Automated Multi-Skill Shift Design for Hospitals

Eivind J. Nilssen
Martin Stølevik
Erik Lien Johnsen
Tomas Eric Nordlander

SINTEF ICT, Department of Applied Mathematics, Oslo, Norway
eivind.j.nilssen@sintef.no martin.stolevik@sintef.no
erik.lien@sintef.no tomas.nordlander@sintef.no

Abstract: For labour intensive organizations, finding a good match between the predicted workload and the scheduled workforce work capacity is crucial: One important step in this matching process is shift design. In a multi-skill shift design problem, the model must reflect the skill attributes of the employees and the time-dependent demand of each skill type. For hospitals, a number of constraints and objectives complicate the picture, and in this paper we introduce models which reflect many of the challenges faced by planners at two reference hospitals in Norway. Experiments using mixed-integer programming solvers show promising results, and optimal solutions are in some cases found within seconds or a few minutes.

Keywords: Multi-skill shift design, Mixed-integer programming

1. Introduction

Workforce management (WFM), which involves matching workload with workforce, is important within the labour intensive health sector. A mismatch can lead to longer waiting times or expensive labour cost. When looking at the WFM process it is often divided into four steps; Workload prediction, staffing, shift design and/or scheduling, and nurse rostering (Bhulai et al. 2008). By nurse rostering, we mean the short term timetabling of staff, not to be confused with other uses of the term 'rostering' across the literature (Burke et al. 2004). Hospital planners create work schedules by assigning hospital employees to given shifts while considering skills, fairness, laws, regulations, etc.. This is a combinatorial explosive problem which often is handled inefficiently by use of manual procedures (involving highly qualified health personnel). A prerequisite to good nurse rosters is *shift design*, which consists in finding a suitable set of shifts which covers the work demand for the planning horizon. Good shifts are important not only for the hospital (e.g. lowering cost, managing staff coverage, staff satisfaction) but also for its staff (e.g. healthy working shifts). This paper deals with the multi-skill shift design problem for hospitals, where shift details (start and end times, required staffing for each shift and each skill type) are not given as input, but must rather be found as a solution to an optimization problem. We present BeAct, a prototype of a decision support tool that automates multi-skill shift design.

Shift design is the process of creating a set of shifts (start and end times) and assigning staffing needs to the shifts, so that the time-dependent demand for workers is covered while satisfying a set of constraints. The number of possible solutions increase exponentially with the size of the problem: it quickly becomes too large for an efficient manual approach. Optimization methods excel on these types of combinatorial explosive problem: searching for optimality by comparing feasible solutions against each other upon certain criteria (e.g. minimization of personnel cost).

Our idea of automating shift design originates from discussions with AHUS (Akershus universitetssykehus HF) and SAB (Sykehuset Asker og Bærum HF), two Norwegian hospitals in Oslo. It was highlighted that estimating the staffing needs of the shifts for their nurse rostering tool was problematic, while the input required for the shift design problem was relatively easy to specify. These two hospitals have been involved in the development process with ideas, criticisms, and data provision.

We have used mixed -integer programming (MIP) as our solution method and Xpress-MP as the solver. The relevant planning personnel at both hospitals are users of Microsoft Excel, which therefore has been used as the Graphical User Interface; for holding data, parameter values, and for presenting solutions graphically. The mathematical models were constructed through an iterative development process with our two reference hospitals. The constraints in the model include among others: a maximum of different shifts to be used during different periods; overlap between shifts; a common leaving time for all day shifts; one main shift having most of the workers assigned each day. Our model allows for work tasks requiring certain skills that only a part of the workforce possesses. The requirement of meeting the demand is formulated as a ‘hard' constraint, but a possibility for doing some of the tasks at another time than originally specified is also implemented. Some of the model objectives are: minimization of personnel cost; minimization of the total number of different shifts.

The core contribution of this paper is threefold:

1. We introduce BeAct, a prototype for automated multi-skill shift design for hospitals.
2. We target an important but comparatively unattended research domain that is a prerequisite for finding efficient nurse rosters.

