Local Search Neighbourhoods to Deal with a Novel Nurse Rostering Model

Burak Bilgin · Patrick De Causmaecker · Benoît Rossie · Greet Vanden Berghe

Received: date / Accepted: date

Abstract A novel nurse rostering model is developed to represent real world problem instances more accurately. The proposed model is generic in a way that different problem instances can be successfully represented. Novel local search neighbourhoods are developed to take advantage of the problem properties represented by the model. These neighbourhoods are utilised within a variable neighbourhood search algorithm. The performance of the solution method is evaluated empirically on real world data. Experimental results are processed using statistical methods. An adaptive version of the solution method that uses problem instance properties to select the neighbourhood set is also proposed and experimented with. The proposed model is prone to further extensions to cover personnel planning problems in different sectors and countries. Associated new solution methods can be studied to handle these extensions.

Keywords Nurse Rostering · Hospital Personnel Planning · Variable Neighbourhood Search

1 Introduction

The nurse rostering problem, which consists of the assignment of shifts to nurses according to several criteria, is a complex personnel planning problem [12]. The problem becomes even more complicated in Belgian hospitals where the schedule periods are

B. Bilgin, G. Vanden Berghe
KaHo Sint-Lieven, Information Technology
Gebroeders Desmetstraat 1, 9000 Gent, Belgium
E-mail: {Burak.Bilgin, Greet.VandenBerghe}@kahosl.be

P. De Causmaecker
K.U.Leuven Campus Kortrijk
Computer Science and Information Technology
Etienne Sabbelaan 53, 8500 Kortrijk, Belgium
E-mail: Patrick.DeCausmaecker@kuleuven-kortrijk.be

B. Rossie
SAGA Consulting NV, Products Division
Weiveldlaan 41 bus 35, B-1930 Zaventem, Belgium
E-mail: Benoit.Rossie@saga.be

flexible, problem elements like shift types and skill types are user defined, legal restrictions and contractual agreements impose complex constraints and cyclic assignments are not the standard practice [8, 11, 17]. Furthermore, the problem has a dynamic nature due to the ever changing labour legislation, contractual agreements, and nurse preferences. New aspects and constraint types are introduced in the course of time. The models and solution methods of the automation tools have to comply with the latest state of the problem.

The nurse rostering problem is tackled using diverse solution methods. Variable neighbourhood search (VNS) is a metaheuristic that systematically switches between the neighbourhoods of a predefined neighbourhood set during the local search [14]. Burke et al. [9] apply a hybridisation of heuristic ordering and VNS to a nurse rostering problem instance in intensive care units in Dutch hospitals. Brucker at al. [7] propose a three step method to tackle the same problem instance. An initial schedule is created via a cyclic constructive method based on hard and soft constraints that is followed by a procedure to fulfill the coverage constraints. Then the resulting initial schedule is improved with a VNS. A tabu search hyperheuristic is applied to a real world problem at a hospital in the UK by Burke et al [10]. Aickelin and Li [1] apply an estimation of distribution algorithm to real world problem instance from a UK hospital. Bellanti et al. [4] developed a greedy algorithm that completes partial solutions, created by either a tabu search or an iterative local search algorithm. The solution method developed is applied to the nurse rostering problem instance encountered at a ward in an Italian hospital. Maenhout and Vanhoucke [15] evaluate the performance of various problem specific cross-over methods on a synthetic data set called NSPLib and combine these methods within a genetic algorithm using three different strategies: VNS, parent-related and instance-related hybridisations. A novel population based metaheuristic that is inspired by electromagnetism is also applied to the NSPLib [16]. A hybrid solution method based on lagrangian relaxation of coverage constraints is developed as a composition of various optimisation strategies and applied to real world problem instance encountered in a US hospital by Bard and Purnomo [2].

In [6], constraints are grouped into rule categories. The fact that coverage constraints and *maximum number of assignments per nurse per schedule period* constraint are considered hard, enables the reduction of the solution space by inference prior to search. Özcan proposes the utilization of a hyperheuristic as a hill-climber in a memetic algorithm in [18]. The heuristic set deployed within this hyperheuristic consists of constraint-based hill-climbers. The performance of the proposed method is empirically evaluated and compared against the performance of the standart genetic algorithm and two self-generating multimeme memetic algorithms. The testbed for this study was a randomly generated set of nurse rostering problem instances.

The nurse rostering problem in Belgian hospitals has been subject of many research papers. The problem has been tackled with a hybrid tabu search algorithm that utilises diversification heuristics and greedy postprocessing [11], a hybridisation of this algorithm with a memetic approach deploying a steepest descent heuristic [8], and a VNS that takes advantage of the synergy between simple and greedy neighbourhoods [11]. The contribution of coverage constraint relaxation to the production of higher quality schedules is investigated in [17]. Beliën and Demeulemeester [3] compare and discuss the performance and modelling capabilities of two different problem decomposition strategies within a branch-and-price algorithm. The problem instance tackled is a trainee scheduling problem at a Belgian hospital department.

The motivation of this study was twofold. We aimed towards a generic nurse rostering model to allow a broad range of real world problems to be defined accurately. The proposed model has an increased ability to reflect possible developments on the nurse rostering problem and extra soft constraints in different hospitals, sectors and countries. The second part of the motivation was to develop an associated solution method. The contribution of this work considering the solution methods is the development of neighbourhoods that take advantage of different properties of the model we propose. These neighbourhoods, when deployed among other traditional neighbourhoods in VNS, have demonstrated higher performance compared to the performance of the traditional neighbourhoods alone.

Nurses belong to different skill categories according to their job descriptions, qualifications, experience, and responsibilities. Every nurse has a primary skill type. Some of the nurses have secondary skill types, which means they can substitute nurses with these skill types. The coverage constraints restrict the number of nurses with a specific skill type that should be present at a given date and a given shift type [1, 11]. The coverage constraints are considered as soft constraints in the new model. A new hard constraint is defined, which allows operations to be executed only on defined assignments. The assignments are defined in the coverage constraints.

