Nurse Scheduling Web Application

Zdeněk Bäumelt[1,2], Přemysl Šůcha[1], Zdeněk Hanzálek[1,2]

[1]Department of Control Engineering, Faculty of Electrical Engineering
Czech Technical University in Prague, Czech Republic, {baumez1, suchap, hanzalek}@fel.cvut.cz

[2]Merica s. r. o., Czech Republic, {imedica, hanzalek}@merica.cz

Abstract

The focus of this paper is on the development of a web application for solving Nurse Scheduling Problem. This problem belongs to scheduling problems domain, exactly timetabling problems domain. It is necessary to consider large amount of constraints and interactions among nurses, that can be simplified through web access.

Introduction

Preparation of multishift schedule is rather difficult process which incorporates couple of constraints (e.g. minimum number of nurses for each type of shift, nurses' workload, balanced shift assignment) and interaction of several users (nurses' requests consideration). Even though single-user nurse scheduling applications avoid rather painful manual process, they do not allow easy access of all nurses to interact with each other. This problem can be efficiently solved using modern web technologies, while carefully considering all specific features of such application; e.g. large amount of human interactions, dramatic impact on satisfaction of individual nurse as well as good mood in nurse team.

Definition of Nurse Scheduling Problem

Nurse Scheduling Problem (NSP) is NP-hard problem, that belongs to timetabling or personnel scheduling domain. The solution of this problem should satisfy all constraints, that are set on the input. With larger instances (growing with number of nurses, number of days in schedule, set of constraints) NSP comes to the combinatorial explosion and it is harder to find an optimal solution.

Related Works

There are several views for solving NSP. In background paper (Hung 1995) there is a history of NSP research from the 60's to 1994. Other bibliographic survey with one described approach is in (Cheang *et al.* 2002). More actual survey is presented in (Burke *et al.* 2004).

On one hand, there is the branch of optimal solution approaches. It includes linear programming (LP) and integer linear programming (ILP) (Eiselt & Sandblom 2000). On the other hand, there are some heuristic approaches. One way to find some solution is to use artificial intelligence methods (e.g. declarative and constraint programming (Okada 1992) or expert systems (Chen & Yeung 1993)). The second way is to use some metaheuristics (simulated annealing, tabu search (Berghe 2002) or evolutionary algorithms (Aickelin 1999)).

Contributions

This paper uses Tabu Search approach and the main contribution of this work lies in application structure designed for access via web.

Application Structure

The structure of Nurse Scheduling Web Application (NSWA) is shown in Figure 1. Users can work with the application via common web browsers. All application blocks are on the server side, which brings many other advantages (operating system independence, no installation and upgrades on client side). The scheduling algorithm runs independently as a web service and exchanges data with application and database through communication interface.

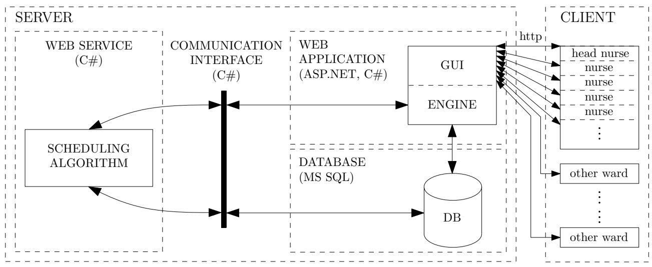


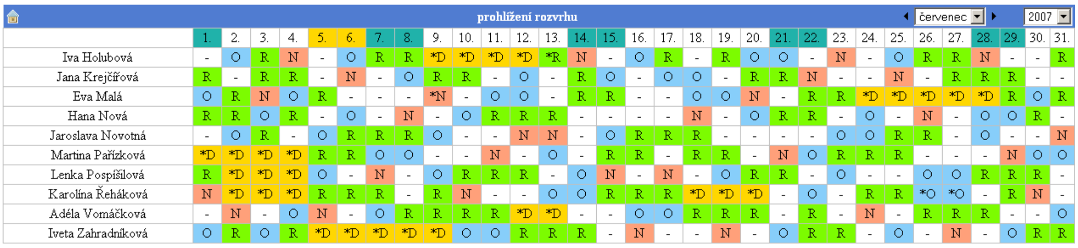
Figure 1: NSWA structure - block design.

Scheduling Algorithm

We decided to use a scheduling algorithm that is based on multicriterial programming implemented as Tabu Search metaheuristic.

