A mathematical model for the nurse rostering problem

Martin Stølevik[a], Tomas Eric Nordlander[a], Atle Riise[a]

(Martin.Stolevik@sintef.no, Tomas.Nordlander@sintef.no,
Atle.Riise@sintef.no)

[a]*SINTEF ICT, Department of Applied Mathematics,*
P.O. Box 124 Blindern, NO-0314 Oslo, Norway.

1   Introduction

In this report we describe a mathematical model for the nurse rostering problem (NRP). A thorough discussion of the NRP and different models can be found in [1].

The model we present in this report is mainly developed on work done together with the company Gatsoft AS. With minor changes and extensions, a version of the model has been implemented and delivered to Gatsoft as a plug-in to the rostering part of their software GAT.

The model consists of two types of constraints; hard and soft. The hard constraints must never be violated. The origin of the hard constraint is typically legislation or working acts. The soft constraints can be violated, but we try to avoid it. Violating the soft constraints is usually necessary to be able to find a solution. A solution to the model is one in which all hard constraints are satisfied. The quality (objective function value) of the solution is measured in terms of violations to the soft constraints.

2   Model

2.1   Symbols and terminology

We define the following sets:

S : The set of shift types. A shift $s \in S$ is a member of one and only one shift category $c \in C$. (see also A_{sc} below).

C : The set of shift categories. A shift category is a "collection" of shifts that usually are at approximately the same time of day. The three categories "Day", "Evening", and "Night" are predefined and always present.

D : The set of days.

E : The set of employees.

I_d : The set of incompatible pairs of shifts on day d and $d + 1$ because the time between the shifts are too short. I_d is generated after input by finding all pairs (s, s') of shifts where the time between the end of s and the start of s' is shorter than either; (1) the minimum time between a night shift and an evening shift, (2) the minimum time between an evening shift and a day shift, or (3) the minimum time between a day shift and a night shift. For example; if a night shift ends at 07:00, and an evening shift starts at 15:00, then this pair is added to I_d if the parameter (1) is larger than 8 hours on day d.

U : The set of unwanted shift patterns.

V : The set of wanted shift patterns.

W : The set of weeks.

The following parameters are used in the mathematical model:

p : The maximum percentage of deviation over the planning period from an employee's contracted working time.

N_c^{min} : The minimum number of consecutive shifts of category c for any employee. Typically used to avoid single night shifts by setting this value to 2.

N_c^{max} : The maximum number of consecutive shifts of category c for any employee.

N^{min} : The minimum number of consecutive shifts of any category for any employee.

N^{max} : The maximum number of consecutive shifts of any category for any employee.

N_{ec}^{max} : The maximum number of shifts of category c that employee e should work.

N_{ec}^{min} : The minimum number of shifts of category c that employee e should work.

P_{ds} : The number of shifts of type s required on day d (i.e., the manpower plan).

T_e : Employee e's maximum weekly working time.

T_e^h : The working time employee e is supposed to work over the planning horizon according to his/her employment contract.

T^f : Minimum time of continuous free time (no work) each week.

T_s : Duration of shift s.

A_{sc} : 1, if shift s is in shift category c, 0 otherwise.

B_{es} : 1, if employee e has the competence to work shift s, 0 otherwise.

We define the following binary decision variables:

x_{eds} : 1, if shift s is assigned to employee e on day d, 0 otherwise.

A *solution*, $\mathbf{x}$, to the nurse rostering problem consist of assigning a value to every decision variable (x_{eds}) in the problem so that no hard constraints are violated. The *quality of a solution* is measured by the value of the objective function (see 2.5) which is a linear combination of the penalties associated with violating the soft constraints.

2.2 Derived Solution Information

To reduce the complexity of the mathematical model we define some derived solution information. The value of the functions below can easily be computed from a given solution $\mathbf{x}$.

$f_{ew}(\mathbf{x})$: The length of the longest free period for employee e in week w in solution $\mathbf{x}$. Measured in the same unit as T^f.

$u_{ei}(\mathbf{x})$: The number of unwanted shift patterns of type i for employee e in solution $\mathbf{x}$.