The time related constraints, considered as soft constraints in the model, restrict the assignments to a specific nurse [12]. Constraints defined by the employment contracts of the nurses, called *horizontal constraints*, are a subset of the time related constraints. The *horizontal constraints* are organized in three general types in this model. These are *counters*, *series*, and *successive series*, each with their own subjects and parameter sets. This organization is more flexible than the approach in [11] where soft constraints were predefined with all or most of their parameters and then applied to all problem instances. *Counter* constraints restrict the number of occurances of an item over a counter period. *Series* is a general term to describe the constraints that restrict the number of consecutive items, like *days worked*, *days idle*, *weekends worked*, etc. In this context, especially *successive series* is a new formulation. *Successive series* restricts the succession of two series. An early example to a *successive series* constraint is *assign two free days after night shifts* constraint in [11]. Another novel aspect of the model is the *compatible shift types* concept. Some of the soft constraints involving shift types are defined on a set of *compatible shift types* instead of on a single shift type as in [11].

The second part of the study is the investigation of associated solution methods. Neighbourhoods based on the proposed model are developed to be utilised within a VNS algorithm. The proposed neighbourhoods make use of problem properties like compatible shift types and secondary skill types. The solution methods developed are evaluated empirically. The soundness of the timetabling research is addressed by Schaerf and Di Gaspero in [19]. The experimental setup and the processing of the results are aimed to be carried out according to the principles given in [19]. The experimentation is undertaken on a diverse set of test data and test scenarios, which are published in [5]. Different algorithm settings are experimentally evaluated on this data and the results are statistically tested.

The research is carried out with an industry partner [1] producing a nurse rostering assistance system [2]. This is a computer aided nurse rostering tool to assist the planners by keeping track of the constraint violations. The implementations of the system are

[1] SAGA Consulting NV

[2] HCPS: Health Care Personnel Scheduling

utilised in many Belgian hospitals. The research is undertaken on the current state of the real life problem by utilisation of previous experience on the field [11]. The automation tool developed within this research project is integrated in the nurse rostering assistance software of our industry partner. The resulting system is being deployed in Belgian hospitals.

The problem definition and the model are described in Section 2. The solution method is presented in Section 3. The experiments and the experimental results are discussed in Section 4. The paper is concluded in Section 5.

2 Problem Definition and Model

The objective of the nurse rostering problem is to assign shifts to nurses in accordance with workforce requirements, legal and contractual restrictions, personal preferences, and further criteria [12]. We define the problem model by the search space, schedule, hard and soft constraints. The set of all possible solutions is represented by the search space, which contains all possible mappings from the shift types and skill types set to the set of nurses and the schedule period. Any candidate solution in the search space corresponds to a schedule. A schedule is feasible only if it fulfills all the hard constraints. The amount to which the soft constraints are satisfied, determines the quality of the schedules.

2.1 Search Space

The search space of the problem is composed of the schedule period, skill types, shift types, and nurses [11].

2.1.1 Schedule Period

The schedule period is defined by a start date and the period length given as a number of days. This definition makes the model more complicated than the approach that is based on fixed schedule periods [1, 7, 9]. The schedule period length differs between wards. Among the real world response groups, the most common schedule periods are one month and four weeks. In some periods of the year like Christmas holidays, shorter schedule periods like two weeks are considered. The bank holidays within the schedule periods are also a part of the problem parameters.

2.1.2 Skill Types

In hospitals tasks are divided between nurses according to their job descriptions, qualifications, experience, and responsibilities. This division is formalised by skill types. Each skill type defines a specific skill category. In our model, each nurse has a primary skill. Nurses can also have one or more secondary skills. The skill categories are not fixed in the problem definition. Instead they are defined by the users for each problem instance. The hierarchical substitution, which means nurses of a higher rank can substitute nurses of a lower rank nurses, are not implicitly foreseen in the model. In Belgian hospitals, nurses do not prefer to carry out the tasks defined for a lower skill type. Therefore only user defined secondary skill types are considered by the solution method.

WD = TD - WED - HD
WD : Working days in the period
TD : Total days in the period
WED: Number of weekend days in the period
HD : Number of bank holidays in the period

Table 1 Calculation of the Working Days

TJT = WD * WJT / 5
TJT : Total job time
WD : Working days in the period
WJT: Weekly job time

Table 2 Calculation of the Total Job Time

2.1.3 Shift Types

The daily assignments are made in terms of shifts. Shifts are time periods defined with specific start and end times, rest periods before the start and after the end time, and a net job time. The definition and number of shift types are taken as parameters. This property makes the problem more complex than usual problem definitions with fixed number of shift types with predetermined definitions [2,4,7,9].

2.1.4 Nurses

The set of nurses is a user-defined parameter. The nurse definition is generic in the model to allow the user to represent the properties of each nurse accurately. An *employment contract* is defined with a start and end date, weekly job time, and a constraint set. Weekly job time denotes the total amount of time the nurse is supposed to work in a week. This definition is based on the ideal state where the nurses work from Monday to Friday. Since the real life situation does not match the ideal state weekly job times need to be adapted to the real life. First the associated period is determined. This period can be the whole schedule period if the associated employment contract covers it. If the associated employment contract covers the schedule period partially then this partial period is taken as the base for the job time calculations. The second step is to determine the number of working days within the period (Table 1). Then the total job time is calculated using the formula in Table 2. Different weekly job times can be applicable to different parts of the same schedule period serially. The fact that each nurse can have her own *employment contract* increases the individuality of nurses and therefore the complexity of the problem.

2.2 Schedule

The schedule is composed of a set of assignments that are defined as quadruples of (nurse, day, shift type, skill type). An empty, full or partially full schedule can be input to the solution method.

2.3 Hard Constraints

The schedule needs to satisfy all the hard constraints in order to be feasible. The hard constraints are presented in Table 3.