Mathematical Model

Our mathematical model is designed as three-shift model – early (E), late (L) and night shift (N) (in Figure 2 early (R),



prohlížení rozvrhu červenec 2007

	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.
Iva Holubová	-	O	R	N	-	O	R	R	*D	*D	*D	*D	*R
Jana Krejčířová	R	-	R	R	-	N	-	O	R	R	-	O	-
Eva Malá	O	R	N	O	R	-	-	-	*N	-	O	O	-
Hana Nová	R	R	O	R	-	O	-	N	-	O	R	R	R
Jaroslava Novotná	-	O	R	-	O	R	R	R	O	-	-	N	N
Martina Pařízková	*D	*D	*D	*D	R	R	O	O	-	-	N	-	O
Lenka Pospíšilová	R	*D	*D	*D	O	-	N	-	O	R	R	R	-
Karolína Řeháková	N	*D	*D	*D	R	R	R	-	R	N	-	-	O
Adéla Vomáčková	-	N	-	O	N	-	O	R	R	R	R	*D	*D
Iveta Zahradníková	O	R	O	R	*D	*D	*D	*D	*D	O	O	R	R

	14.	15.	16.	17.	18.	19.	20.	21.	22.	23.	24.	25.	26.	27.	28.	29.	30.	31.
Iva Holubová	N	-	O	R	-	R	O	O	-	N	-	O	R	R	N	-	-	R
Jana Krejčířová	R	O	-	O	O	-	R	R	N	-	-	N	-	R	R	R	-	-
Eva Malá	R	R	-	-	O	O	N	-	R	R	*D	*D	*D	*D	*D	R	O	R
Hana Nová	-	-	-	-	N	-	O	R	R	-	O	-	N	-	O	O	R	-
Jaroslava Novotná	-	O	R	R	R	-	-	-	-	O	O	R	R	-	O	-	-	N
Martina Pařízková	-	R	R	-	R	R	-	N	O	R	R	R	-	-	-	N	O	O
Lenka Pospíšilová	O	N	-	N	-	O	R	R	-	O	-	-	O	O	R	R	R	-
Karolína Řeháková	O	R	R	R	*D	*D	*D	-	O	-	R	R	*O	*O	-	R	N	-
Adéla Vomáčková	-	-	O	O	R	R	R	-	R	-	N	-	R	R	R	-	-	O
Iveta Zahradníková	R	-	N	-	-	N	-	O	R	R	-	O	-	N	-	O	R	R

Figure 2: Screenshot from our Nurse Scheduling Web Application (july 2007).

late (O), night shift (N), holiday (D) – shifts with star are requested shifts by nurses). Coverage is per shift, under and over coverage is not allowed. We decided for one month scheduling period, because of data export to salary administration. There are no qualification groups (all nurses have the same qualification) and we considered full-time workload in this version of mathematical model.

We optimize objective function Z

$$\min_{x \in X}(Z(x)) \tag{1}$$

where x is one schedule from X state space of schedules. There are two types of constraints in our mathematical model.

- hard constraints have to be fulfilled always
- soft constraints with penalization $f_j(x)$ that are the subject of objective function $Z(x)$, which is defined as

$$\begin{gathered} Z(x) = \mathbf{w} \cdot \mathbf{f}(x) = \sum_{j=1}^{d} w_j f_j(x), \\ w_j \geq 0, d = \dim(\mathbf{f}(x)) \end{gathered} \tag{2}$$

where $\mathbf{w}$ is a vector of weights given by user, $\mathbf{f}(x)$ is a vector function of constraints penalization and d is a number of soft constraints. In our algorithm we considered the following constraints:

Hard Constraints

- required number of nurses for each shift type (#RE, #RL, #RN)
- to consider days from previous month (#H)
- nurses' requests consideration (#R)
- one shift assignment per day (hc1a)
- no early shift after night shift assignment (hc1b)
- no more than five consecutive working days (hc2)
- forbidden shift combinations (FC)

Soft Constraints

- nurses' work-load balance (sc1)
- nurses' day/night shift balance (sc2)
- nurses' weekend work-load balance (sc3)
- avoiding isolated working days (sc4)
- avoiding isolated days-off (sc5)

Head nurses can choose which of hard constraints will be used in our algorithm. Soft constraints are weighted by the head nurses as well. Some hard constraints (hc2, FC) have been converted to the soft constraints with very large weights compared to weights of soft constraints sc1, sc2, sc3, sc4, sc5.

