling problem varied somewhat between wards; this is discussed below.

- Every reply indicated that nurses' requests are very important and should normally be satisfied. Overall, over 90% of nurses requests are satisfied but the problem of satisfying requests while meeting the needs of the service was highlighted in several replies.
- 25 (80.6%) of the respondents admitted that recruitment was an issue for their ward.
- Other problems frequently highlighted by replies included sickness (mentioned by 51.6% of respondents), achieving the correct skill mix (38.7%) and covering for holidays and study days (25.8%).

Almost all of the respondents expressed an interest in this research and several sent datasets for our experiments.

From the replies we were able to produce a general problem definition that includes all the requirements mentioned by different hospitals. Although the majority of responses came from wards requiring 24-hour cover, some day hospitals only needed daytime cover. In either case, the day is divided into blocks of time (shifts) and different nurses work different numbers of shifts according to their individual contracts. The most common divisions of the day are into earlies, lates (both daytime shifts) and night shifts, with the night shifts being longer than the day shifts. A full time nurse will normally work 5 day shifts or 4 night shifts in a week and part time nurses work a different number of shifts according to their individual contracts. However a substantial number of hospitals also allow nurses to work long day shifts which cover both an early and a late shift on the same day. A full time nurse working long shifts works three days per week.

Nurses are graded to reflect their skill and experience. Each ward states their requirements in terms of the number of nurses of each grade required for each shift. In most cases, respondents to the survey state that they have 3 levels of grade although some only consider 2. Generally, a higher grade nurse can do the job of a lower level nurse, but for most wards, the main requirement is for excess senior nurses to be allocated equally between shifts.

Cover requirements are universally considered to be binding, but there are a number of ways in which the schedule can be made more pleasant for staff. These include:

- Requests—see previous section.
- In most wards, nurses are not allowed to work days and nights in the same week.
- It is preferable to schedule days off together.
- Nights and weekends should be rotated.
- Study days should be allocated where possible.
- Nurses should not work more than 6 days or 4 nights in a row.
- For nurses working only long shifts, there should be a day off between such shifts.
- Nurses should not have 2 late shifts in a row.

3 Previous approaches

The problem of producing schedules of work can be traced back as far as the mid 1960's (for example Frances 1966 and Howell 1966). Baker (1974, 1977) considered cyclic schedules but the rigidity of such rosters has lead to more recent literature concentrating on more flexible methods. Nurse scheduling can be considered as a constraint satisfaction problem where weights are associated with each constraint. The objective is then to minimise the sum of the weights using iterative techniques (for examples, see Cheng et al. 1996; Berrada et al. 1996; Abdennadher and Schlenker 1999).

Local search methods have become increasingly popular means for solving the nurse scheduling problem. In particular tabu search (Glover 1989) has been used by several researchers including Dowsland (1998) whose search oscillates between phases that find a feasible covering and improve the quality of the schedule. Bellanti et al. (2004) use tabu search to create a partial solution and complete it using a greedy procedure which avoids infeasible solutions. Valouxis and Housos (2000) solve a relaxed integer linear programming formulation, and improve it using tabu search and Burke et al. (1998) investigate various hybrid tabu search methods. Aickelin and Dowsland (2004) use three different decoders in a genetic algorithm implementation, with each decoder representing a different balance between the need for feasibility and the desire for optimality. Burke et al. (2003) use variable neighbourhood search where the search switches to a different neighbourhood once it appears to have become trapped in a local optimum.

It is also common for the problem to be divided into phases, allowing each to be solved using different, appropriate techniques. Abdennadher and Schlenker (1999) suggest three phases with the first allocating the days off, the second distributing the night shifts and the final phase allocating earlies and lates. Dowsland and Thompson (2000) also propose three phases. Phase one determines whether there are sufficient nurses available to meet the covering requirements using an exact knapsack model. Phase 2 allocates nurses to day and night shifts using tabu search and the final phase determines earlies and lates, using an exact network representation. For a thorough review of applications to nurse rostering, see Burke et al. (2004).

Although these approaches are quoted as being successful on the problems for which they are described there is currently no definitive best approach. As our survey showed that the importance of constraints differ between hospitals the weights placed on them will differ also and this may have a significant effect on solution quality. The following section describes the underlying local search framework that will be used as the basis of all subsequent methods.

