# A Hybrid Model of Integer Programming and Variable Neighbourhood Search for Highly-Constrained Nurse Rostering Problems

Edmund K. Burke, Jingpeng Li* and Rong Qu
{ekb, jpl, rxq}@cs.nott.ac.uk
School of Computer Science and Information Technology
The University of Nottingham
Nottingham, NG8 1BB, United Kingdom

* Corresponding author

**Abstract**: This paper presents a hybrid multi-objective model that combines Integer Programming (IP) and Variable Neighbourhood Search (VNS) to deal with highly constrained nurse rostering problems in modern hospital environments. An IP is first used to solve the subproblem which includes the full set of hard constraints and a subset of soft constrains. A basic VNS then follows as a postprocessing procedure to further improve the IP's resulting solutions, and the satisfaction of the excluded constraints from the preceding IP model is the VNS's major concern in its search. Very promising results are reported compared with a commercial genetic algorithm and a hybrid VNS approach on real instances arising in a Dutch hospital. The comparison results demonstrate that our hybrid approach does combine advantages of both the IP and the VNS to beat pure metaheuristics in solving this type of problems. We also believe that the proposed approach can be applied to other resource allocation problems with a large number of constraints.

**Keywords**: Nurse Rostering, Multi-Objective Optimization, Integer Programming, Variable Neighbourhood Search

## 1 Introduction

Employee scheduling has been extensively studied by operations researchers and computer scientists for more than 40 years (see survey papers in [8, 11, 28]). It can be thought of as the problem of assigning employees to shifts or duties over a scheduling period so that certain constraints (organizational and personal) are satisfied. In the field of hospital nurse rostering, this is particularly challenging due to the presence of different nurse requirements on different days and shifts. Unlike most other organizations, hospitals and medical institutes work around the clock, and thus irregular shift work has an effect on the nurses' well being and job satisfaction [36]. The development of robust and powerful nurse rostering systems that can handle a wide range of requirements and constraints would provide significant benefits for hospital administrators and staff.

Nurse rostering is a type of resource allocation problem, in which the workload needs to be assigned to nurses periodically, taking into account a number of constraints and requirements. Hard constraints are those that must be satisfied in order to have a feasible schedule. They are often generated by physical resource restrictions and legislation. When requirements are desirable but not obligatory they are referred to as soft constraints. Such constraints are often used to evaluate the quality of feasible schedules. In nurse rostering, there are a large number of variations on legal regulations and individual preferences, depending on different countries and institutions. Typical issues concern coverage demand, day-off requirements, weekend-off requirements, minimum and maximum workforce, etc [18].

Most nurse rostering problems in the real world are NP-hard [30] and were regarded as being more complex than the travelling salesman problem by [37]. Over the years, a wide variety of methodologies and models have been developed to deal with them. The survey papers [15, 21, 39] give an overview of the area. The available techniques can roughly be classified into two main categories: exact algorithms and (meta)heuristics. Mathematical programming is the traditional exact method [6, 7, 9, 41], which guarantees to find an optimal solution and to prove its optimality for every instance of a problem. However, computational difficulties exist with this approach due to the enormous size of the search spaces that are generated. To reduce complexity, some researchers have restricted the problem dimensions and developed simplified models. However, this leads to solutions that are not applicable to real hospital situations.

The above observations have led to attempts to solve the real problems by investigating heuristics. In this case, the guarantee of finding optimal solutions is sacrificed for the sake of getting good solutions within a reasonable amount of time. Early heuristic approaches [10, 40] were investigated with some success and metaheuristics have attracted significant attention since the 1990's. Genetic algorithms form an important class of metaheuristics that have been extensively applied in nurse rostering [2, 3, 26, 31]. A number of attempts have also been made by using other metaheuristics, such as simulated annealing [13], tabu search [14], variable neighbourhood search [17, 18], memetic algorithms [16] and estimation of distribution algorithms [4].

However, the major drawback of these metaheuristics is they can not provably produce optimal solutions nor can they provably reduce the search space. Also, they usually do not have well defined stopping criteria. Moreover, as most nurse rostering problems are highly constrained problems which mean the feasible regions of their solution space are disconnected (i.e. separated by the infeasible area), metaheuristics generally have difficulty in dealing with the situation. Considering the advantages and disadvantages for both categories of approaches, this research is therefore looking at new attempts for an appropriate integration. The long-term aim of this research is to investigate efficient ways of decomposing real world personnel scheduling problems into tractable subproblems, without losing much optimality.

In the field of nurse rostering, some decomposition techniques have been investigated over recent years. Aickelin and Dowsland [3] developed a genetic algorithm with an indirect representation. Different heuristic decoders (i.e. decomposers) were employed to construct the schedule, taking care of coverage and nurses' preferences from different aspects. Ikegami and Niwa [3] grouped the constraints into shift constraints and nurse constraints, based on which the problems were decomposed into subproblems and solved repeatedly by tabu search. Brucker et al [12] implemented decomposition by cyclically assigning predefined blocks of shifts to groups of nurses. The rest of the shifts were then assigned by hand and the resulting schedule was improved by local search.

In this paper, we present a new decomposition technique by combining Integer Programming (IP) and Variable Neighbourhood Search (VNS) to deal with constraints and requirements. More details about these techniques can be seen in [19]. The IP is first used to solve a subproblem including all hard constraints and a subset of soft constraints. When determining if a soft constraint should be included in the subset, we give more priority to the constraints that have the following characteristics: low complexity (i.e. the number of variables and constraints it may add in the IP model), high importance (i.e. the degree to which the constraint is considered to be desirable by the hospital), or a trade-off between complexity and importance. The VNS is then used as a postprocessing procedure to make the improvement, and the satisfaction of the constraints that are not included in the subset would be the major concern in designing the VNS's neighbourhood structures. Note that in the first IP phase, it is not necessary to solve the subproblem to optimality because, under most circumstances, this process would still take an extremely long time to finish. We can obtain an intermediate solution by setting up a stopping condition (e.g. maximum runtime or acceptable solution quality) to the IP.

The rest of the paper is organized as follows. We first describe the nurse rostering problem to be addressed and formulate its full multi-objective IP model by taking into account all the constraints. We then formulate an alternative heuristic model and propose a basic VNS approach to deal with it. Later, we carry out the experiments on twelve real instances arising in a Dutch hospital and demonstrate how the proposed IP and the VNS are not capable of solving the problem in their own right, and how the suggested combination can well achieve the optimization. Finally, we give concluding remarks and possible future research directions.