$w_{ed}(\mathbf{x})$: A number in the range $[0, 1]$ indicating if the shift for employee e on day d is "close" to being part of a complete wanted shift. $w_{ed}(\mathbf{x})$ is set to 1 either if the shift (possibly together with neighboring shifts) is part of a wanted pattern. Values less than 1 is used if the shift on this day (and possibly together with neighboring shifts) is close to a wanted pattern but not exactly equal. Thus the penalty will increase the farther "away" from a wanted pattern one is.

2.3 Hard Constraints

These are our hard constraints, which cannot be violated:

$$\sum_{s \in S} x_{eds} \leq 1 \ , \quad \forall \, e \in E, d \in D \tag{1}$$

$$\sum_{e \in E} x_{eds} = P_{ds} \ , \quad \forall \, d \in D, s \in S \tag{2}$$

$$(1 - p)T_e^h \leq \sum_{d \in D, s \in S} T_s x_{eds} \leq (1 + p)T_e^h \ , \quad \forall \, e \in E \tag{3}$$

$$\sum_{s \in S} B_{es} x_{eds} = 1 \ , \quad \forall \, e \in E, d \in D \tag{4}$$

$$x_{eds} + x_{e(d+1)s'} \leq 1 \ , \quad \forall \, e \in E, d \in D, (s, s') \in I_d \tag{5}$$

$$f_{ew}(\mathbf{x}) > T^f \ , \quad \forall \, e \in E, w \in W \tag{6}$$

$$\sum_{d=7w}^{7w+6} \sum_{s \in S} T_s x_{eds} \leq T_e \ , \quad \forall \, e \in E, w \in W \tag{7}$$

Equation 1 says that maximum one shift is assigned each day to each employee. In the case of a day off, no shift will be assigned. Equation 2 (sometimes refered to as the 'vertical sum constraint' or 'cover requirement') ensures that the manpower plan is covered exactly each day. It is not uncommon in other models to require that there is at least as many employees working as specified in the manpower plan (thus 2 would become the $\sum_{e \in E} x_{eds} \geq P_{ds}$, $\forall \, d \in D, s \in S$). Equation 3 says that the sum of working hours for each employee must not deviate too much with respect to the employee's contracted hours (T_e^h) over the planning period. Equation 4 makes

sure that employees only work shifts for which they have the required competence. The minimum time between shifts on consecutive days is handled by Equation 5. In Equation 6 we make sure that every week has a minimum continuous free period. In Norway, this constraint is typically used to handle the "F1" constraint: Every week for all employees should contain a free shift (named "F1"). Lastly, Equation 7 ensures that the maximum weekly working time is not violated.

2.4   Soft constraints

To be able to handle the actual scheduling problems face in nurse rosterin, the model must be rich enough. We solve a rich optimization problem with quite many soft constraints.

Since the soft constraints might be violated, a penalty function is associate to every type of violation. The penalties are weighted and summed in the objective function (see end of this section). The penalty functions are:

$$p_1(\mathbf{x}) = \sum_{c \in C} \sum_{e \in E} \sum_{d=1}^{|D|-N_c^{max}} \left(\prod_{d'=d}^{d+N_c^{max}} y_{ecd'} \right) \tag{8}$$

$$p_2(\mathbf{x}) = \sum_{e \in E} \sum_{d=1}^{|D|-N^{max}} \left(\prod_{d'=d}^{d+N^{max}} z_{ed'} \right) \tag{9}$$

$$p_3(\mathbf{x}) = \sum_{c \in C} \sum_{e \in E} \sum_{d=1}^{|D|-N_c^{min}} \max\left(0, y_{ec(d+1)} - y_{ecd} \right) \left(N_c^{min} - \sum_{d'=d+1}^{d+N_c^{min}} \left(\prod_{d''=d+1}^{d+N_c^{min}} y_{ecd''} \right) \right) \tag{10}$$

$$p_4(\mathbf{x}) = \sum_{e \in E} \sum_{d=1}^{|D|-N^{min}} \max\left(0, z_{e(d+1)} - z_{ed} \right) \left(N^{min} - \sum_{d'=d+1}^{d+N^{min}} \left(\prod_{d''=d+1}^{d+N^{min}} z_{ed''} \right) \right) \tag{11}$$

$$p_5(\mathbf{x}) = \sqrt{\sum_{c \in C} \sum_{e \in E} \left(\max(0, N_{ec}^{min} - \sum_{d \in D} y_{ecd}, \sum_{d \in D} y_{ecd} - N_{ec}^{max}) \right)^2} \tag{12}$$