2.3.1 Single Assignment Start Per Nurse Per Day

For each nurse only one assignment start per day is foreseen in the model, similar to [7,9].

2.3.2 No Overlap between Assignments

The assignment of two shifts with an overlap in the work periods to one nurse is not allowed, as opposed to the model in [8]. A hypothetical shift type, called *free shift*, is assigned to a nurse to balance the extra hours worked before. The nurses do not actually work during *free shifts* and therefore these shifts are the only exception to the *No Overlap between Assignments* constraint.

2.3.3 Honour Skill Types

An assignment is allowed only if the skill type matches one of the nurses skill types, either primary or secondary.

2.3.4 Schedule Locks

In some real life situations the automation tool is prevented to make alterations on some specific parts of the input schedule. These can be simple cases like the grant of an absence request or more complicated situations like the partial rescheduling due to an unforeseen absence of a nurse. In the latter case the assignments of the absent nurse need to be redistributed among other nurses. However, due to the time related constraints this operation may modify the unaffected parts of the planning period as well. On the other hand for several reasons alterations on an announced roster are avoided to the extent that it is possible. To meet this criterion, the unaffected parts of the schedule are locked before it is given as an input to the algorithm. Schedule locks are defined with (nurse, day) pairs. The objective function always evaluates the complete schedule regardless of schedule locks.

2.3.5 Operations on Defined Assignments Only

The shift types are not always relevant to all skill types. For example, night shifts are not assigned to head nurses in many wards. Therefore coverage constraints are given with associated skill types. The assignments are only allowed if they are defined in the coverage constraints. On the other hand some input schedules contain assignments that are not mentioned in the coverage constraints. These are preassigned schedule parts which the solution method is not allowed to modify, even if they are not locked. These restrictions are defined as *Operations on Defined Assignments Only*. They reduce the size of the feasible search space and prevent the solution method from searching the irrelevant parts of the search space.

Hard Constraints	Soft Constraints
Single Assignment Start Per Nurse Per Day	Coverage Constraints
No Overlap between Assignments	Assignment to the Primary Skill
Honour Skill Types	Rest Times
Operations on Defined Assignments Only	Requests
Schedule Locks	Horizontal Constraints

Table 3 Constraints

Short Early [08:00 12:00]	Assigned to part timers
Regular Early [08:00 14:36]	Assigned to 70% time working nurses
Long Early [08:00 17:00]	Assigned to full timers

Table 4 Compatible Shift Types Set of Early Shift Types

2.4 Soft Constraints

Satisfying the soft constraints is not necessary for the feasibility of the solution but the degree of the soft constraint satisfaction determines the quality of the solution. The quality of a schedule is measured by an objective function that is the linear combination of the number of violations for each soft constraint, similar to [13]. The soft constraints of this problem are presented in Table 3. Coverage constraints, rest times, assignment to the primary skill constraints are global, meaning that they are applicable to the whole schedule. Requests and horizontal constraints are specific to nurses. However nurses with similar contracts have similar horizontal constraints.

The threshold values of the coverage constraints and horizontal constraints, can be either a minimum, a maximum, or a range defined by a minimum and a maximum value. The weights used in the objective functions are specific for each *coverage constraint* and *horizontal constraint*. They are global for the *rest times* and *assignment to the primary skill* constraints. The weights can be any positive integer value. The weight parameterisation is a very complex task for the planner because of the high number of possible combinations, nevertheless it is crucial to get satisfactory rosters [9]. The planners will have to experiment with different weight settings and build up experience over time in order to find the weight settings that result in rosters, which satisfy their priorities. In some preliminar testing and demo sessions we could actually performed this kind of tuning operation when prompted to do so by a planner.

2.4.1 Compatible Shift Types

A new concept introduced in this research paper is called compatible shift types. The coverage constraints and horizontal constraints that involve shift types are defined on a non-empty set of compatible shift types, instead of on a single shift type. The compatible shift types sets may not be the same for each constraint. A hypothetical example is given in Tables 4 and 5. In Table 4, the shifts are grouped according to their start times, where the compatible shift types set in Table 5 is constructed around the preferences of nurses.

Early Morning [05:00 11:00]	Starts too early
Day [11:00 20:36]	Covers the whole day
Late [17:00 02:00]	Ends too late

Table 5 Compatible Shift Types Set of Unpreferred Shift Types

2.4.2 Coverage Constraints

The number of nurses needed for each day, skill type and a compatible shift types set are called *coverage constraints* [12]. Coverage constraints are considered as hard constraints in many problem instances [1, 7, 9, 11]. However, the over-constrained nature of the problem in Belgium make relaxations in coverage constraints necessary. The relaxation methods were investigated and implemented in automated nurse rostering tools [17]. The coverage constraints are considered to be soft constraints by the real world response groups.

Although compatible shift types sets increase the accuracy of modeling the real life problem, they should be handled with care. In some situations errors may occur, if the domains of two different soft constraint instances overlap. If the compatible shift types sets of two coverage constraints on the same day and for the same skill type have common elements, any assignment of such elements contributes to both coverage constraints. The compatible shift types sets should be disjoint in such cases to avoid errors.

2.4.3 Assignment to the Primary Skill

An assignment to one of the secondary skills of an nurse is considered a soft constraint violation.

2.4.4 Rest Times

For each shift type a period of rest time is defined before the start and after the end time. An assignment of a shift that overlaps with the rest period of another assignment is considered as a soft constraint violation.

2.4.5 Requests

In nurse rostering models, nurses are allowed to request specific assignments of free days or periods [11,4]. In this model two types of requests are defined: assignment and absence requests. Assignment requests are defined by a shift type preferred to be assigned to a specific nurse on a specific day. Absence requests are defined by a specific day and a specific period where no assignment is foreseen for a specific nurse. An absence request also involves a job time which contributes to the total job time of the nurse if the absence request is granted. More than one absence request in the same day can be defined for a nurse but the periods of the requests are not allowed to overlap.