## 2 Problem Description

The nurse rostering problem we are tackling is provided by ORTEC, an international consultancy company specializing in planning, optimization and decision support solutions. The problem is based on the situation of intensive care units in a Dutch hospital, which involves assigning four types of shifts (i.e. shifts of *early*, *day*, *late* and *night*) within a scheduling period of 5 weeks to 16 nurses of different working contracts in a ward. The problem has the following key characteristics:

1. Dutch national legislation and the collective labour agreements in force in hospitals must be complied with. They are translated to a shorter period to be meaningful for a situation without (much) history;
2. The nurses' requests are very important and should be met as much as possible;
3. It is not necessary to consider qualifications as all nurses are highly qualified. Nurses still in training are not considered in the original planning and are added by hand afterwards.

In brief, the problem has the following hard constraints which must be met under any circumstances, otherwise the schedule is considered to be infeasible and unacceptable:

- HC1: Daily coverage requirement of each shift type
- HC2: For each day, a nurse may not start more than one shift
- HC3: Maximum number of total working days during the scheduling period
- HC4: Maximum number of on-duty weekends during the scheduling period
- HC5: Maximum number of *night* shifts during the scheduling period
- HC6: No stand-alone *night* shift (i.e. no *night* shift between two non-*night* shifts)
- HC7: Minimum two free days after a series of *night* shifts
- HC8: Maximum number of consecutive *night* shifts
- HC9: Maximum number of consecutive working days
- HC10: No *late* shifts for one particular nurse

In addition, the problem has the following soft constraints which should be satisfied as much as possible although in real world circumstances it is usually unavoidable to violate some of them:

- SC1: Complete weekends (i.e. either no shifts or two shifts in weekends)
- SC2: Avoiding any stand-alone shift (i.e. a single day between two days off)
- SC3: Minimum number of free days after a series of shifts
- SC4: Maximum / minimum number of consecutive assignments of *early* and *late* shifts

- SC5: Maximum / minimum number of weekly working days
- SC6: Maximum number of consecutive working days for part-time nurses
- SC7: Avoiding certain shift type successions (e.g. a *day* shift followed by an *early* one, etc)

## 3 A Multi-objective Mathematical Model

To specify the above problem, slack and surplus variables can be introduced into the soft constraints, and the objectives are to minimize the values of individual variables. We formulate the entire problem associated with a 5-week scheduling period as the following IP model, which can be relatively altered to adapt to other problems with different constraints.

Parameters:
$I$ = Set of nurses available;
$I_t \mid t \in \{1,2,3\} =$ Subset of nurses that work 20, 32, 36 hours per week respectively, $I = I_1 + I_2 + I_3$;
$J$ = Set of indices of the last day of each week within the scheduling period = {7, 14, 21, 28, 35};
$K$ = Set of shift types = {1(*early*), 2(*day*), 3(*late*), 4(*night*)};
$K'$ = Set of undesirable shift type successions = {(2,1), (3,1), (3,2), (1,4)};
$d_{jk}$ = Coverage requirement of shift type $k$ on day $j$, $j \in \{1,...,7|J|\}$;
$m_i$ = Maximum number of working days for nurse $i$ within the scheduling period;
$n_1$ = Maximum number of consecutive *night* shifts within the scheduling period;
$n_2$ = Maximum number of consecutive working days within the scheduling period;
$c_k$ = Desirable upper bound of consecutive assignments of shift type $k$;
$g_t$ = Desirable upper bound of weekly working days for the $t$-th subset of nurses;
$h_t$ = Desirable lower bound of weekly working days for the $t$-th subset of nurses.

Decision variable $x_{ijk}$ is 1 if nurse $i$ is assigned shift type $k$ for day $j$, 0 otherwise, defined as:

$$x_{ijk} = 0 \text{ or } 1, \quad \forall i \in I, j \in \{1,...,7|J|\}, k \in K \tag{1}$$

Slack/surplus variables are the positive/negative deviations from individual goals, defined as:

$$s_{ij}^1 \geq 0, \; s_{ij}^2 \geq 0, \quad \forall i \in I, j \in J \tag{2}$$

$$s_{ij}^3 \geq 0, s_{ij}^4 \geq 0, \quad \forall i \in I, j \in \{2,...,7|J|-1\} \tag{3}$$

$$s_{irk}^5 \geq 0, \quad \forall i \in I, r \in \{1,...,7|J|-3\}, k \in \{1,3\} \tag{4}$$

$$s_{ijk}^6 \geq 0, \quad \forall i \in I, j \in \{2,...,7|J|-1\}, k \in \{1,3\} \tag{5}$$

$$s_{tiw}^7 \geq 0, s_{tiw}^8 \geq 0, \quad \forall t \in \{1,2,3\}, i \in I_t, w \in \{1,...,|J|\} \tag{6}$$

$$s_{ir}^9 \geq 0, \quad \forall i \in I_1, r \in \{1,...,7|J|-3\} \tag{7}$$

$$s_{ijk'}^{10} \geq 0, \quad \forall i \in I, j \in \{1,...,7|J|-1\}, k' = (k_1, k_2) \in K' \tag{8}$$

Target function:

$$\text{Min } F(x) = [f_1(x)^T, f_2(x)^T, f_3(x)^T, f_4(x)^T, f_5(x)^T, f_6(x)^T, f_7(x)^T]^T, \tag{9}$$

where the vector functions $f_i(x), i \in \{1,...,7\}$, are defined as

$$f_1(x) = [s_{ij}^1, s_{ij}^2]^T, \quad \forall i \in I, j \in J \tag{10}$$

$$f_2(x) = [s_{ij}^3, s_{ij}^4]^T, \quad \forall i \in I, j \in \{2,...,7|J|-1\} \tag{11}$$

$$f_3(x) = [s_{irk}^5]^T, \quad \forall i \in I, r \in \{1,...,7|J|-3\}, k \in \{1,3\} \tag{12}$$

$$f_4(x) = [s_{ijk}^6]^T, \quad \forall i \in I,\ j \in \{2,...,7|J|-1\}, k \in \{1,3\} \tag{13}$$

$$f_5(x) = [s_{tiw}^7, s_{tiw}^8]^T, \quad \forall t \in \{1,2,3\}, i \in I_t, w \in \{1,...,|J|\} \tag{14}$$

$$f_6(x) = [s_{ir}^9]^T, \quad \forall i \in I_1, r \in \{1,...,7|J|-3\} \tag{15}$$

$$f_7(x) = [s_{ijk'}^{10}]^T, \quad \forall i \in I,\ j \in \{1,...,7|J|-1\}, k' = (k_1, k_2) \in K' \tag{16}$$

Subject to:

HC1
$$\sum_{i \in I} x_{ijk} = d_{jk}, \quad \forall j \in \{1,...,7|J|\}, k \in K \tag{17}$$

HC2
$$\sum_{k \in K} x_{ijk} \le 1, \quad \forall i \in I,\ j \in \{1,...,7|J|\} \tag{18}$$

HC3
$$\sum_{j=1}^{7|J|} \sum_{k \in K} x_{ijk} \le m_i, \quad \forall i \in I \tag{19}$$

HC4
$$\sum_{j \in J} \sum_{k \in K} x_{ijk} \le 3, \quad \forall i \in I \tag{20}$$

HC5
$$\sum_{j=1}^{7|J|} x_{ij4} \le 3, \quad \forall i \in I \tag{21}$$

HC6
$$x_{i(j-1)4} - x_{ij4} + x_{i(j+1)4} \ge 0, \quad \forall i \in I,\ j \in \{2,...,7|J|-1\} \tag{22}$$

HC7 is equivalent to the following three sub-constraints which rule out the sequences of ‘N01’, ‘N10’ and ‘N11’ respectively, where ‘N’ denotes a *night* shift, ‘0’ a off-duty day and ‘1’ a on-duty day:

$$x_{i(j-1)4} - \sum_{k=1}^{3} x_{ijk} + \sum_{k=1}^{3} x_{i(j+1)k} \le 1, \quad \forall i \in I,\ j \in \{2,...,7|J|-1\} \tag{23}$$

$$x_{i(j-1)4} + \sum_{k=1}^{3} x_{ijk} - \sum_{k=1}^{3} x_{i(j+1)k} \le 1, \quad \forall i \in I,\ j \in \{2,...,7|J|-1\} \tag{24}$$

$$x_{i(j-1)4} + \sum_{k=1}^{3} x_{ijk} + \sum_{k=1}^{3} x_{i(j+1)k} \le 2, \quad \forall i \in I,\ j \in \{2,...,7|J|-1\} \tag{25}$$

HC8
$$\sum_{j=r}^{r+n_1} x_{ij4} \le n_1, \quad \forall i \in I, r \in \{1,...,7|J|-n_1\} \tag{26}$$

HC9
$$\sum_{j=r}^{r+n_2} \sum_{k \in K} x_{ijk} \le n_2, \quad \forall i \in I, r \in \{1,...,7|J|-n_2\} \tag{27}$$

HC10
$$x_{16(j)3} = 0, \quad \forall j \in \{1,...,7|J|\} \tag{28}$$

SC1
$$\sum_{k \in K} [x_{i(j-1)k} - x_{ijk}] + s_{ij}^1 - s_{ij}^2 = 0, \quad \forall i \in I,\ j \in J \tag{29}$$

SC2
$$\sum_{k \in K} [x_{i(j-1)k} - x_{ijk} + x_{i(j+1)k}] + s_{ij}^3 \ge 0, \quad \forall i \in I,\ j \in \{2,...,7|J|-1\} \tag{30}$$

SC3
$$\sum_{k \in K} [x_{i(j-1)k} - x_{ijk} + x_{i(j+1)k}] - s_{ij}^4 \le 1, \quad \forall i \in I,\ j \in \{2,...,7|J|-1\} \tag{31}$$

SC4
$$\sum_{j=r}^{r+3} x_{ijk} - s_{irk}^5 \le c_k, \quad \forall i \in I, r \in \{1,...,7|J|-3\}, k \in \{1,3\} \tag{32}$$

$$x_{i(j-1)k} - x_{ijk} + x_{i(j+1)k} + s_{ijk}^6 \ge 0, \quad \forall i \in I,\ j \in \{2,...,7|J|-1\}, k \in \{1,3\} \tag{33}$$

SC5 $$\sum_{j=7w-6}^{7w} \sum_{k \in K} x_{ijk} - s_{tiw}^{7} \le g_t, \quad \forall t \in \{1,2,3\}, i \in I_t, w \in \{1,...,|J|\} \tag{34}$$

$$\sum_{j=7w-6}^{7w} \sum_{k \in K} x_{ijk} + s_{tiw}^{8} \ge h_t, \quad \forall t \in \{1,2,3\}, i \in I_t, w \in \{1,...,|J|\} \tag{35}$$

SC6 $$\sum_{j=r}^{r+3} \sum_{k \in K} x_{ijk} - s_{ir}^{9} \le 3, \quad \forall i \in I_1, r \in \{1,...,7|J|-3\} \tag{36}$$

SC7 $$x_{ijk_1} + x_{i(j+1)k_2} - s_{ijk'}^{10} \le 2, \quad \forall i \in I, j \in \{1,...,7|J|-1\}, k' = (k_1, k_2) \in K' \tag{37}$$

The number of objectives in the above model could be reduced by summing up the slack/surplus variables that are associated with soft constraints of the same type and also have the same domains. Hence, the target function $F(x)$ in (9) can be replaced by $G(x)$ which only consists of the following seven sub-functions (i.e. objectives)

$$G(x) = [g_1(x), g_2(x), g_3(x), g_4(x), g_5(x), g_6(x), g_7(x), g_8(x)], \tag{38}$$

where

$$g_1(x) = \sum_{i \in I} \sum_{j \in J} (s_{ij}^{1} + s_{ij}^{2}), \tag{39}$$

$$g_2(x) = \sum_{i \in I} \sum_{j=2}^{7|J|-1} s_{ij}^{3}, \tag{40}$$

$$g_3(x) = \sum_{i \in I} \sum_{j=2}^{7|J|-1} s_{ij}^{4}, \tag{41}$$

$$g_4(x) = \sum_{i \in I} \sum_{r=1}^{7|J|-3} \sum_{k \in \{1,3\}} s_{ijk}^{5}, \tag{42}$$

$$g_5(x) = \sum_{i \in I} \sum_{j=2}^{7|J|-1} \sum_{k' \in \{1,3\}} s_{ijk'}^{6}, \tag{43}$$

$$g_6(x) = \sum_{t=1}^{3} \sum_{i \in I_t} \sum_{w=1}^{|J|} (s_{tiw}^{7} + s_{tiw}^{8}), \tag{44}$$

$$g_7(x) = \sum_{i \in I_1} \sum_{r=1}^{7|J|-3} s_{ir}^{9}, \tag{45}$$

$$g_8(x) = \sum_{i \in I} \sum_{j=1}^{7|J|-1} \sum_{k' \in K'} s_{ijk'}^{10}. \tag{46}$$

This is a multi-objective problem, and its goal is to find or to approximate the set of Pareto-optimal solutions. The traditional approach for this type of problem is the weighted-sum approach which combines the multiple objectives into one scalar objective [20, 27]. Over the recent years, researchers have proposed a number of evolutionary multi-objective optimization approaches, such as the neighbourhood constraint GA in [34], the Pareto envelope based selection algorithms in [23], the strength Pareto-EA in [42], the non-dominant sorting GA in [25] and the differential evolution based EMO in [5]. A wider range of other evolutionary approaches can be found in the survey paper [22].