4 Local search framework

We have divided the problem into three phases, where the first determines which nurses work days and which work nights, the second determines the days and nights of the week each nurse works and the final phase allocates day nurses to earlies, lates and long shifts. Here we will concentrate on the second phase. The first is solved using a greedy search with backtracking to determine the set of nurses who should work nights so that cover requirements are satisfied and night shifts are rotated. This is because our survey indicated that rotation of night shifts was a binding constraint. The requirement for nurses not to have two late shifts in a row makes the network flow model used by Dowsland and Thompson (2000) unsuitable, but the methods proposed here for Phase 2 can equally be applied to the final phase.

In the second phase, the solution space is defined as the set of solutions for which each nurse has been allocated to the correct number of shifts. No other constraints are considered in the definition of the solution space. The objectives and constraints to be dealt with here are listed in Table 1, together with their weights that were determined by negotiation with hospital schedulers.

Constraints 1 are only considered for nurses who only work long shifts. Constraints 1 and 2 should always be satisfied, 3, 4 and 5 are hard constraints with a slight amount of flexibility and the others can be considered to be soft constraints.

For all except constraint group 2, the cost function is calculated as $\sum w_i x_i$ where w_i is the weight associated with constraint i and x_i is a measure of the number of times that

Table 1 Constraints and weights in the cost function

Constraint group	Description	Weight
1	Day off after long shift	1000
2	Cover constraints	500
3	Requests	320
4	Max of 6 consecutive days / 4 nights	100
5	Spread nurses across shifts	75
6	Rotate weekends	25
7	Days off together	5

constraint i is broken. For constraints 1, 3, 4 and 7, x_i is incremented by 1 each time such a constraint is broken. x_2 is calculated as the shortfall between the required number and the allocated number of nurses, summed over all grades and shifts. x_5 is incremented by 1 whenever the difference between the number of nurses of each grade allocated to any pair of shifts exceeds 1. Initial experiments showed that constraint 6 was satisfied if each nurse was given one weekend off in 3, so x_3 was increased by 1 for any nurse for whom this was not achieved.

Several starting solutions were evaluated, namely a random start where each nurse is allocated to random days or nights of the week, a greedy start that allocates nurses to shifts that are still undercovered so as to maximise the reduction in the cover constraints over all relevant grades and a consecutive start that allocates all nurses at the start of the week.

Two neighbourhoods were also evaluated. The initial neighbourhood (Move) simply altered the shift pattern of a single nurse by one day i.e. a day off and a day worked were swapped. A Swap neighbourhood also allows two nurses to swap days worked, thus not affecting the cover constraints (unless their grades differ) but potentially affecting the other constraints and objectives.

5 Search strategies

As stated previously determining the weighting for constraints purely according to their importance may lead to a solution space which is difficult to search. Our search strategies were selected as they adjust the weights, in the first case intelligently and in the second randomly. These are considered in turn.

5.1 SAWing

Eiben and van Hemert (1999) proposed Stepwise Adaptation of Weights (SAWing) as a means of intelligently modifying the weights on constraints. They suggest increasing the weights of those constraints that are still violated after a certain number of iterations, thus increasing the probability of them being satisfied. This can be considered to be a form of diversification, which re-shapes the solution space, allowing a search algorithm to escape from local optima. We propose four SAWing variants. The first three treat the groups of constraints as single entities whereas the fourth applies different weights to each individual constraint. In each case the starting weight w_i is as set by the nursing staff.

SAW1: Weights are increased by Δw_i if they are not satisfied after L iterations.

SAW2: A potential problem with SAW1 is that less important constraints may be given higher weights than more important ones. To avoid this, a hierarchical ladder of constraint groups is produced and the weight on each group is constrained to not exceed the weight of the next most important constraint.

SAW3: For a similar reason to SAW2, weights are constrained to remain within $x\%$ of their original values.

SAW4: A drawback with the above methods is that they update the weights on a constraint group regardless of how many constraints in that group are broken. For example it may be difficult to give one particular nurse a weekend off, but trivial for all other nurses. In this case, it does not seem fitting to increase the weight on the whole of group 6. Rather, a weight Ψ_{ij} is assigned to each constraint/nurse pair and these weights are only increased for the particular nurse for whom the constraint cannot be satisfied. These weights apply for constraints number 1, 3, 4, 6 and 7 and for other constraints, a weight Ω_{ik} is assigned to each constraint/day pair, as these constraints can only be assessed for each day rather than for individual nurses.