The outline of full Nurse Scheduling Algorithm is described in Algorithm 1 below.

Algorithm 1 – Nurse Scheduling Algorithm

1. read the scheduling parameters and the nurses requests;
2. find a feasible solution x_{init} satisfying hard constraints;
3. optimization (Algorithm 2);
4. user choice
   - schedule is acceptable, goto 7;
   - schedule is acceptable with manual corrections, goto 6;
   - schedule is not acceptable, user can reconfigure scheduling parameters, goto 5;
5. reconfiguration of scheduling parameters, goto 1, 3 or 6;
6. manual corrections, goto 3, 5 or 7;
7. end of optimization, save the schedule.

Tabu Search Algorithm

Tabu Search algorithm shown in detail in Algorithm 2 is used to reduce the state space of schedules.

In our implemenentation, $TabuList$ represents the list of forbidden shift exchanges and has three attributes. The indexes i_1 and i_2 represent origin and target row (nurse) of shift exchange. The third index j is day index. Length of $TabuList$, so called $TabuList\ tenure$, was set to 8.

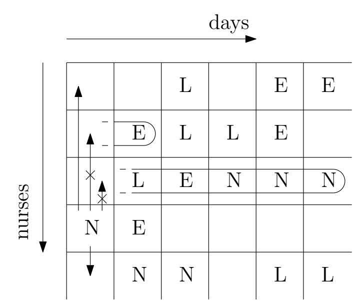


Figure 3: The *candidate* search, non-permissible shift exchanges.

Table 1: NSWA experiments.

n	m	#RE	#RL	#RN	#H	#R	hc1	hc2	FC	w(sc1)	w(sc2)	w(sc3)	w(sc4)	w(sc5)	$t_s[s]$
28	12	4	3	2	0	0	1	1	0	0	0	0	0	0	1.336
28	12	4	3	2	5	0	1	1	0	0	0	0	0	0	2.396
28	12	4	3	2	5	1	1	1	0	0	0	0	0	0	1.038
28	12	4	3	2	5	0	1	1	0	100	100	0	0	0	2.860
28	12	4	3	2	5	0	1	1	0	100	100	100	100	100	4.342
28	12	4	3	2	5	0	1	1	'NNLL'	100	100	100	100	100	6.460
28	12	4	3	2	5	0	1	1	'NNLL' 'LNLE'	100	100	100	100	100	7.327
28	20	7	5	3	5	0	1	1	'NNLL' 'LNLE'	100	100	100	100	100	88.588
28	20	3	2	2	5	0	1	1	'NNLL' 'LNLE'	100	100	100	100	100	35.607

Let the $candidate$ be a possible shift exchange in one day that satisfies hard constraints (see Figure 3 – two candidates are forbidden due to hard constraints hc1b, hc2). Let x_{cand} be the schedule x within updated $candidate$ shift exchange and $Z(x_{cand})$ be the value of objective function of this schedule.

Algorithm 2 – Tabu Search Algorithm

1. compute $Z(x_{init})$;
2. $x := x_{init}$; $x_{next} := x_{init}$;
   $Z(x) := Z(x_{init})$; $Z(x_{next}) := Z(x_{init})$;
3. **while** $((Z(x) > 0)$ & $(\exists$ not forbidden $f_j(x)))$ (Figure 4)
   - choose $\max(w_j f_j(x))$, $j \in$ not forbidden constraints;
   - **for** $\forall candidate$
     - **if** $(candidate \notin TabuList)$
       - compute $Z(x_{cand})$;
       - **if** $(Z(x_{cand}) < Z(x_{next}))$
         - $x_{next} := x_{cand}$;
         - $Z(x_{next}) := Z(x_{cand})$;
       - **endif**
     - **endif**
   - **endfor**
   - **if** $(Z(x_{next}) < Z(x))$
     - $x := x_{next}$;
     - $Z(x) := Z(x_{next})$;
     - add opposite exchange to the top of $TabuList$;
     - clear all forbidden constraints (Figure 4, step 2);
   - **else**
     - add an empty record to the top of $TabuList$;
     - forbid the chosen constraint (Figure 4, steps 1, 3, 4, 5, 6);
   - **endif**

   endwhile
4. **return** x, $Z(x)$.