## 4 A Multi-objective Heuristic Model and Variable Neighbourhood Search (VNS)

This section presents an alternative heuristic model and a VNS metaheuristic to deal with the problem.

### 4.1 Model Formulation

Compared with the IP model in Section 3, the multi-objective heuristic model satisfies the hard constraints (i.e. HC1 – HC10) by heuristic search method and formulates all the soft constraints (i.e. SC1 – SC7) as non-linear objective functions which measure derivations from the most desirable goals defined by these soft constraints. Hence, the alternative heuristic model has the advantage of including fewer constraints.

More specifically, the above problem can be modelled as

$$\text{Min } \overline{G}(x) = [\overline{g}_1(x), \overline{g}_2(x), \overline{g}_3(x), \overline{g}_4(x), \overline{g}_5(x), \overline{g}_6(x), \overline{g}_7(x), \overline{g}_8(x)]\,. \tag{47}$$

Subject to constraints (1), (17) – (28) in Section 3, where

$$\overline{g}_1(x) = \sum_{i\in I}\sum_{j\in J}\left|\sum_{k\in K}[x_{i(j-1)k} - x_{ijk}]\right| \tag{48}$$

$$\overline{g}_2(x) = \sum_{i\in I}\sum_{j=2}^{7|J|-1}\max\left\{0, \sum_{k\in K}[-x_{i(j-1)k} + x_{ijk} - x_{i(j+1)k}]\right\} \tag{49}$$

$$\overline{g}_3(x) = \sum_{i\in I}\sum_{j=2}^{7|J|-1}\max\left\{0, \sum_{k\in K}[x_{i(j-1)k} - x_{ijk} + x_{i(j+1)k}] - 1\right\} \tag{50}$$

$$\overline{g}_4(x) = \sum_{i\in I}\sum_{r=1}^{7|J|-3}\sum_{k\in\{1,3\}}\max\left\{0, \sum_{j=r}^{r+3} x_{ijk} - c_k\right\} \tag{51}$$

$$\overline{g}_5(x) = \sum_{i\in I}\sum_{j=2}^{7|J|-1}\sum_{k'\in\{1,3\}}\max\left\{0, -x_{i(j-1)k} + x_{ijk} - x_{i(j+1)k}\right\} \tag{52}$$

$$\overline{g}_6(x) = \sum_{t=1}^{3}\sum_{i\in I_t}\sum_{w=1}^{|J|}\left[\max\left\{0, \sum_{j=7w-6}^{7w}\sum_{k\in K} x_{ijk} - g_t\right\} + \max\left\{0, h_t - \sum_{j=7w-6}^{7w}\sum_{k\in K} x_{ijk}\right\}\right] \tag{53}$$

$$\overline{g}_7(x) = \sum_{i\in I_1}\sum_{r=1}^{7|J|-3}\max\left\{0, \sum_{j=r}^{r+3}\sum_{k\in K} x_{ijk} - 3\right\} \tag{54}$$

$$\overline{g}_8(x) = \sum_{i\in I}\sum_{j=1}^{7|J|-1}\sum_{k'\in K'}\max\left\{0, x_{ijk} + x_{i(j+1)k_2} - 2\right\} \tag{55}$$

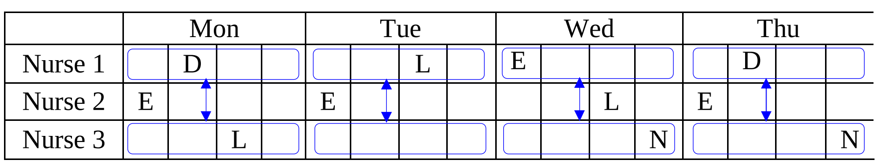
### 4.2 The VNS Approach

As the major purpose of this paper is to investigate the availability of combining exact algorithms and metaheuristics in handling a variety of constraints and requirements from nurse rostering, we believe that the metaheuristics applied should be as simple as possible, i.e. without any additional local search or combination with the other (meta)heuristics. Also, to effectively achieve the task of decomposition, we design the metaheuristics so that they can focus on the search from particular aspects of the problem, i.e. the satisfaction of the constraints that have not been taken into account by the exact algorithm. The motivation for investigating a VNS approach is based on these two considerations.

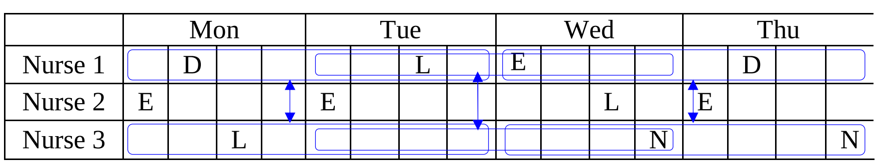
VNS is a relatively recent metaheuristic based on the simple idea of changing neighbourhood within a local search to identify better local optima [35]. It has been applied to a wide variety of NP-hard problems such as the travelling salesman problem, the resource-allocation problem, the clustering problem [32], the linear ordering problem [29], vehicle routing [24], nurse rostering [18] and university course timetabling [1]. An introduction can be seen in [19].

VNS is able to drive the search towards certain desirable objectives by defining the appropriate neighbourhood structures associated with these objectives, although each resulting solution still needs to be evaluated by all the objectives in target function (47). As the VNS here is mainly used to make refinement on the IP's resulting solution which has taken most constraints into account, the VNS's neighbourhood structures should not be too complicated. In this paper we apply the neighbourhoods of swapping groups of consecutive shifts which are inspired by human scheduling processes of re-allocating sections of schedules. All possible swaps have been considered in these neighbourhoods, that is, shifts in a period from one day to the whole scheduling period can be switched between any set of two nurses in the schedule. To guarantee the satisfaction of the hard constraint HC1 of daily shift demands, swaps will only be allowed vertically. If any of the swaps results in an infeasible solution, this swap will be deemed as invalid.