5.2 Noising

The noising method (Charon and Hurdy 1993, 2001) involves varying the cost function during a search by adding random error (noise) to the data. The noise is added so that $w_i' = w_i + (rnd * R* \max(w_i))$ for a constraint i, where w_i is the original weight, w_i' is the noised weight, rnd is a random number drawn uniformly between -1 and $+1$, R is a parameter (perturbation rate) and $\max(w_i)$ is the greatest absolute weight of constraint i. The parameter R decreases during the search between two limiting values, $R_{\max}$ and $R_{\min}$ so that the search will converge to a local optimum. The algorithm works as follows:

The noising method
Produce random solution S. Cost = $f(S)$, f is as defined earlier.
Set best solution $S_{\text{best}} = S$.
$Step = (R_{\max} - R_{\min}) / \ number\_of\_cycles$
$Rate = R_{\max}$
For $cycle = 1$ to $number\_of\_cycles$
Produce noised data using $R = Rate$
Perform descent using the noised data, starting from S. Resulting local optimum is S'.
Perform descent using the actual data, starting from S'. Resulting local optimum is S.
If $f(S) < f(S_{\text{best}})$ then set $S_{\text{best}} = S$.
$Rate = Rate - step$
Next $cycle$

In order to judge the relative performance of the SAWing and noising methods, a simulated annealing algorithm was also applied to the test problems. Simulated annealing accepts worsening moves with probability $e^{-\delta/t}$ where δ is the change in the cost function and t is a parameter known as the temperature (see Kirkpatrick et al. 1983 and Dowsland 1993 for detailed discussions). Simulated annealing was chosen due to its similarities with the noising method.

In a similar fashion to SAWing, noising can be applied to either each constraint group or each individual constraint. It should be noted that although noising involves several descents, most should be relatively quick as the solution space landscape is being tweaked rather than being totally changed and each search begins from the previous local optimum. The next section describes the results of our experiments.

Table 2 Data characteristics

Dataset	No. nurses	Long shifts?	No. grades	No. weeks
1	95	Yes	3	1
2	23	No	2	2
3	25	No	3	4
4	23	Yes	3	4

Table 3 Mean descent and simulated annealing results

Dataset	1	2	2	3	3	3	3	4	4	4	4
Week	1	1	2	1	2	3	4	1	2	3	4
Descent	180	55	45	105	180	135	140	2505	2500	3050	3050
SA (1)	90	50	25	100	178	128	120	1000	500	1380	575
SA (2)	62	50	25	100	169	117	107	1005	500	625	525

SA (1) is simulated annealing with a geometric cooling schedule

SA (2) is simulated annealing with a Lundy and Mees cooling schedule

6 Results

Data sets were obtained from four local hospitals; these are available from the authors. Their characteristics are listed in Table 2.

Thus 11 weeks of data were used in total. In all experiments with data running over several weeks, the previous weeks schedule is fixed so in each case, each method is solving the same problem. In all cases descents are terminated after 2000 iterations without improvement and computational time is limited to 100 seconds on a Pentium 4 to enable fair comparison between methods. 100 seconds is chosen as many of the respondents to the survey indicated that ward computers are relatively slow yet nurses would expect a schedule to be produced relatively quickly. 10 random runs are performed in each case.

The first experiments were designed to test the initial algorithm implementation and to produce some results using the simulated annealing method. As expected, the move/swap neighbourhood was far superior to the move neighbourhood alone, and the starting solution did not have a significant effect on solution quality. To optimise the simulated annealing implementation, various cooling schedules were evaluated including geometric cooling i.e. $t_{n+1} = \sigma t_n$ where σ is a parameter < 1 and Lundy and Mees (1986) i.e. $t_{n+1} = t_n/(1 + \beta t_n)$ where β is a parameter. In both cases, starting temperatures between 5 and 100 were compared and all runs were terminated when the temperature reached a value of 0.1. In our experiments, σ and β were varied between 0.8 and 0.99, and between 0.0001 and 0.002 respectively. The best values found were $\sigma = 0.99$ and $\beta = 0.0002$. Table 3 shows the best results from both cooling schedules and shows that the Lundy and Mees cooling schedule performs better on average. The results are robust with respect to different values of β, with good solutions being achieved for values of β between 0.0001 and 0.001.