2.4.6 Horizontal Constraints

The soft constraints imposed by the employment contracts of the nurses are called horizontal constraints. Various instances of horizontal constraints are present in nurse rostering problems from different countries [4,7,11]. In the proposed model, horizontal constraints are generalized in three categories being *counters*, *series* and *successive series*. This is a generic approach that allows users to define horizontal constraints, with specific subjects and further parameters.

Counters. The horizontal constraints that restrict the number of specific instances over a period are called counters. The counter period is defined by a start time and a length, which is given as a number of days. It does not necessarily match the schedule period. If the counter period starts before the schedule period, the counter value at the start of the schedule period is given as an input to the solution method. Minimum thresholds cannot induce violations if the counter period exceeds the schedule period, because they can be met in the upcoming schedule period [11]. There are six subjects for counters: *hours worked*, *shift types worked*, *days worked*, *days idle*, *weekends worked* and *weekends idle*. *Shift types worked* counters are defined on a compatible shift types set. Apart from *weekends worked* and *weekends idle*, all counters have the *day types* parameter. The *day types* parameter can have the value of either any, holidays or a set of week and weekend days.

Series. The number of consecutive occurences of specific instances are restricted by series. There are five subjects for series: *shift types worked*, *days worked*, *days idle*, *weekends worked* and *weekends idle*. *Shift types worked* series are also defined on a compatible shift types set. The algorithm not only checks the series that start and end within the schedule period but also the series that start in the previous schedule period and extend to the current schedule period. Similar to the counters, the minimum threshold violations by series that can be compensated in the next schedule period are not penalized [11]. The problem model also considers the previous schedule period when series have started but not finished. Therefore, the schedule information of the associated parts of the previous schedule period are taken as a part of the input by the solution method.

Successive Series. Another type of horizontal constraint is the restriction and imposition of the succession of two series, each with their own threshold values. Any occurance of the first series implies the second series to follow. Deviations from the second series are penalized. Possible orders of series are *days worked - days idle*, *days idle - days worked*, *shift types worked - days idle*, *days idle - shift types worked* and *shift types worked - shift types worked*. Similar to the series and counters, successive series involving shift types are defined on a compatible shift types set. Again the minimum threshold violations that can be covered in the next schedule period are not penalised [11]. Similar to the series, successive series that have been started but not ended in the previous schedule period are considered within the problem model and the associated schedule information is taken as input by the solution method. Similar to the coverage constraints, the compatible shift types should be used with care in successive series. In case of the succession of two *shift types worked* series, the compatible shift types sets of both series should be disjoint. If these sets contain common elements, an ambiguity about the end of the first series and the start of the second series might occur.

3 Solution Method

The solution method consists of a VNS algorithm preceeded by a preprocessing method. The solution method does not schedule nurses with different skill types seperately as in [11]. This allows to exploit the advantages offered by secondary skill types. The solution method is not allowed to make modifications that result in an infeasible schedule. The termination criterion involves the maximum execution time of the algorithm, without taking the preprocessing step into account. The pseudocode of the algorithm is presented in Algorithm 1.

Algorithm 1 Pseudocode of the solution method

```
S = Initial Schedule
Preprocessing
Add assignments randomly to S to meet the minimum coverage constraints
Add assignments randomly to S to meet the minimum job times
Variable Neighbourhood Search
CQ = Circular Queue of the Neighbourhoods
N = First Neighbourhood in CQ
BS = S
while Termination criterion not met do
   S* = Tabu search(N(S))
   if cost(S*) < cost(BS) then
      Decrement tabu length
      BS = S*
   else
      Increment tabu length
      if cost(S*) ≥ cost(S) then
         N = next neighbourhood in CQ
      end if
   end if
   S = S*
end while
return BS
```

3.1 Preprocessing

The solution method accepts an input schedule from the user. The input schedule can be an empty, a partial or a complete schedule. The preprocessing method consists of two steps. This method tries to fulfill the minimum coverage constraints first and the weekly job times of each nurse in the second step by adding assignments.

3.2 Variable Neighbourhood Search

The VNS algorithm utilises several neighbourhoods and holds the parameters of the executed moves in a tabu list. The neighbourhoods utilised are presented in Table 6. At each iteration of the algorithm, a single neighbourhood is applied. Each possible move in the neighbourhood is checked and the best move that complies with the hard constraints and is allowed by the tabu list is executed. The schedule remains feasible throughout the execution of the algorithm. The exceptions to the strict steepest descent

Assign Shift
Delete Shift
Single Shift-Day
Change Assignment based on Compatible Shift Type
Change Assignment based on Skill Type
General Assignment Change

Table 6 Problem-specific Neighbourhoods

practice are discussed in the corresponding paragraphs about each neighbourhood. The neighbourhoods are held in a circular queue and called in this sequence. If the applied neighbourhood does not result in an improving move, the algorithm switches to the next neighbourhood in the queue. Different neighbourhood sets are subject to experiments in order to measure their contribution to the search in Table 10.

The function of the tabu list is to avoid the cycles of the algorithm around local optima. The parameters of the executed moves are expressed in quadruples (nurse, day, shift type, skill type). The length of the tabu list is variable during the execution. It is increased at each non-improving iteration and decreased if there is an improvement. Prime numbers are used as values for the length of the tabu list and a lower and an upper bound limits this variation, in order to avoid hash collisions, when encoding the elements of the tabu list. We set the lower bound equal to seven and the upper bound is a parameter. An aspiration method that allows the moves in the tabu list resulting in overall best candidate solutions, is used.

3.2.1 Assign Shift

Since an assignment is defined as a quadruple of (nurse, day, shift type, skill type) (Section 2.2), the Assign Shift neighbourhood operates on these quadruples. When an assignment is made, not only a shift type is assigned, but also the associated skill type for that assignment. High numbers of shift types, which is common in Belgian hospitals, result in a large Assign Shift neighbourhood. To overcome possible inefficiencies as a result of this fact, only one random shift type is considered for each (nurse, day, skill type) triple.