$$p_6(\mathbf{x}) = \sqrt{\sum_{e \in E} \left(T_e^w - \sum_{d \in D} \sum_{s \in S} T_s x_{eds} \right)^2} \tag{13}$$

$$p_7(\mathbf{x}) = \sqrt{\sum_{e \in E} \left(\sum_{d=1}^{|D|-1} \max\left(0, z_{ed} - z_{e(d+1)} \right) \right)^2} \tag{14}$$

$$p_8(\mathbf{x}) = |E||D| - \sum_{e \in E, d \in D} w_{ed}(\mathbf{x}) \tag{15}$$

$$p_9(\mathbf{x}) = \sum_{e \in E, i \in U} u_{ei}(\mathbf{x}) \tag{16}$$

Where

$y_{ecd} = \sum_{s \in S} A_{sc} x_{eds}$ is 1 if employee e is working a shift in category c on day d, 0 otherwise.

$z_{ed} = \sum_{s \in S} x_{eds}$ is 1 if employee e is working on day d, 0 otherwise.

The penalty functions associated with the soft constraints are:

Eq 8: *Avoid too many consecutive working days with the same shift category.*

Eq 9: *Avoid too many consecutive working days.*

Eq 10: *Avoid too few consecutive working days with the same shift category.* This soft constraint is typically used to avoid single night shifts by setting $N_{c'}^{min} = 2$ (where c' is the night shift category).

Eq 11: *Avoid too few consecutive working days.* Typically used to avoid working periods of just one or two days.

Eq 12: *Minimize deviation from minimum and maximum number of shifts in each category.* The number of shifts in each category c for an employee (e) should be between a minimum (N_{ec}^{min}) and a maximum (N_{ec}^{max}) limit.

Eq 13: *Minimize deviation from the employee's available working time.* One of the attributes of an employee is the availability; how much time that employee is expected to work throughout the planning horizon. The expected amount of working time for employee e over the planning period is T_e^w.

It is in general impossible to find a solution where all employees work exactly the number of minutes they are supposed to work according to their working contract. Any deviations are compensated monetary or by adjusting the employee's expected working time for the next planning period. Although deviations can be handled this way, one wants to minimize the deviation.

In a solution, the actual working time of employee e is $\sum_{d,s} T_s x_{eds}$. The deviation to the contracted working time is squared so that many small deviations are preferred to fewer and larger.

Eq 14: *Cluster days off.* It is desirable to cluster days off. Instead of a day off on Tuesday and then again on Thursday, it would be better to have a consecutive period of two days off, either Tuesday-Wednesday

or Wednesday-Thursday. To reach a penalty of zero is in general impossible since that requires only one free period per employee and that would typically violate some of the hard constraints.

Eq 15: *Maximize wanted shift patterns.* We want to maximize the number of wanted shift patterns. The penalty decreases with the number of wanted patterns found. The large number $(|E||D|)$ in the penalty ensures that the penalty never becomes negative.

Eq 16: *Minimize unwanted shift patterns.* This soft constraint penalizes unwanted patterns. A penalty of 1 is added for every unwanted pattern found.

2.5 Objective function

The total objective value is equal to a weighted sum of the different penalties. Thus, we have the following objective function:

$$\text{Minimize:} \ \ f = \sum_{m=1}^{9} K_m p_m \tag{17}$$

Where $K_m = K_m^1 K_m^2 K_m^3$. These three constants are set or computed by different stakeholders. K_m^1 is computed in our software based on the case being run, to handle the fact that the different penalties are on different numeric levels. As an example, let's look at penalty p_6 and p_9; p_6 includes terms of working time which can grow pretty large whilst p_9 is simply a count of the number of unwanted patterns occurring in the current solution, which is typically low, especially if there is a low number of unwanted patterns. After p_m is multiplied by K_m^1, the product will be in or close to the range $[0, 1]$. This means that the different terms in f will have more or less equal impact. K_m^2 are fixed constants set in our software to reflect a weighting between the penalties that from our experience causes the most efficient search for good solutions. Finally, K_m^3 is set by the user of our model, the software system GAT, to reflect their opinion of the relative importance of the penalties. Initially, the weights are preset in GAT, but they are tunable by the users in an interface for advanced users.

References

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