3. Our approach is developed in close cooperation with Norwegian hospitals, and tested on synthetic instances that are based on detailed discussions with the hospitals.

Our experiment results show very good results on the instances. On our more advanced tests, within five minutes we find a MIP solution with an objective value gap to the best lower bound (obtained from LP relaxations) usually below 5 %, and in some cases optimal solutions are found within seconds or a few minutes.

The paper is organized as follows: Section 2 provides the background information and related research. The problem is introduced and formally stated in section 3. In section 4, we describe the experiment setup and discuss the results. We conclude and present the planned future work in section 5.

2. Background

For labour intensive organizations, finding a good match between the predicted workload and the scheduled workforce work capacity is crucial. Mismatch leads either to longer waiting times or unnecessarily expensive labour costs. This matching process is part of the so-called workforce management (WFM) and is often done stepwise; predicting work load, staffing, shift design and/or scheduling, and nurse rostering (Musliu et al. 2004; Bhulai et al. 2008). These steps are all interrelated, and should ideally be treated simultaneously, but due to tractability they are often treated sequentially. For the hospitals, these steps would read as; listing the staff tasks (prediction of workload); estimating their required working time (staffing); designing shifts and scheduling them to minimize the mismatch (shift scheduling); and lastly assigning employees to the roster (rostering). In our reference hospitals, most time is spent on the rostering part. The research community have had the same focus, providing a large body of literature on nurse rostering (Burke et al. 2004) for the last 40 years (Ikegami and Niwa 2003). Less attention has been given to optimal shift design, even though good shifts are a prerequisite for high quality nurse rosters.

2.1 Shift Design

The process of shift design consists of generating a set of shifts (which fulfils requirements on minimum and maximum shift duration, legal start and end times, etc.), with an associated staffing demand assigned to each shift, so that the time-dependent and skill-dependent demands are satisfied in all time periods. The demands are supplied as input to the shift design problem, typically covering 24h with 15 minutes intervals, and varying between the days of the week. Finding an optimal shift design involves evaluating different solutions against each other through user specific objectives (e.g. minimizing cost, minimizing the number of shifts, ensuring overlap between shifts, etc.). For hospitals, a multi-skill feature must be considered during shift design; because the staff skills and work tasks skill requirements vary, a skill matching process is needed for efficient shift design. For real size problems the solution space is too large for an efficient manual approach: Even though finding a solution manually is possible, it is unlikely to be an optimal or even good solution. Still, shifts are in most cases designed manually.

2.2 Related Work

The literature on shift design is sparse and driven by the growth of call-centres, where the personnel cost is the predominant one (60-70% (Fukunaga et al. 2002). A range of solution methods have been investigated on the shift design problem: For example, constraint programming, integer and mixed-integer programming, and local search (Musliu et al. 2004; Canon 2005; Bhulai et al. 2008).

The most relevant research for us are (Musliu et al. 2004) and (Bhulai et al. 2008). Musliu et al. have similarities but our research differs on several points. Firstly, while they focus on solving the general shift design problem, we focus on modeling and solving the shift design problem for hospitals. Secondly, their attention is on the algorithm and how it works, while our focus is on modeling the problem and developing a prototype for hospitals. Thirdly, their work is on single skilled shift design while ours is on multi-skill shift design. Lastly, they use local search, an approximate solution method that can not guarantee finding optimality (but is known to often find good solutions in a relatively short time). We are using MIP, an exact solution method, and we manage to find optimality within seconds for several of the cases.

Another closely related work is conducted by Bhulai et al. (Bhulai et al. 2008). Both our work relates to multi-skilled shift design and while we both apply exact solution methods, their paper centres on presenting a new method while ours also presents a prototype. Another difference is that they focus on solutions for call centres while we aim for the hospital domain. Lastly, theirs is a generalized model while ours is tailored with more constraints.