The next set of experiments compared the four SAW variants. For each, various parameters had to be determined and for reasons of brevity, only results for the best parameters are given here. For the methods treating the groups of constraints as single entities, values of Δw_i between 1 and 100 and L between 5 and 1000 were tested. The best results were obtained using $\Delta w_i = 5$ and $L = 500$. For SAW4, a shorter update period i.e. $L = 50$ was

Table 4 Comparison of SAWing techniques

Dataset	1	2	2	3	3	3	3	4	4	4	4
Week	1	1	2	1	2	3	4	1	2	3	4
SAW1	115	55	40	100	170	140	115	1050	1250	1425	620
SAW2	110	56	45	105	120	143	100	1070	775	1025	1070
SAW3	120	51	55	100	172	167	115	1030	1050	1220	1000
SAW4	110	50	25	100	165	158	112	1000	500	1050	725

Table 5 Mean results for the noising method

Dataset	1	2	2	3	3	3	3	4	4	4	4
Week	1	1	2	1	2	3	4	1	2	3	4
Noising	80	50	25	100	145	105	100	5	125	25	25

required and for SAW3, $x\% = 10\%$ was chosen. Table 4 shows the mean results for each dataset using these parameters. These parameters were consistently better across all datasets and changes in the parameters produced dramatic decreases in solution quality.

In general the results are not as good as those for simulated annealing. Of the SAWing methods, SAW2 and SAW4 gave the best results. An analysis of the final weights shows that for most datasets, the weights for the more important constraints never change as these were satisfied anyway, but the weights for soft constraints 6 and 7 in particular varied considerably between nurses indicating that the SAWing technique had succeeded in differentiating between constraints and nurses for whom it is difficult to satisfy the constraints, and those for whom it is relatively straightforward. In cases where the important constraints were not immediately satisfied, increasing their weight did not often lead to an improvement.

The final experiments evaluated the noising method and it was found that adding noise to constraint groups produced poor quality solutions. Thus the results in Table 5 are for adding noise to each nurse/constraint pair and each day/constraint pair. Literature quotes successful parameter values for $R_{\max}$ of 0.9 and $R_{\min}$ of 0.4 and our experiments also found these to be suitable parameters. The best results were obtained using parameters of $number\_of\_cycles = 40$. The results were robust with respect to the parameter levels with schedules of similar quality being produced for $R_{\max}$ between 0.7 and 0.9 and $R_{\min}$ between 0.2 and 0.4.

It can be seen that the noising method matches or outperforms simulated annealing in all but one case and although it was found that simulated annealing results could be improved using slower cooling, they required run times in excess of the noising method to produce results of comparable quality. Noising is particularly successful in solving problems for which it appears difficult to obtain feasible solutions. For example simulated annealing and SAWing struggled to obtain schedules that satisfied the cover constraint for the fourth hospital, whereas noising managed this in all cases.

Combining noising and simulated annealing with SAWing produced little improvement but dramatically lengthened run times. From a practical point of view, the noising method appears to be the best method of those compared here for producing high quality schedules within a reasonable run time.

7 Conclusions

A nurse scheduling system has been developed that has been shown to produce feasible schedules for a variety of datasets. Across eleven datasets from four different hospitals, the system produced better solutions than those produced manually. There are many different objectives to consider, each of differing importance and various strategies for dealing with these have been examined. Combining them into a linear cost function and optimising them using simulated annealing has been compared with using the SAWing technique which places more emphasis on those constraints that are difficult to satisfy. Additionally the noising method has been used to add random variation to the weights. It was found that the noising method worked particularly well and produced schedules for a variety of real datasets that were superior to those produced manually, and also superior to those produced using simulated annealing. All the constraints and objectives mentioned in our survey have been dealt with and this system is sufficiently flexible to deal with different wards placing different emphasis on each objective. The resulting schedules have been examined by nurses from the respective wards and they were happy with the schedules, thought they were fairer to all nurses and were satisfied with the run time of 100 seconds.

Experiments with Phase 3 of our model produced similar results with the noising method again performing strongly. Run times were lower than for Phase 1 at about 10 seconds meaning that in total, schedules are produced within 2 minutes. It seems that these methods are suitable to serve as the basis for a robust nurse scheduling tool which is sufficiently rapid that the user could produce several schedules in a short period of time.