3.2.2 Delete Shift

The deletion of an assignment is feasible only if the assignment is defined in coverage constraints and not locked (Table 3). Since the coverage constraints are not defined as hard constraints, deletions that violate coverage constraints are considered feasible as well (Section 2.4). Delete Shift neighbourhood consists of all the feasible deletion moves. A deletion move, not only deletes the assigned shift type at that given timeslot, but also the skill type, since this is a property of the assignment as well.

3.2.3 Single Shift-Day

An assignment is removed from a nurse's schedule and added to another nurse on the same day if the second nurse has no assignment on that day and has the associated skill type [11].

3.2.4 Change Assignment based on Compatible Shift Type

The shift type of an assignment is changed to another compatible shift type defined in the coverage constraints for the associated day and skill type. One random shift type from the same compatible shift types set is considered for each assignment according to the same motivation in *Assign Shift* neighbourhood.

3.2.5 General Assignment Change

An assignment is changed to another shift type where the skill type of the assignment remains the same. This neighbourhood does not necessarily consist of the compatible shift types from the same coverage constraint, as in *Change Assignment based on Compatible Shift Type* neighbourhood. Again a subset of the complete neighbourhood is considered as in the *Assign Shift* neighbourhood. The subset simply consists of a single alternative random shift type for each assignment.

3.2.6 Change Assignment based on Skill Type

This neighbourhood operates on nurses with at least two different skill types. It deletes an assignment and adds another assignment to one of the nurse's other skill types.

4 Experiments

The solution method needs to cope with different situations and scenarios that might occur in real world. Hospitals are organized in wards, each with different settings of problem variables: schedule periods, nurse properties, shift types, skill types, and soft constraints. Variations and unexpected changes in the workload of hospital wards are not rare. Sample scenarios are *overload of work*, for example in case of an epidemic, and *unexpected absence of a nurse* in case of an illness. In the latter case, partial rescheduling of the complete nurse roster is needed. Various solution method settings are experimented with different scenarios in order to measure the performance of the method.

As is often the case in nurse rostering papers, we could not compare our methods with existing benchmarks since the systems under study have too many unique properties. Instead, we performed a carefully planned series of experiments to arrive at a statistically relevant and internally consistent comparison of solution methods. Consequently, the contribution of this paper is in designing and validating new models and associated solution methods for nurse rostering in situations where we identified a number of new characteristics.

4.1 Experimental Settings

The real world data of six different wards from two different hospitals are covered in the experiments. These are Emergency, Psychiatry, Reception, Meal Preparation, and Geriatrics from Hospital 1 and Palliative Care from Hospital 2. For each of the wards, three different scenarios are considered. The first scenario has normal settings with an empty input schedule (*normal*). The second scenario considers an overload of work,

Ward	Start Date	Period	Absence Start Date
Emergency	03/12/2007, Monday	4 weeks	10/12/2007, Monday
Psychiatry	01/12/2007, Saturday	1 month	10/12/2007, Monday
Reception	14/4/2008, Monday	6 weeks	5/5/2008, Monday
Meal Preparation	1/2/2008, Friday	1 month	18/2/2008, Monday
Geriatrics	25/2/2008, Monday	4 weeks	17/3/2008, Monday
Palliative Care	31/12/2007, Monday	13 weeks	4/2/2008, Monday

Table 7 Schedule Periods and Absence Start Dates

Ward	Shift Types	Skill 1	Skill 2	Skill 3	Skill 4	Total Employees
Emergency	27	1	15	4	26	27
Psychiatry	14	1	17	1	-	19
Reception	19	1	1	3	15	19
Meal P.	9	1	31	-	-	32
Geriatrics	9	4	20	-	-	21
P. Care	23	1	21	4	1	27

Table 8 Number of shift types and number of nurses with each skill type

resulting in higher values for the coverage constraints (*overload*). The last scenario is the unexpected absence of a nurse (*absence*). In that case the complete schedule is taken into account but only the affected parts are required and allowed to be modified. The execution time is ten minutes for *normal* and *overload* scenarios and one minute for the *absence* scenario.

Start dates and period properties of the schedule periods of the wards are given in Table 7. In the *absence* scenarios the entire schedule is locked except the one week periods that start at dates mentioned in Table 7. The number of shift and skill types for each ward is given in Table 8. Although some of the shift types in these wards have identical working times, they have different tasks attached. Therefore they are treated within different compatible shift types sets in horizontal constraints and coverage constraints. The number of nurses with each skill type and the total number of nurses are presented in Table 8. It is clear from this table that several nurses have secondary skill types. The wards have different contract types according to their weekly job time. The weekly job time and the number of nurses for each weekly job time is given in Table 9. Some of the nurses change from one employment contract to another within the given schedule period. As a result these nurses have more than one weekly job time within the same schedule period, which are applied serially.

The experiments were undertaken with eleven different solution method settings. The settings were composed according to their neighbourhood set and upper bound of the tabu list length. The neighbourhood sets used are presented in Table 10. The neighbourhoods are called in the order they appear in Table 10. Two different upper bounds for the tabu list length are 97 and 197. The model and the solution method is implemented in C#. The experiments are carried out on MS Visual Studio 2005 Professional Edition. The operating system was MS Windows Server 2003 Enterprise Edition SP 2 running on an Intel Pentium 4 CPU with 2.40 GHz and 2.00 GB of RAM.