3. Mathematical Model

As already mentioned in the introduction, our proof-of-concept model aims at designing shifts (start and end times) and assigning a number of workers to these shifts, thereby satisfying the demand of workers for each time slot. The demand is given as the number of employees needed, for each skill type and each time period (every 15th minute the entire day), for all weekdays. Furthermore, the workforce can be subdivided into groups based on skill attributes. All workers in a group have exactly the same set of skills. The cost of having an employee assigned to a shift is given for each group and each time slot. The cost is different for time slots belonging to daytime, evening, and night intervals.

We now state the mixed-integer programming models that are used to design shifts. We start with the simplest model that minimizes the cost while satisfying the demand. Later we present the modifications needed in order to take into account additional realistic constraints and objectives.

The notation used in these models is presented first. The indices and sets are:

s : shift (defined by a start and end time).

d : day.

c : skill.

g : group of workers, all with the same set of skills.

S : set of possible shifts.

S_t : set of shifts that cover time slot *t*.

D : set of days, {Monday, Tuesday, ..., Sunday}.

C : set of skills.

G : set of groups.

$G_{C'}$: set of groups that have all of the skills in $C' \subseteq C'$. In other words: workers belonging to any of the groups in $G_{C'}$ have at least all of the skills in *C'*.

The parameters are:

K_{sdg} : the cost of one member of group g working shift *s* on day *d*.

B_{tdc} : the demand for skill *c* in period *t* on day *d*.

The decision variables are:

$x_{sdg} \in \{0, 1, ...\}$ = the number of members from group g working shift s on day d.

The model, in its simplest for, is then:

$$\text{minimize} \sum_{s\in S, d\in D, g\in G} K_{sdg} x_{sdg} \tag{3.1}$$

such that

$$\sum_{s\in S_t, g\in G_{C'}} x_{sdg} \geq \sum_{c\in C'} B_{tdc}\,, \quad t \in T, d \in D, C' \subseteq C \tag{3.2}$$

Constraint 3.2 says that the demand must (at least) be met in all time periods, on all days and for all sub sets of skills. This generates a lot of constraints, out of which many are redundant and are removed before entering the solver stage.

Before we look at a more extended model we introduce more notations. The additional indices and sets are:

i : an interval of the day (day, evening or night).

s' : shift.

I : the set of intervals for 24 hours: {day, evening, night}.

$S_{\mathrm{day}} \subset S$: the set of day shifts.

S_{M} : the set of possible main shift candidates, which is a subset of S_{day}.

$S_{\mathrm{evening}} \subset S$: the set of evening shifts.

$S_{\mathrm{night}} \subset S$: the set of night shifts.

T_{E} : the set of possible end times for day shifts.

S_t^E : the set of shifts ending at the end of time period t.

Additional parameters are:

z_{ig} : the maximum allowed number of shifts used in one day in interval i for group g.

η : the fraction of the day workers which should be on each main shift.

n : the maximum number of different main shifts allowed per day. n is typically set to 1.

τ_s : the length of shift s.

$M^1{}_{sg}$, $M^2{}_{sdg}$, $M^3{}_{s'dg}$ and $M^4{}_{td}$ are different big M constants.

Additional help variables are:

$y_{sg} \in \{0, 1\} = 1$ if shift s is used by group g; 0 otherwise.

$z_{sdg} \in \{0, 1\} = 1$ if shift s is used by group g on day d; 0 otherwise.

$e_{td} = 1$ if some shifts are ending at the end of time period t on day d; 0 otherwise.

We may have objective components other than minimizing the total cost. The following objective is minimizing the number of shifts used:

$$\text{minimize} \sum_{s \in S, g \in G} y_{sg} \qquad (3.3)$$

We also have to add the following constraints to model y_{sg} and z_{sdg}:

$$\sum_{d \in D} x_{sdg} \leq M^1_{sg} y_{sg}, \quad s \in S, g \in G \tag{3.4}$$

$$x_{sdg} \leq M^2_{sdg} z_{sdg}, \quad s \in S, d \in D, g \in G \tag{3.5}$$

The effect of this 'big-M' notation is that y_{sg} and z_{sdg} can be zero only when all x$_{sdg}$ in the corresponding constraints are zero.