It is intended to perform further research into these methods. Results are robust for different parameter values for the noising method but a limited number of datasets have been tested here. It would be interesting to test the robustness of the algorithm on further datasets and if the parameters outlined here do not prove as successful, to try to identify means of setting parameter values according to characteristics of the datasets. For the noising method, adding additional diversification power, for example by biasing nurses towards new work patterns may enable a wider search of the solution space to take place. Adding intensification, for example by returning to previously found best solutions, may enable the search to home in on better solutions.

References

Abdennadher, A., & Schlenker, H. (1999). Nurse scheduling using constraint logic programming. In *Eleventh annual conference on innovative applications of artificial intelligence (IAAI)*.

Abramson, D. (1991). Constructing school timetables using simulated annealing: sequential and parallel algorithms. *Management Science*, *37*, 98–113.

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

Baker, K. (1974). Scheduling a full time workforce to meet cyclic staffing requirements. *Management Science*, *20*, 1561–1568.

Baker, K., & Magazine, M. (1977). Workforce scheduling with cyclic demands and day off constraints. *Management Science*, *24*, 161–167.

Bellanti, F., Carello, G., Della Croce, F., & Tadei, R. (2004). A greedy based neighbourhood search approach to a nurse rostering problem. *European Journal of Operational Research*, *153*, 28–40.

Berrada, I., Ferland, J. A., & Michelon, P. (1996). A multi-objective approach to nurse scheduling with both hard and soft constraints. *Socio-Economic Planning Science*, *30*, 183–193.

Burke, E., De Causmaecker, P., & Vanden Berghe, G. (1998). A hybrid tabu search algorithm for the nurse rostering problem. In *Lecture notes in artificial intelligence: Vol. 1585. Simulated evolution and learning, selected papers of SEAL, Canberra* (pp. 187–194).

Burke, E., De Causmaecker, P., Petrovic, S., & Vanden Berghe, G. (2003). Variable neighbourhood search for nurse rostering problem. In M. G. C. Resende & J. P. de Sousa (Eds.), *Combinatorial optimization book series. Metaheuristics: computer decision-making* (pp. 153–172). Dordrecht: Kluwer, Chapter 7.

Burke, E., De Causmaecker, P., Vanden Berghe, G., & Van Landeghem, H. (2004). The state of the art of nurse rostering. *The Journal of Scheduling*, *7*, 441–499.

Charon, I., & Hurdy, O. (1993). The noising method: a new method for combinatorial optimization. *Operations Research Letters*, *14*, 133–137.

Charon, I., & Hurdy, O. (2001). The noising methods: a generalization of some metaheuristics. *European Journal of Operational Research*, *135*, 86–101.

Cheng, B. M. W., Lee, J. H. M., & Wu, J. C. K. (1996). *A nurse rostering system using constraint programming and redundant modelling*. Technical report, Department of Computer Science and Engineering at The Chinese University of Hong Kong.

Dowsland, K. A. (1993). Simulated annealing. In C. Reeves (Ed.), *Modern heuristic techniques for combinatorial problems* (pp. 20–63). Oxford: Blackwell.

Dowsland, K. A. (1998). Nurse scheduling with tabu search and strategic oscillation, *European Journal of Operational Research*, 393–407.

Dowsland, K. A., & Thompson, J. M. (2000). Nurse scheduling with knapsacks, networks and tabu search. *Journal of the Operational Research Society*, 825–833.

Eiben, A. E., & van Hemert, J. I. (1999). SAW-ing EAs: adapting the fitness function for solving constrained problems. In *New ideas in optimization* (pp. 389–402). London: McGraw-Hill, Chapter 26.

Frances, M. N. (1966). Implementing a program of cyclical scheduling of nursing personnel. *Hospitals*, 108–125, July 16.

Glover, F. (1989). Tabu search. Part I. *ORSA Journal on Computing*, 190–206.

Howell, J. P. (1966). Cyclical scheduling of nursing personnel. *Hospitals*, 77–85, January 16.

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 671–680.

Lundy, M., & Mees, A. (1986). Convergence of an annealing algorithm. *Mathematical Programming*, 111–124.

Thompson, J. M., & Dowsland, K. A. (1996). Variants of simulated annealing for the examination timetabling problem. *Annals of Operations Research*, *63*, 105–128.

Valouxis, C., & Housos, E. (2000). Hybrid optimization techniques for the workshift and rest assignment of nursing personnel. *Artificial Intelligence in Medicine*, *20*, 155–175.

Wright, M. (1996). School timetabling using heuristic search. *Journal of the Operational Research Society*, *47*, 347–357.