Let $next$ be the best $candidate$ (with respect to the objective function) at each optimization step. When we have gone through all possible $candidates$, we compare values of $Z(x)$ and $Z(x_{next})$ and choose the better one for the next step of optimization. The idea of soft constraint choice and algorithm termination is demonstrated in Figure 4 for the case of four soft constraints.

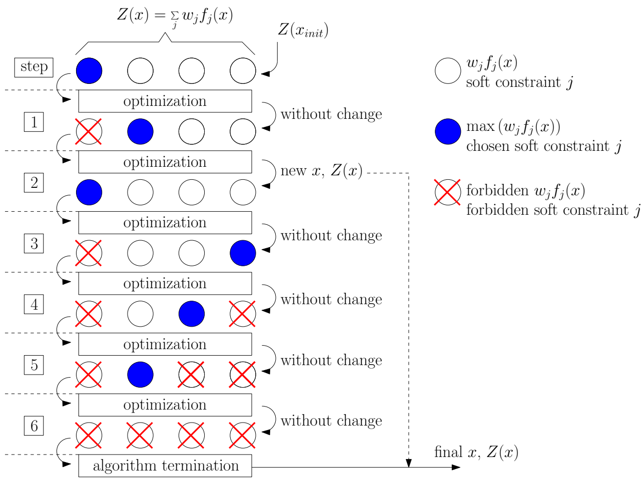


Figure 4: Choice of soft constraints for the next step of optimization and algorithm termination.

Experiments

We used our NSWA called iMEDICA[1] for the instances, that are presented Table 1. Columns from n to w(sc5) are input parameters (n stand for the number of nurses and m for number of days in schedule, other columns are hard and soft constraints). Column FC shows considered forbidden shift combinations (e.g. 'LNLE' - late, night, late and early shift). Output parameter t_s is scheduling time in seconds including steps 1-4 from Algorithm 1 and web communication. The instances were computed on server Intel Pentium 3.4 GHz@4 GB DDR.

In order to evaluate our NSWA, we implemented optimal solution via ILP for simplified two-shift type (day and night) mathematical model (Azaiez & Sharif 2005). We used free solver GLPK[2]. Scheduling times for instances with $n \sim 10$ nurses and $m = 28$ days were hundreds of seconds (more results are in (Baumelt 2007)).

[1]iMEDICA, `http://imedica.merica.cz/`, the product of Merica s. r. o.

[2]GPLK, `http://www.gnu.org/software/glpk/`

Conclusions

In this paper we briefly presented our Nurse Scheduling Web Application. We have got the feedback from several hospitals in the Czech Republic. In cooperation with these hospitals we are working on the improvement of the mathematical model and application interface.

Acknowledgements

This work was supported by the Ministry of Education of the Czech Republic under Project 2C06017.

References

Aickelin, U. 1999. *Genetic Algorithms for Multiple-Choice Optimisation Algorithms*. Ph.d. diss., European Business Management School University of Swansea.

Azaiez, M. N., and Sharif, S. S. 2005. A 0-1 goal programming nodel for nurse scheduling. *Computers & Operations Research* 32.

Baumelt, Z. 2007. *Hospital Nurse Scheduling*. Master thesis, Department of Control Engineering, Faculty of Electrical Engineering, Czech Technical University in Prague.

Berghe, G. V. 2002. *An Advanced Model and Novel Meta-Heuristic Solution Methods to Personnel Scheduling in Healthcare*. Ph.d. diss., University of Gent.

Burke, E. K.; de Causmaecker, P.; Berghe, G. V.; and van Landeghem, H. 2004. The state of the art of nurse rostering. *Journal of Scheduling* 7:441–499.

Cheang, B.; Li, B.; Lim, A.; and Rodrigues, B. 2002. Nurse rostering problems – a bibliographic survey. *European Journal of Operations Research*.

Chen, J., and Yeung, T. 1993. Hybrid expert system approach to nurse scheduling. *Computers in Nursing*.

Eiselt, H. A., and Sandblom, C. L. 2000. *Integer Programming and Netwotk Models*. Springer-Verlag Berlin and Heidelberg, 1st edition.

Hung, R. 1995. Hospital nurse scheduling. *Journal of Nursing Administration* 25.

Okada, M. 1992. An approach to the generalised nurse scheduling problem – generation of a declarative program to represent institution-speciffic knowledge. *Computers and Biomedical Research* 25.