The following constraints are added to limit the number of day, evening and night shifts used for each group, each day:

$$\sum_{s \in S_i} z_{sdg} \leq \overline{z}_{ig}, \quad i \in I, d \in D, g \in G \tag{3.6}$$

Constraints demanding at least η percent of the workers to work on the main shifts and maximum n main day shifts for each group, each day:

$$x_{s'dg} - M^3_{s'dg}(z_{s'dg} - 1) \geq \eta \sum_{s \in S_{day}} x_{sdg}, \quad s' \in S_M, d \in D, g \in G \tag{3.7}$$

$$\sum_{s' \in S_M} z_{s'dg} \leq n, \quad d \in D, g \in G \tag{3.8}$$

Constraints for common end time, i.e. all day shifts ending at the same time:

$$\sum_{t \in T_E} e_{td} \leq n_E \quad \forall \quad d \in D \tag{3.9}$$

$$\sum_{s \in S^E_t, g \in G} x_{sdg} \leq M^4_{td} e_{td} \quad \forall \quad t \in T_E, d \in D \tag{3.10}$$

$$x_{sdg} = 0 \quad \forall \quad s \in S_{day} \setminus \bigcup_{t \in T_E} S^E_t \tag{3.11}$$

$$e_{td} \in \{0, 1\} \quad \forall \quad t \in T, d \in D \tag{3.12}$$

3.11 states that day shifts not ending on possible end times (t ∈ T_E) cannot be used.

Constraint to ensure overlap between shifts:

$$\sum_{s \in S_t^E} x_{sdg} < \sum_{s \in S_t} x_{sdg} \quad \forall \quad t \in T, d \in D, g \in G \qquad (3.13)$$

3.13 states that there must be more people at work in each time period than those quitting in the end of that time period.

If it is not necessary to do the tasks in exactly the specified period, the demand should not be hard constraints. A way to do this is to allow for changes in demand, within some limits, by modifying the demand constraint 3.2. First we introduce some additional parameters and help variables.

Additional parameters:

$\overline{\delta}_c^{+}$: the maximum allowed increase in demand for skill *c* in any period.

$\overline{\delta}_c^{-}$: the maximum allowed decrease in demand for skill *c* in any period.

$\underline{b}_c$: the minimum demand for skill *c* after a decrease in any period.

Δ_c : the maximum change in demand for skill *c* allowed during one day. Note: In the code only changes of the demand for skill level 1 is allowed, which is equal to setting $\Delta_c = 0$ and/or $\overline{\delta}_c^{+} = \overline{\delta}_c^{-} = 0$

Additional help variables:

δ^{+}_{tdc} : the increase in demand for skill c at time t on day d.

δ^{-}_{tdc} : the decrease in demand for skill c at time t on day d.

The soft demand constraints model then becomes:

$$\sum_{s \in S_t, g \in G_{C'}} x_{sdg} \geq \sum_{c \in C'} (B_{tdc} + \delta^{+}_{tdc} - \delta^{-}_{tdc}) \quad \forall \quad t \in T, d \in D, C' \subseteq C \qquad (3.14)$$

$$\overline{\delta}_c^{+} \geq \delta^{+}_{tdc} \geq 0 \quad \forall \quad t \in T, d \in D, c \in C \qquad (3.15)$$

$$\min(\max(0, b_{tdc} - \underline{b}_c), \overline{\delta}_c^{-}) \geq \delta^{-}_{tdc} \geq 0 \quad \forall \quad t \in T, d \in D, c \in C \qquad (3.16)$$

$$\sum_{t \in T} \delta^{-}_{tdc} \leq \Delta_c \quad \forall \quad d \in D, c \in C \qquad (3.17)$$

$$\sum_{t \in T} (\delta^{+}_{tdc} - \delta^{-}_{tdc}) \geq 0 \quad \forall \quad d \in D, c \in C \qquad (3.18)$$

Where 3.15 and 3.16 gives limitations on the change in each period (the min function in the latter is to ensure that the demand is not set below a certain limit), 3.17 limits the total change each day, and 3.18 makes sure that the total demand after the changes is not lower than before.