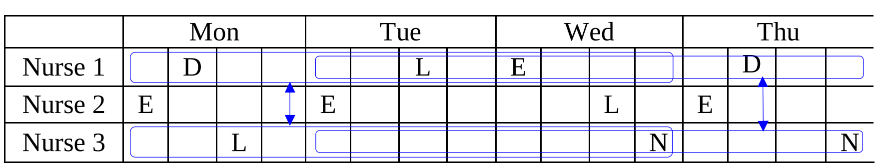
Figure 1 illustrates the moves that are allowed in the proposed neighbourhoods. For clarity, a small part of the scheduling period (i.e. a 4-day period) is shown and each day a nurse can work no more than one of the four shift types: *Early* (E), *Day* (D), *Late* (L) and *Night* (N). An arrow denotes a possible move in the neighbourhood.



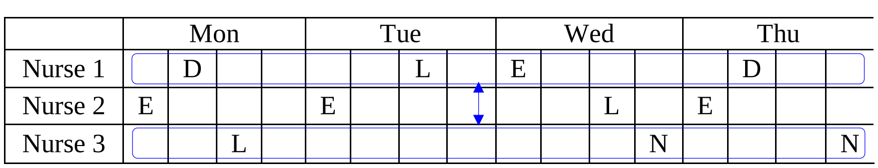

Neighbourhood $N_k$, $k = 1$




Neighbourhood $N_k$, $k = 2$




Neighbourhood $N_k$, $k = 3$




Neighbourhood $N_k$, $k = 4$

Figure 1. Possible moves in neighbourhoods $N_k$ between the schedules of nurse 1 and nurse 3

Given an initial solution, our VNS was implemented in a greediest or steepest decent manner. It starts to search from its first neighbourhood (i.e. single swap) between the schedules of any pair of two nurses. If the best solution contained in the current neighbourhood is no better than the previous best, the algorithm goes to its next neighbourhood and searches there. Otherwise, the best solution is updated and the algorithm comes back to search from the first neighbourhood of this best solution. The algorithm stops if no improvement has been achieved after finishing the search in its last neighbourhood. Figure 2 gives a schematic overview of the basic VNS.

```
The VNS ( )
{
   Define a set of neighbourhood structures Nk, k = 1, …, kmax;
   Create an initial solution x;
   Set k ← 1;
   While k ≤ kmax {
      Explore the neighbourhood Nk of x;
      Find the best solution x′ in Nk;
      If x′ < x {
         Set x ← x′;
         Set k ← 1;
      }
      Else
         Set k ← k + 1;
   }
   Return the best solution found;
}
```

Figure 2. Pseudo-code of the basic VNS

Note that the proposed neighbourhoods may be applied in different orders, which would affect the search results significantly. According to experiments regarding the neighbourhood order in nurse rostering [18], the best way is to develop algorithms that explore neighbourhoods with increasing size. Hence, our approach starts over from the first (finest) neighbourhood when the current neighbourhood contains a better solution.

Defined by its neighbourhood structures, the proposed basic VNS has a time complexity of $O(m^2 \times n^2)$, where $m$ is the number of nurses and $n$ is the length of the scheduling period. The benefit of applying this approach is that the number of neighbouring solutions is not particularly large, and thus an exhaustive search in the proposed neighbourhoods for solution improvement is not a very time-consuming process.

## 5 Computational Results

Whilst the hard constraints of our problem are requirements that must be met under any circumstance, the objectives for both of our IP model and heuristic model are to satisfy the soft constraints as much as possible. Regarding the degree of desirableness for the soft constraints listed in Section 2, we have the following priority ordering: [SC1, SC2] $\succ$ SC3 $\succ$ [SC4, SC5, SC6] $\succ$ [SC7], where '$\succ$' denotes "*be more preferred than*". This ordering is determined in close consultation with the hospital.

Accordingly, regarding the priority ordering of the objective functions, in the proposed IP model we have $[g_1(x), g_2(x)] \succ [g_3(x)] \succ [g_4(x), g_5(x), g_6(x), g_7(x)] \succ g_8(x)$, and in the heuristic model we have $[\bar{g}_1(x), \bar{g}_2(x)] \succ [\bar{g}_3(x)] \succ [\bar{g}_4(x), \bar{g}_5(x), \bar{g}_6(x), \bar{g}_7(x)] \succ \bar{g}_8(x)$. Hence, we can solve a single-objective problem for each multi-objective model, in which the constraint set remains unchanged and the objective functions (37) and (46) change to

$$G(x) = \sum_{i=1}^{8} \lambda_i g_i(x) \tag{56}$$

and

$$\overline{G}(x) = \sum_{i=1}^{8} w_i \bar{g}_i(x), \tag{57}$$

respectively. Parameters $\lambda_i$ and $w_i$ are the weights of objectives which are determined by taking their priority ordering into account.

In fact, problems exist with the above weighted sum functions if the Pareto surfaces are non-convex. Moreover, a proper determination of weight distributions would be difficult. However, as the results of previous approaches on this problem are all obtained by using the same type of weighted-sum objective functions, in this paper we continue to use them for benchmark comparisons.

| Data | Hybrid GA (after 1 hr) | Hybrid VNS (after 1 hr) |
|---|---|---|
| JAN | 775 | 735 |
| FEB | 1791 | 1866 |
| MAR | 2030 | 2010 |
| APR | 612 | 457 |
| MAY | 2296 | 2161 |
| JUN | 9466 | 9291 |
| JUL | 781 | 481 |
| AUG | 4850 | 4880 |
| SEP | 615 | 647 |
| OCT | 736 | 665 |
| NOV | 2126 | 2030 |
| DEC | 625 | 520 |
| **AVE.** | **2225** | **2145** |

Table 1. Results of the hybrid GA and the hybrid VNS after 1-hour runtime

Table 1 shows the results of two other approaches on twelve real-world data instances of the problem. The first approach is a hybrid GA which has been integrated into ORTEC's Harmony™ system [38]. In this approach, a local search is carried out after each generation of the GA to improve the individual solutions to their local optima. The moves taken in the local search include moving a shift from one nurse to another and swapping two single shifts assigned to two nurses.

The second approach is a hybrid VNS with heuristic ordering [17], in which the same types of move as in the above hybrid GA are used in its VNS step. Starting from an initial schedule, created by an adaptive ordering technique, this approach sequentially runs the steps of VNS, feasibility correction, schedule disruption and schedule reconstruction in a loop until a stopping condition is reached. Computational results demonstrate that the search can be extended and the solution quality can be significantly improved by the careful combination and repeated use of heuristic ordering, VNS and backtracking.

Both of the above approaches are coded in Delphi 5 and implemented on a Pentium 1.7 GHz PC under Window 2000 operating system. In general, the hybrid VNS outperforms the hybrid GA.