Ward	38 hours (100%)	34.2 hours (90%)	30.4 hours (80%)	28.5 hours (75%)	22.8 hours (60%)	19 hours (50%)
Emergency	24	-	-	3	-	-
Psychiatry	13	-	-	2	-	4
Reception	5	-	-	7	-	7
Meal P.	3	2	-	1	-	28
Geriatrics	9	-	-	9	1	3
P. Care	13	-	2	4	1	7

Table 9 The weekly job time and the corresponding number of nurses

Set 1	assign shift, delete shift, single shift-day
Set 2	assign shift, delete shift, single shift-day, change assignment based on compatible shift type
Set 3	assign shift, delete shift, single shift-day, change assignment based on skill type
Set 4	assign shift, delete shift, single shift-day, general assignment change
Set 5	assign shift, delete shift, single shift-day, general assignment change, change assignment based on skill type

Table 10 Neighbourhood Sets

4.2 Experimental Results

The objective of the experimentations is to find the algorithm setting that is among the best performers on most of the ward-scenario couples with respect to the penalty cost. The performance results of different algorithm settings on each ward-scenario couple are compared with each other using Student's t-test. Each solution method setting is executed ten times on each ward-scenario couple and the confidence interval for Student's t-test is taken as 95%. For each ward-scenario couple; the input data, a sample solution obtained from the algorithm, and penalty details to this sample solution are published in [5].

The experimental results for each algorithm setting, ward-scenario couple are presented in Tables 11, 12, 13, 14, 15, and 16. In these tables the algorithm settings that performed significantly better than the rest are highlighted with bold characters. The best results on normal scenarios, that are achieved by human planners using the nurse rostering assistance system (mentioned in Section 1) are also given in the tables.

The experimental results are analysed based on the performance variances between different algorithm settings. Therefore the results on the ward-scenario couples, where no performance variances are encountered, are not taken into account. These are Hospital 1 Emergency-Overload, Psychiatry-Absence, Reception-Absence, and Hospital 2 Palliative Care-Absence scenarios. On the remaining ward-scenario couples, different algorithm settings performed significantly better than the remaining settings.

The algorithm settings with upper bound 197 for the tabu list length are not among the best performing group except on the Hospital 1 Emergency-Normal, Geriatrics-Absence and Hospital 2 Palliative Care-Normal scenarios. For these exceptional cases, this algorithm setting is not the unique best performer. These results suggest that the

A. Setting		Normal		Overload		Absence	
N. Set	T. Limit	Average	St. Dev	Average	St. Dev	Average	St. Dev
1	97	12157.17	480.29	26871.33	258.41	21861.17	123.66
1	197	15003.83	664.81	27262.83	290.42	21872.17	29.95
2	97	**11292.67**	**298.89**	27550.00	579.28	21874.67	34.33
2	197	14425.00	634.12	27027.67	254.43	21877.67	21.19
3	97	**11288.17**	**213.98**	27553.00	377.40	21231.67	34.88
3	197	11807.17	260.60	26890.33	174.51	21214.67	50.23
4	97	**11463.67**	**332.02**	26974.67	168.46	21359.67	121.41
4	197	13097.33	573.81	26881.67	192.11	21606.33	256.07
5	97	**11322.17**	**355.68**	27562.17	318.85	**21175.67**	**19.12**
5	197	**11228.67**	**181.49**	26945.17	88.41	21224.67	42.77
6	97	**11294.67**	**281.96**	27800.17	370.83	**21177.67**	**26.33**
Human Planner		49236		-			

Table 11 Hospital 1 Emergency Results. A. Setting, N. Set, T. Limit, St. Dev. denote algorithm setting, neighbourhood set, upper bound for the tabu list length, and standard deviation respectively.

choice of the value 97 for the upper limit of the tabu list length will result in better performance.

The basic neighbourhood set 1 (Table 10) is not among the best performing neighbourhood sets except for the Hospital 1 Geriatrics-Normal and Geriatrics-Absence scenarios and Hospital 2 Palliative Care-Normal scenario. For these exceptional cases, the basic neighbourhood is again not the unique best performer. The contribution of the problem specific neighbourhoods that take advantage of the problem properties like secondary skill types and compatible shift types are emphasized by this result.

The performance comparison between the neighbourhood sets 2 and 4 is meaningful in a sense that the *general assignment change* neighbourhood used in set 4 is a variant of the *change assignment based on compatible shift type* in set 2. Neighbourhood set 4 is among the best performers eleven times, where neighbourhood set 2 is among the best performers only four times. This result suggest that the choice of neighbourhood set 4 over neighbourhood set 2 is logical.

A similar comparison can be undertaken between neighbourhood sets 3 and 5. Both neighbourhood sets contain the basic neighbourhoods of neighbourhood set 1 and the *change assignment based on skill type* neighbourhood. In addition to these, neighbourhood set 5 contains the *general assignment change* neighbourhood. In this case, neighbourhood set 5, which is among the best performers on twelve ward-scenario couples, is more successful than neighbourhood set 3 that is among the best performers on only three ward-scenario couples.

Although neighbourhood sets 4 and 5 perform better than the remaining neighbourhood settings there are no clear performance variations between each other that will make us favour one over another. A solution proposal to this decision problem is to use a selection strategy that chooses between neighbourhood sets 4 and 5 according to the input data properties. The only difference between neighbourhood sets 4 and 5 is that the neighbourhood set 5 involves the *change assignment based on skill type* among other good performing neighbourhoods. An adaptive version of the solution method can decide for neighbourhood set 5 if the problem instance involves nurses with secondary skill types and for neighbourhood set 4 otherwise. Such an adaptive version is implemented as the neighbourhood set 6.

A. Setting		Normal		Overload		Absence	
N. Set	T. Limit	Average	St. Dev	Average	St. Dev	Average	St. Dev
1	97	9202.00	105.70	11924.00	377.48	13095.00	136.81
1	197	10445.00	364.12	14729.00	715.86	12855.00	168.14
2	97	9164.00	126.95	11858.00	304.26	13015.00	253.04
2	197	10720.00	336.65	14525.00	623.44	12912.00	162.06
3	97	9099.00	123.69	11782.00	317.59	12966.00	233.01
3	197	10498.00	311.98	14913.00	463.73	12866.00	163.52
4	97	**8706.00**	**134.43**	**10643.00**	**65.50**	13034.00	209.14
4	197	9238.00	123.72	11700.00	216.90	12870.00	172.43
5	97	**8715.00**	**150.06**	**10741.00**	**109.18**	12858.00	169.17
5	197	9277.00	132.08	11684.00	227.46	12899.00	259.72
6	97	**8721.00**	**154.16**	**10658.00**	**105.60**	12819.00	255.45
Human Planner		35480		-			

Table 12 **Hospital 1 Psychiatry Results**. A. Setting, N. Set, T. Limit, St. Dev. denote algorithm setting, neighbourhood set, upper bound for the tabu list length, and standard deviation respectively.