4. Empirical Studies

Our case study refers to the shift design problem faced by our reference hospital in Oslo, AHUS. We conduct several experiments concerning variations of the model. Models were implemented in MOSEL and solved by XPRESS-MP 19.00 on an Intel Xeon 3.0 GHz microprocessor. A simple GUI in Excel was developed to demonstrate the BeAct prototype and enable discussions with the end users. The experiments are based on synthetic parameter values and 24-hour demand data, with realistic values set based on input from our reference hospital. The use of synthetic data is justified by the focus of this work as a proof-of-concept, and by the present lack of high-quality real 24-hour demand data suitable for our purposes; our reference hospitals have plans for improving and standardizing their 24-hour demand data. The 24-hour demand data is subdivided into time slots of 15 minutes, and the demand values in a single time slot are for each skill type. The 24-hour demand data can be different for different weekdays.

According to our data, the workforce (nurses, head nurse, and so on) is subdivided in up to 5 groups, each group having the same skill set. The skills in a skill set can be of 5 different skill types. In cases with hierarchical groups, the lowest group contains only one skill type, whereas the most advanced worker group contains all the skill types. Our multi-skill case studies have focused on the slightly more difficult problem where the skill groups can have arbitrary skill type configurations. A cost matrix defines the hourly monetary cost of scheduling workers from each skill group, with different parameters for day time, evening, night, and week-end hours. The 24-hour demand data is a 24-hour demand curve which details, for every 15th minute, the demand for each skill type. In our easy test cases, the 24-hour demand is the same for all weekdays (except week-end), whereas the more difficult cases have demands that vary both intra- and inter-day.

The shifts designed to meet the demand must satisfy some time-related constraints. Specifically for our test cases, minimum and maximum durations of shifts are 6 hours and 10 hours, respectively. The earliest start time of a shift is 07:00, and the latest start time is 22:30. No shifts can end between 00:00 and 07:00. All shift durations are integer multiples of 30 minutes. In a pre-processing stage, the tool generates all possible shifts adhering to these constraints. For models respecting the overlap constraint, at least one worker from the preceding shift must overlap at least 30 minutes with the next shift. This does not require equal skill groups of the overlapping shifts. For models respecting the main shift constraint, one main shift is required to start between 07:00 and 09:00, its duration must be 7-8.5 hours, and at least 70 % of the daytime workers must be assigned to the main shift. Although the model supports a parameterized number of main shifts, for our experiments we have limited this parameter to 1 (i.e., a single main shift per group. For the reported results, this is also a single main shift per week, although the model can handle different main shifts from day to day).

An early input from our end users was to include the common end time constraint in some models, due to organizational issues (transfer of information between shifts). For these cases, shifts that end between 12:00 and 18:00 must end at the same time. Another desire was to test models where there is a maximum number of shifts in use within the three intervals day, evening, and night. For these cases, we set the maximum to 3 day shifts, 2 evening and 2 night shifts, per day and per skill group.

Table 1 summarizes the results of our experiments on these cases. All of the reported cases are for multi-skill problems with arbitrary skill sets for each group of workers (i.e., no hierarchical configuration), different 24-hour demand data each day, and a basic objective of minimizing the monetary cost. The table reports the plan costs, run times, and gaps to lower bounds (if any) for the different cases; Basic multi-skill, main shift requirement, common end time for day shifts, limited number of shifts for day, evening and night time, the overlap between shifts constraint, and all of the above combined. For the combined problem, the solver times out after 1 hour, with a 6,55 % optimality gap to the LP relaxation. We note that for the corresponding combined problem in a single skill setting, the solver finds the optimal solution after 630 seconds (not shown in the table).