For our approach, the IP part is solved by the latest ILOG CPLEX 10.0, a general purpose mathematical programming solver (http://www.ilog.com accessed 30 July 2006), on a Pentium 2.0 GHz PC under Windows XP. The VNS part is coded in Java 2 and run on the same machine. As the hybrid GA and the hybrids VNS in comparisons all used the same weight distribution of [1000, 1000, 100, 10, 10, 10, 5] for soft constraints SC1-SC7 in their target functions, we set $\lambda_i$ and $w_i$ as [1000, 1000, 1000, 100, 10, 10, 10, 5] in Equation (56) and (57) for consistency.

To demonstrate the availability of our proposed decomposition technique in handling constraints, two groups of experiments are carried out on each of the twelve instances. We first consider the situation without decomposition, under which the entire problem would be solved by including all the constraints in the IP model or by inputting a randomly generated initial solution to the VNS. Table 2 lists the problem size (i.e. the number of variables and the number of constraints) of each full IP model and the results of CPLEX after 2 hours' runtime. For information, Table 2 also gives the lower bound provided by the linear programming relaxation to the problem (denoted as "Relaxed IP"). This keeps the objective function and all the constraints but relaxes the integrality restrictions. These lower bound values are mostly found within one minute. The last column of Table 2 presents the results of our basic VNS giving 1 hour's runtime. These results are obtained by restarting the VNS many times from randomly generated initial solutions, after each run of the VNS terminates in about 30 seconds.

| Data | Problem size | | Relaxed IP | | IP's result | | Our VNS's result |
|---|---|---|---|---|---|---|---|
| | Constraint | Variable | Result | Time (sec) | After 1 hr | After 2 hrs | After 1 hr |
| JAN | 9206 | 7915 | 140 | 75 | 3381 | 3241 | None |
| FEB | 8437 | 7316 | 440 | 16 | None | None | None |
| MAR | 9059 | 7830 | 440 | 19 | 7280 | 5240 | None |
| APR | 8787 | 7616 | 40 | 17 | 8989 | 1518 | None |
| MAY | 9218 | 7935 | 240 | 19 | 7658 | 6618 | None |
| JUN | 8836 | 7641 | 7510 | 19 | 10394 | 10394 | None |
| JUL | 9090 | 7831 | 40 | 74 | 11983 | 6643 | None |
| AUG | 9298 | 8019 | 340 | 6 | 9332 | 7922 | None |
| SEP | 8723 | 7564 | 0 | 17 | 6250 | 1070 | None |
| OCT | 9154 | 7883 | 140 | 75 | 3345 | 4205 | None |
| NOV | 9059 | 7830 | 440 | 18 | 4647 | 4647 | None |
| DEC | 9026 | 7805 | 0 | 18 | 3767 | 947 | None |
| **AVE.** | **8991** | **7765** | **814** | **31** | **7002** | **4767** | **None** |

Table 2. Results of the IP and the VNS without decomposition

The next stage carries out the experiments with decomposition. The whole set of soft constraints (i.e. sets SC1-SC7) is partitioned into subsets which will be satisfied by different approaches (i.e. the IP and the basic VNS). As constraint set SC7 would cause the most computational complexity (by introducing about 2000 variables and 2000 constraints for each data instance) and as it is regarded as less important by the hospital, we exclude it from the IP model and leave it to be satisfied by the next VNS. Regarding the maximum runtime allowed for the proposed approach, as the PC for our current experiments is about 10% faster than the ones for the hybrid GA and the hybrid VNS, we set it be 50 minutes, instead of 1 hour, per data instance for a fair comparison: 49 minutes for the IP and 1 minute for the basic VNS. Table 3 lists the sizes of the reduced IP model, the intermediate solutions produced by the IP and the final solutions improved by our VNS. Note that the method employed for constraint partition in this paper is for illustration purpose only, and we can certainly make other kinds of partition among the constraints.

| Data | Reduced IP's size | | Reduced IP's result | | Our VNS's results | |
|---|---|---|---|---|---|---|
| | Constraint | Variable | After 49 mins | Δ %* | After 1 min | Δ % |
| JAN | 7286 | 5995 | 631 | 14.1 | 460 | 37.4 |
| FEB | 6709 | 5588 | 1822 | -1.7 | 1526 | 14.8 |
| MAR | 7203 | 5974 | 3890 | -94 | 1713 | 14.8 |
| APR | 6931 | 5760 | 1268 | -177 | 391 | 14.4 |
| MAY | 7298 | 6067 | 5348 | -147 | 2090 | 3.3 |
| JUN | 7044 | 5849 | 9126 | 1.8 | 8826 | 5.0 |
| JUL | 7170 | 5911 | 2498 | -419 | 425 | 11.6 |
| AUG | 7362 | 6067 | 4582 | 5.5 | 3488 | 28.1 |
| SEP | 6867 | 5708 | 680 | -11.0 | 330 | 46.3 |
| OCT | 7234 | 5963 | 605 | 9.0 | 445 | 33.1 |
| NOV | 7203 | 5974 | 2605 | -28.0 | 1613 | 20.5 |
| DEC | 7106 | 5859 | 1037 | -99.0 | 405 | 22.1 |
| **AVE.** | **7118** | **5893** | **2841** | **-33.0** | **1809** | **15.2** |

Table 3. Results of the proposed decomposition approach
*: 'Δ %' denotes the relative percentage deviation over the previous best solution

According to the results in the above tables, we can see the full IP model for the entire problem cannot produce good solutions if there is a realistic runtime restriction. In addition, the basic VNS alone is not applicable at all as it can produce no feasible solutions for all data instances. The VNS's results are unsurprising due to the highly constrained nature of the problem which makes a single heuristic (i.e. without the use of strong domain knowledge and/or the hybridization with other heuristic searches) extremely difficult to find and maintain its feasible solution. However, the preliminary results by the combination of IP and VNS are very promising. Compared with the commercial hybrid GA in [38] and the complex hybrid VNS in [17], our proposed hybrid approach outperforms them on all instances and significantly improves their best solutions by 15.2% on average.

The behaviours of each run of the decomposed IP and the basic VNS are as expected: while the IP is able to find several integer solutions via big jumps in the solution space within an acceptable time, the VNS however is particularly good at achieving local refinement quickly within the defined neighbourhoods. Figure 3 and Figure 4 respectively depict the improvement of the solution cost versus the runtime (in seconds) for the IP and the VNS on the APR instance, in which the IP finds 12 integer solutions within 45 minutes, and the VNS makes 53 improvements within half a minute from the best solution of the IP. Although the actual values may differ among various instances, the characteristic shapes of the curves are similar.