A. Setting		Normal		Overload		Absence	
N. Set	T. Limit	Average	St. Dev	Average	St. Dev	Average	St. Dev
1	97	23715.67	407.31	54745.17	348.49	28866.17	208.91
1	197	26005.17	605.38	55646.17	420.90	28924.67	203.49
2	97	**22234.67**	**197.33**	53797.17	292.49	28841.17	192.59
2	197	23537.17	393.03	54038.67	239.83	28832.67	163.33
3	97	23741.67	261.85	54976.17	357.49	28793.67	189.58
3	197	25800.17	380.86	56035.67	467.77	28841.67	131.30
4	97	22586.67	206.32	**53419.67**	**143.86**	28691.67	155.51
4	197	23301.17	344.81	54112.17	206.36	28868.17	139.28
5	97	22462.17	233.15	**53414.17**	**228.00**	28648.67	120.74
5	197	23426.67	378.95	54077.17	264.33	28825.67	247.34
6	97	22438.67	190.00	**53477.17**	**167.22**	28649.67	130.77
Human Planner		48358		-			

Table 13 **Hospital 1 Reception Results**. A. Setting, N. Set, T. Limit, St. Dev. denote algorithm setting, neighbourhood set, upper bound for the tabu list length, and standard deviation respectively.

The experimental results of neighbourhood set 6 are foreseeable since this set is a selection between the sets 4 and 5 according to the problem instance. As a matter of course, the neighbourhood set 6 is among the best performers on all ward-scenario couples with two exceptions. The first exceptional case, on the Hospital 1 Reception-Normal scenario, both neighbourhood sets 4 and 5 are not among the best performers. The second exceptional case, on the Hospital 1 Meal Preparation-Normal scenario, our rule of thumb to select between neighbourhood sets 4 and 5 does not work as expected. Even with these exceptions taken into account, neighbourhood set 6 performs at least as good as neighbourhood sets 4 and 5. From the experimental results, it can be concluded that the neighbourhood set 6 has a convincing potential of successfully tackling the nurse rostering problem instances defined within the model presented in Section 2.

A. Setting		Normal		Overload		Absence	
N. Set	T. Limit	Average	St. Dev	Average	St. Dev	Average	St. Dev
1	97	4771.93	541.06	11925.20	382.26	5756.83	158.38
1	197	3967.30	271.48	11638.30	153.78	5746.67	318.52
2	97	4499.47	451.61	12351.00	373.16	5705.17	185.89
2	197	3930.37	408.76	11652.70	113.07	5739.17	101.73
3	97	4866.83	797.98	12009.10	543.19	5724.33	192.69
3	197	4324.27	250.05	11627.70	118.08	5874.83	234.96
4	97	3059.10	35.62	**10997.30**	**34.40**	**5493.83**	**171.54**
4	197	3126.37	43.98	11190.90	43.44	5853.33	277.29
5	97	**3018.80**	**36.10**	**10991.90**	**39.18**	**5501.33**	**134.55**
5	197	3116.70	54.22	11183.30	46.86	5775.50	282.53
6	97	3055.73	20.89	**10982.10**	**35.41**	**5448.33**	**116.53**
Human Planner		22100		-			

Table 14 Hospital 1 Meal Preperation Results. A. Setting, N. Set, T. Limit, St. Dev. denote algorithm setting, neighbourhood set, upper bound for the tabu list length, and standard deviation respectively.

A. Setting		Normal		Overload		Absence	
N. Set	T. Limit	Average	St. Dev	Average	St. Dev	Average	St. Dev
1	97	**5145.50**	**237.13**	12422.83	381.85	**8999.83**	**282.10**
1	197	7526.00	1204.73	15804.33	982.08	**9326.33**	**231.33**
2	97	**5233.33**	**238.56**	12499.00	380.78	**9347.83**	**355.24**
2	197	7510.17	1026.12	14602.50	1622.13	**9112.00**	**386.31**
3	97	**5228.00**	**425.58**	12622.83	526.01	9684.50	311.04
3	197	11242.50	1328.49	19617.17	1341.32	9689.83	467.05
4	97	**5207.83**	**181.26**	**11484.50**	**398.61**	**9077.33**	**397.37**
4	197	6103.17	389.54	12783.17	756.22	**9138.33**	**315.69**
5	97	**5132.17**	**293.30**	11841.00	298.61	**9171.83**	**296.10**
5	197	8613.17	854.50	15191.50	898.16	**9332.83**	**348.94**
6	97	**5263.83**	**234.20**	**11487.33**	**331.37**	**9349.17**	**401.30**
Human Planner		28594		-			

Table 15 Hospital 1 Geriatrics Results. A. Setting, N. Set, T. Limit, St. Dev. denote algorithm setting, neighbourhood set, upper bound for the tabu list length, and standard deviation respectively.

5 Conclusions

A new nurse rostering model is proposed to accurately represent the current situation of the problem in real world environments. This model is also open to extensions with extra soft constraints encountered in different hospitals, other sectors and different countries. To cope with the new properties of the problem, associated neighbourhoods are defined and utilised within a VNS algorithm. These neighbourhoods exploit the problem properties like compatible shift types and secondary skill types.

Experimental results show that different settings of the solution method perform better on different scenarios. This performance variation favours a selection between two neighbourhood sets, set 4 and 5, based on the fact whether the nurses involved in the problem instance have secondary skill types or not. A solution method that chooses between neighbourhood settings 4 and 5 based on this fact is implemented.