Type	**Cost**	**# shifts**	**Runtime (s)**	**Gap**
Basic	160,2	58	1	0
Main shift	165,0	39	1	0
Common end	164,6	44	5	0
# shifts in intervals	156,7	37	300	1,55 %
Overlap	152,9	55	1	0
All	183,8	46	3600	6,55 %

Table 1. Experimental results for the different constraint models, showing the monetary cost of the plans, the total number of shifts per week, the runtime in seconds, and the optimality gap.

The reported results used the single objective of minimizing cost. Replacing this with a weighted sum of objective components *cost* and the *number of shifts* in use, the runtime increases noticeably. We remark that the problem of minimizing the number of shifts in use is NP hard and also hard to approximate (Musliu, Schaerf et al. 2004). Also, for test runs where the main shift is allowed to vary from day to day, the runtime increases noticeably compared to the case of a single main shift for the entire week. Note that depending on the demand, the main shift constraint may force an undesirable large assignment of workers to evening shifts—due its shorter maximum duration.

Next, we have also experimented with the flexible model where demand can be moved in time, to a limited degree. Smoother demand curves should enable the solver to find better shift designs. So far, we have only experimented with modifying the demand for a single skill type. The moved demand must stay within the same day, the maximum amount of moved demand is limited to a total of 8 hours per day, and there is an upper bound on the amount of demand that can be moved in a single time slot. Early results indicate that the model works according to our intentions, with smoother demand curves and fewer shifts as a result. More input on end-user requirements is needed before further experiments on this aspect can be conducted.

5. Conclusion and Future Work

In this paper, we present a novel approach that will assist users in both tactical as well as in strategic staff planning responsibilities. Our work focuses on the development of a proof-of-concept prototype for hospital planners, with a variety of models implemented that reflects real-world constraints. We have tested our approach using instances of synthetic data based on input from our reference hospital, and present the results in this paper. They show that the approach manages to solve shift design instances which are realistic both in size and in problem characteristics. The developed tool handles shift design for multi-skill demand cases given a workforce which is grouped based on employees' skill attributes, and hospital-specific constraints are modeled, such as the optimal design of a main shift and a common end time for day shifts. Variants of these implementations are the subject of future research. Also, the related problem of distributing excess capacity in a robust manner, given a pre-specified workforce size, has been implemented, but more work is needed here. Along with our reference hospitals, we plan to continue our work by improving the model and developing a tool that allows them assistance in both strategic and tactical decisions.

6. Acknowledgements

We wish to thank Øyvind Bakke, Elisabeth Reitan, and Henning Moen from the hospitals AHUS (Akershus universitetssykehus HF) and SAB (Sykehuset Asker og Bærum HF) for providing us with data and their precious comments and suggestions throughout our research.

7. References

Bhulai S, Koole G, and Pot A (2008). Simple Methods for Shift Scheduling in Multiskill Call Centers. *Manufacturing & Service Operations Management* **10**(3): 411-420.

Burke EK, De Causmaecker P, Berghe GV, and Landeghem HV (2004). The State of the Art of Nurse Rostering. *Journal of Scheduling* **7**(6): 441-499.

Canon C. (2005). *Personnel scheduling in the Call Center industry*. PhD thesis, Université François-Rabelais de Tours, Laboratoire d'Informatique, Tours, France.

Fukunaga A, Hamilton E, Fama J, Andre D, Matan O, and Nourbakhsh I (2002). Staff Scheduling for Inbound Call Centers and Customer Contact Centers. *In Eighteenth National Conference on Artificial intelligence*: 822-829.

Ikegami A and Niwa A (2003). A subproblem-centric model and approach to the nurse scheduling problem. *Journal Mathematical Programming* **97**(3): 517-541.

Musliu N, Schaerf A, and Slany W (2004). Local search for shift design. *European Journal of Operational Research* **153**(1): 61-64.