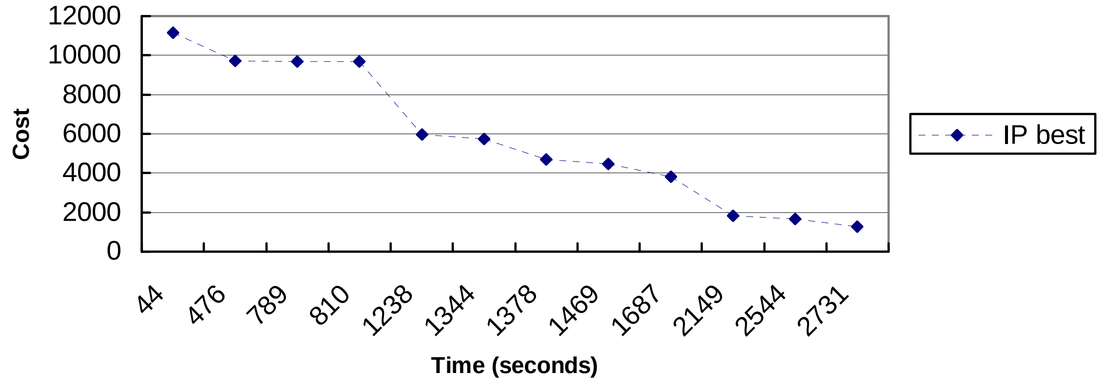



Figure 3: Sample run of the decomposed IP (for instance APR)

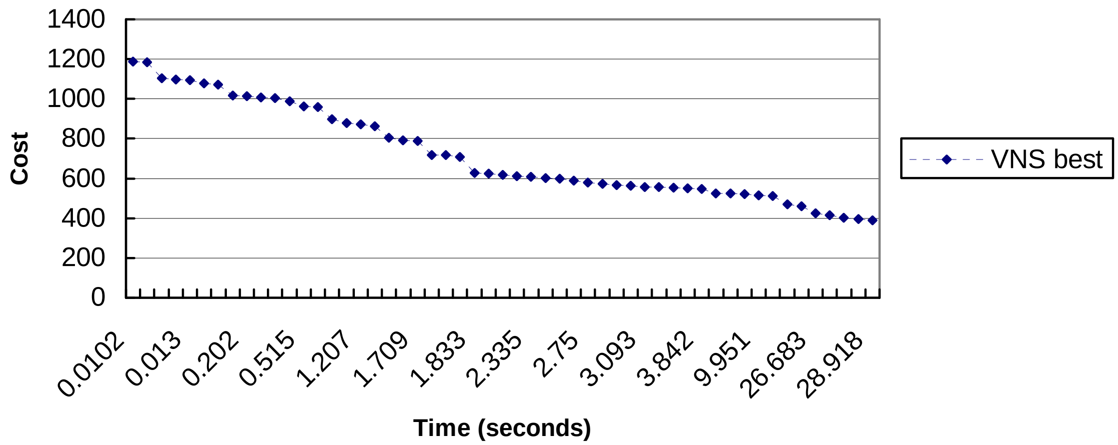



Figure 4: Sample run of the basic VNS (for instance APR)

## 6 Conclusions

This paper proposes a hybrid model within a multi-objective framework for hospital nurse rostering, in which IP and VNS are combined for global optimization. The IP is used to solve an easy handled subproblem by only including the constraints that would cause less computing complexity or be regarded as more important. A VNS with the neighbourhood of swapping blocks of shifts is then used to make the improvement on the LP's resulting solution, mainly from the aspects of satisfying the constraints that are not considered in the preceding IP model. Hence, the proposed hybrid model is able to handle all the requirements and constraints of nurse rostering in today's complex hospital environments.

Although the work in this paper is presented in terms of nurse rostering, it is suggested that the ways of decomposing the whole set of problem constraints into subsets and solving them accordingly by different approaches could be applied to a wider ranger of other problems (i.e. resource allocation problems) defined by a large number of constraints. It is also hoped that this research would shed light on the significant issue of integrating the two broad methodologies of exact algorithms and metaheuristics, and may therefore be of interest to practitioners and researchers in areas of mathematical programming and heuristic design.

Our future work will investigate other efficient ways of combining IP and local search in solving this type of highly-constrained problems. In this paper, although we have formulated a full IP model for the entire problem (in Section 3), we actually solve a partial problem due to the computational difficulty arising in handling all the constraints at one time. The interesting thing is, as shown in Table 2, that the relaxed entire IP problem (i.e. the problem without the integrality restrictions) are all solved by CPLEX to optimality at an astonishingly fast speed. Our next work is therefore to use the information gained from these relaxed fractional solutions to guide the local search, rather than simply starting the local search from the integer solutions of a partial problem.

**Acknowledgements**

The work was funded by the UK's Engineering and Physical Sciences Research Council (EPSRC), under grant GR/S31150/01.

**Reference**

[1] S. Abdullah, E.K. Burke, and B. McCollum, "An investigation of variable neighbourhood search for university course timetabling," in Proceedings of *the* 2*nd Multidisciplinary Conference on Scheduling: Theory and Applications* (*MISTA*), pp. 413-427, 2005.
[2] U. Aickelin and K. Dowsland, "Exploiting problem structure in a genetic algorithm approach to a nurse rostering problem," *Journal of Scheduling*, vol. 3, pp. 139-153, 2000.
[3] U. Aickelin and K. Dowsland, "An indirect genetic algorithm for a nurse scheduling problem," *Computers and Operations Research*, vol. 31, pp. 761-778, 2004.
[4] U. Aickelin and J. Li, "An estimation of distribution algorithm for nurse scheduling," *Annals of Operations Research*, vol. 155, pp. 289-309, 2007.
[5] B. Babu and M.M.L. Jehan, "Differential evolution for multi-objective optimization," in *Proceedings of Congress on Evolutionary Computation (CEC)*, pp. 2696-2703, 2003.
[6] J. Bard and H.W. Purnomo, "Preference scheduling for nurses using column generation," *European Journal of Operational Research*, vol. 164, pp. 510-534, 2005.
[7] N. Beaumont, "Scheduling staff using mixed integer programming," *European Journal of Operational Research*, vol. 98, pp. 473-484, 1997.
[8] S.E. Bechtold, M.J. Brusco, and M.J. Showalter, "A comparative evaluation of labor tour scheduling methods," *Decision Sciences*, vol. pp. 683–699, 1991.
[9] I. Berrada., J.A. Ferland, and P. Michelon, "A multi-objective approach to nurse scheduling with both hard and soft constraints," *Socio-Economic Planning Science,* vol. 30, pp. 183-193, 1996.
[10] R. Blau, "Multishift personnel scheduling with a microcomputer," *Personnel Administrator*, vol. 20, pp. 43–58, 1985.
[11] D. Bradley and J. Martin, "Continuous personnel scheduling algorithms: a literature review," *Journal of the Society for Health Systems*, vol. 2, pp. 8-23, 1990.
[12] P. Brucker, R. Qu, E.K. Burke, and G. Post, "A decomposition, construction and post-processing approach for a specific nurse rostering problem," in Proceedings of *the 2nd Multidisciplinary International Scheduling: Theory and Applications Conference* (*MISTA*), pp. 397-406, 2005.
[13] M.J. Brusco and L.W. Jacobs, "A simulated annealing approach to the cyclic staff-scheduling problem," *Naval Research Logistics*, vol. 40, pp. 69–84, 1993.