A. Setting		Normal		Overload		Absence	
N. Set	T. Limit	Average	St. Dev	Average	St. Dev	Average	St. Dev
1	97	56750.75	7034.01	66878.25	4057.00	57017.00	342.21
1	197	**51824.50**	**2645.01**	62331.75	1761.92	57217.50	412.21
2	97	66209.50	7569.78	72101.25	5309.99	57445.50	440.71
2	197	55831.75	4588.94	65921.50	2701.34	57606.00	541.99
3	97	56733.50	5115.32	66927.50	5272.66	57234.50	287.08
3	197	**50723.50**	**1672.81**	61620.75	2091.52	57220.00	534.12
4	97	**50614.75**	**2379.90**	**52050.00**	**1041.03**	56791.00	478.33
4	197	**50449.25**	**1271.19**	53096.50	1109.61	56829.50	477.89
5	97	**51184.00**	**2054.13**	**51217.50**	**685.19**	56728.00	527.09
5	197	**50201.25**	**1137.85**	53074.75	726.67	56738.00	466.16
6	97	**50613.25**	**1799.34**	**51404.25**	**983.97**	56626.50	557.74
Human Planner		183859		-			

Table 16 Hospital 2 Palliative Care. A. Setting, N. Set, T. Limit, St. Dev. denote algorithm setting, neighbourhood set, upper bound for the tabu list length, and standard deviation respectively.

Experimental results show that this solution method is among the best performing neighbourhood sets on most of the ward-scenario couples.

The exploitation of the extendibility of the model by adding extra soft constraints and other properties to cover real world problems encountered in other hospitals, sectors and different countries, is a natural future research direction. The application of further optimisation techniques, like hyperheuristics, can also be investigated. In this case heuristics that are specialised to tackle different aspects of the scenarios can be studied and deployed within hyperheuristics.

Acknowledgements This research project was funded by IWT (Institute for the Promotion of Innovation by Science and Technology in Flanders) within the projects IWT 060376, and IWT 060353 (RAP: Framework for Automated Personnel Rostering).

References

1. Aickelin, U., Li, J.: An estimation of distribution algorithm for nurse scheduling. Annals of Operations Research **155**(1), 289–309 (2007)
2. Bard, J.F., Purnomo, H.W.: Cyclic preference scheduling of nurses using a lagrangian-based heuristic. Journal of Scheduling **10**(1), 5–23 (2007)
3. Beliën, J., Demeulemeester, E.: On the trade-off between staff-decomposed and activity-decomposed column generation for a staff scheduling problem. Annals of Operations Research **155**(1), 143–166 (2007)
4. Bellanti, F., Carello, G., Della Croce, F., Tadei, R.: A greedy-based neighborhood search approach to a nurse rostering problem. European Journal of Operational Research **127**(1), 28–40 (2004)
5. Bilgin, B.: Project web page of automation of nurse rostering in Belgian hospitals. http://allserv.kahosl.be/~burak/project.html (2008)
6. Bourdais, S., Galinier, P., Pesant, G.: HIBISCUS: A constraint programming application to staff scheduling in health care. Lecture Notes in Computer Science **2833**, 153–167 (2003)
7. Brucker, P., Qu, R., Burke, E.K., Post, G.: A decomposition, construction and post-processing approach for a specific nurse rostering problem. In: Proceedings of Multidisciplinary International Conference on Scheduling : Theory and Applications, Aug, 2005, New York, pp. 397–406 (2005)

8. Burke, E.K., Cowling, P.I., De Causmaecker, P., Vanden Berghe, G.: A memetic approach to the nurse rostering problem. Applied Intelligence **15**(3), 199–214 (2001)
9. Burke, E.K., Curtois, T., Post, G., Qu, R., Veltman, B.: A hybrid heuristic ordering and variable neighbourhood search for the nurse rostering problem. European Journal of Operational Research **188**(2), 330–341 (2008)
10. Burke, E.K., Kendall, G., Soubeiga, E.: A tabu-search hyperheuristic for timetabling and rostering. Journal of Heuristics **9**(6), 451–470 (2003)
11. Burke, E.K., De Causmaecker, P., Vanden Berghe, G.: Handbook of Scheduling: Algorithms, Models and Performance Analysis, chap. Novel Metaheuristic Approaches to Nurse Rostering Problems in Belgian Hospitals, pp. 44.1–44.18. CRC Press (2004)
12. Burke, E.K., De Causmaecker, P., Vanden Berghe, G., Landeghem, H.V.: The State of the Art of Nurse Rostering. Journal of Scheduling **7**(6), 441–499 (2004)
13. Burke, E.K., De Causmaecker, P., Petrovic, S., Vanden Berghe, G.: Fitness evaluation for nurse scheduling problems. In: Proceedings of the 2001 Congress on Evolutionary Computation CEC2001, pp. 1139–1146. IEEE Press (2001)
14. Hansen, P., Mladenović, N.: Handbook of Metaheuristics, chap. Variable Neighborhood Search, pp. 145–184. Springer (2003)
15. Maenhout, B., Vanhoucke, M.: Comparison and hybridization of crossover operators for the nurse scheduling problem. Annals of Operations Research (2007)
16. Maenhout, B., Vanhoucke, M.: An electromagnetic meta-heuristic for the nurse scheduling problem. Journal of Heuristics **13**(4), 359–385 (2007)
17. De Causmaecker, P., Vanden Berghe, G.: Relaxation of coverage constraints in hospital personnel rostering. Lecture Notes in Computer Science **2740**, 129–147 (2003)
18. Özcan, E.: Memes, self-generation and nurse rostering. Lecture Notes in Computer Science **3867**, 85–104 (2007)
19. Schaerf, A., Di Gaspero, L.: Measurability and reproducibility in university timetabling research: Discussion and proposals. Lecture Notes in Computer Science **3867**, 40–49 (2007)