[14] E.K. Burke, P. De Causmaecker, and G. Vanden Berghe, "A hybrid tabu search algorithm for the nurse rostering problem," *Lecture Notes in Artificial Intelligence*, Springer, vol. 1585, pp. 187-194, 1999.
[15] E.K. Burke, P. De Causmaecker, G. Vanden Berghe, and H. Landeghem, "The state of the art of nurse rostering," *Journal of Scheduling*, vol. 7, no. 6, pp. 441-499, 2004.
[16] E.K. Burke, P. Cowling, P. De Causmaecker, and G. Vanden Berghe, "A memetic approach to the nurse rostering problem," *Applied Intelligence*, vol. 15, pp. 199–214, 2001.
[17] E.K. Burke, T. Curtis, G. Post, R. Qu, and B. Veltman, "A hybrid heuristic ordering and variable neighbourhood search for the nurse rostering problem," *European Journal of Operational Research*, vol. 188, pp. 330-341, 2008.
[18] E.K. Burke, P. De Causmaecker, S. Petrovic, and G. Vanden Berghe, "Variable neighborhood search for nurse rostering problems," in M.G.C. Resende and J.P. De Sousa (eds.), *Metaheuristics: Computer Decision-Making (Combinatorial Optimization Book Series)*, Kluwer, Chapter 7, pp. 153-172, 2004.
[19] E.K. Burke and G. Kendall (eds.), *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*. Springer, 2005.
[20] V. Chankong and Y.Y. Haimes, *Multi-objective Decision Making Theory and Methodology*. North-Holland, New York, 1983.
[21] B. Cheang, H. Li, A. Lim, and B. Rodrigues, "Nurse rostering problems––a bibliographic survey," *European Journal of Operational Research*, vol. 151, pp. 447-460, 2003.
[22] C.A. Coello Coello, "Evolutionary multi-objective optimization: a history view of the field," *IEEE Conputational Intelligence Magazine*, pp. 28-36, 2006.
[23] D. Corne, J. Knowles, and M. Oates, "The Pareto envelop-based selection algorithm for multi-objective optimization," in *Proceedings of the 6th International Conference on Parallel Problem Solving from Nature* (*PPSN-VI*), pp.839-848, 2000.
[24] J. Crispim and J. Brandão, "Reactive tabu search and variable neighbourhood descent applied to the vehicle routing problem with backhauls," in *Proceedings of the 4th Metaheuristics International Conference* (*MIC*), pp. 631–636, 2001.
[25] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A fast and elitist multi-objective genetic algorithm: NSGA-II," *IEEE Transaction on Evolutionary Computation*, vol. 6, pp.182-197, 2002.
[26] F.F. Easton and N. Mansour, "A distributed genetic algorithm for deterministic and stochastic labor scheduling problems," *European Journal of Operational Research*, vol. 118, pp. 505–523, 1999.
[27] M. Ehrgott, *Multicriteria Optimization*. Springer, Berlin, 2000.
[28] A.T. Ernst, H. Jiang, M. Krishnamoorthy, and D. Sier, "Staff scheduling and rostering: a review of applications, methods and models," *European Journal of Operational Research*, vol. 153, pp. 3-27, 2004.
[29] C.G. Gonzáles and D. Pérez-Brito, "A variable neighbourhood search for solving the linear ordering problem," in *Proceedings of the 4th Metaheuristics International Conference,* (*MIC*), pp. 181–185, 2001.
[30] R.M. Karp, "Reducibility among combinatorial problems," In R.E. Miller and J.W. Thatcher (eds), *Complexity of Computer Computations*, Plenum Press, New York, pp. 85–103, 1972.
[31] H. Kawanaka, K. Yamamoto, T. Yoshikawa, T. Shinigi, and S. Tsuruoka, "Genetic algorithm with the constraints for nurse scheduling problem," in *Proceedings of Congress on Evolutionary Computation* (*CEC*), pp. 1123-1130, 2001.
[32] P. Hansen and N. Mladenović, "Variable neighbourhood search: principles and applications," *European Journal of Operational Research*, vol. 130, pp. 449-467, 1999.

[33] A. Ikegami and A. Niwa, "A subproblem-centric model and approach to the nurse scheduling problem," *Mathematical Programming*, vol. 97, pp. 517-541, 2003.
[34] D.H. Loughlin and S. Ranjithan, "The neighbourhood constraint method: a multi-objective optimization technique," in *Proceedings of the 7th International Conference on Genetic Algorithms*, pp.663-673, 1997.
[35] N. Mladenović and P. Hansen, "Variable neighbourhood search," *Computers and Operations Research*, vol. 24, pp. 1097-1100, 1997.
[36] C.W.Mueller and J.C. McCloskey, "Nurses' job satisfaction: a proposed measure," *Nursing Research*, vol. 39, pp. 113-117, 1990.
[37] J.M. Tien and A. Kamiyama, "On manpower scheduling algorithms," *Society for Industrial and Applied Mathematics*, vol. 24, pp. 275–287, 1982.
[38] G. Post and B. Veltman, "Harmonious Personnel Scheduling," in *Proceedings of the 5th International Conference on Practice and Automated Timetabling* (*PATAT*), pp. 557-559, 2004.
[39] D. Sitompul and S. Randhawa, "Nurse scheduling models: a state-of-the-art review," *Journal of the Society of Health Systems*, vol. 2, pp. 62-72, 1990.
[40] L.D. Smith, D. Bird, and A. Wiggins, "A computerized system to schedule nurses that recognizes staff preferences," *Hospital & Health Service Administration*, pp. 19–35, 1979.
[41] M. Warner and J. Prawda, "A mathematical programming model for scheduling nursing personnel in a hospital," *Management Science*, vol. 19, pp. 411–422, 1972.
[42] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: improving the strength Pareto evolutionary algorithm for multi-objective optimization," in K.C. Giannakoglou et al. (eds.) *Proceedings of Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, pp.95-100, 2001.