New Computational Results for the Nurse Scheduling Problem: A Scatter Search Algorithm

Broos Maenhout[1] and Mario Vanhoucke[1,2]

[1] Faculty of Economics and Business Administration,
Ghent University, Gent, Belgium
[2] Operations & Technology Management Centre,
Vlerick Leuven Gent Management School, Gent, Belgium
`{broos.maenhout, mario.vanhoucke}@ugent.be`

Abstract. In this paper, we present a scatter search algorithm for the well-known nurse scheduling problem (NSP). This problem aims at the construction of roster schedules for nurses taking both hard and soft constraints into account. The objective is to minimize the total preference cost of the nurses and the total penalty cost from violations of the soft constraints. The problem is known to be NP-hard. The contribution of this paper is threefold. First, we are, to the best of our knowledge, the first to present a scatter search algorithm for the NSP. Second, we investigate two different types of solution combination methods in the scatter search framework, based on four different cost elements. Last, we present detailed computational experiments on a benchmark dataset presented recently, and solve these problem instances under different assumptions. We show that our procedure performs consistently well under many different circumstances, and hence, can be considered as robust against case-specific constraints.

Keywords: meta-heuristics; scatter search; nurse scheduling.

1 Introduction

Personnel scheduling problems are encountered in many application areas, such as public services, call centers, hospitals, and industry in general. For most of these organizations, the ability to have suitably qualified staff on duty at the right time is of critical importance when attempting to satisfy their customers' requirements and is frequently a large determinant of service organization efficiency [1,2]. This explains the broad attention given in literature to a great variety of personnel rostering applications [3,4]. In general, personnel scheduling is the process of constructing occupation timetables for staff to meet a time-dependent demand for different services while encountering specific workplace agreements and attempting to satisfy individual work preferences. The particular characteristics of different industries result in quite diverse rostering models which leads to the application of very different solution techniques to solve these models. Typically, personnel scheduling problems are highly constrained and complex optimization problems [4,5].

In this paper, a procedure is presented to solve the nurse scheduling problem (NSP) which involves the construction of duty rosters for nursing staff over a pre-defined period. Problem descriptions and models vary drastically and depend on the characteristics and policies of the particular hospital. Due to the huge variety of hard and soft constraints, and the several objective function possibilities, the nurse scheduling problem has a multitude of representations, and hence, many exact and heuristic procedures have been proposed to solve the NSP in various guises. Recent literature surveys [6,7] give an overview of all these procedures, and mention simulated annealing, tabu search and genetic algorithms as popular meta-heuristics for the NSP.

In constructing a nurse schedule, a set of nurses need to be assigned to days and shifts in order to meet the minimal coverage constraints and other case-specific constraints and to maximize the quality of the constructed timetable. Consequently, the NSP under study is acquainted with three main components, i.e.

- Each nurse needs to express his/her *preferences* as the aversion to work on a particular day and shift. According to [8], quantifying these nurses' preferences in the objective function maintains fairness and guarantees the quality of the nurse roster over the scheduling horizon.
- The *minimal coverage constraints* embody the minimal required nurses per shift and per day, and are inherent to any shift scheduling problem. These constraints are handled as soft constraints that can be violated at a certain penalty cost expressed in the objective function.
- The *case-specific constraints* are not inherent to any NSP instance but are rather *case-specific*, i.e. determined by personal time requirements, specific workplace conditions, national legislation, etc…. These constraints are handled as hard constraints which can not be violated at a certain penalty cost.

Hence, the objective of the NSP is to minimize the sum of the nurses' preferences and the total penalty cost of violating the minimal coverage requirements subject to different case-specific (hard) constraints. For a more formal description of the NSP, we refer to [9]. The problem is known to be NP-hard [10].

In section 2 of this paper, we briefly review the philosophy of the scatter search template provided by [11]. Moreover, we discuss and illustrate the underlying principles and the implementation of the scatter search framework for the nurse scheduling problem. In section 3, we present new computational results tested on the NSPLib dataset proposed by [12]. In section 4, conclusions are made and directions for future research are indicated.

2   Scatter Search for NSP

Scatter search [11] is a population-based meta-heuristic in which solutions are combined to yield better solutions using convex linear or non-linear combinations. This evolutionary meta-heuristic differs from other evolutionary approaches, such as genetic algorithms, by providing unifying principles for joining solutions based on generalized path constructions in Euclidian space and by utilizing strategic designs where other approaches resort to randomization. The scatter search methodology is

very flexible, since each of its elements can be implemented in a variety of ways and degrees of sophistication. Hence, the scatter search template has been successfully applied in several application areas. However, to the best of our knowledge, the scatter search framework has been applied only once to personnel rostering, more precisely on a labour scheduling problem by [13]. In their paper, they describe the development and implementation of a decision support system for the optimization of the passenger flow by trading off service quality and labour costs at an airport. In their search for the minimal number of employees, their path relinking approach concentrates on the shifts which are staffed differently in the parent solutions.

For an overview of the basic and advanced features of the scatter search meta-heuristic, we refer to [14,15]. In the following, we describe our implementation of the scatter search approach to the nurse scheduling problem. The pseudo-code for our generic scatter search template to solve the NSP is written below.

```
Algorithm Scatter Search NSP
  Diversification Generation Method
  While Stop Criterion not met
    Subset Generation Method
    Subset Combination Method
    Improvement Method
    Reference Set Update Method
  Endwhile
```

The Diversification Generation Method. In this initialization step, a large pool of P initial solution vectors is generated. A solution point is encoded by the nurse-day representation [6] which basically indicates the assignment of each nurse to a shift on each day. Since useful information about the structure of optimal solutions is typically contained in a suitably diverse collection of elite solutions, the initial solutions are generated in such a manner a critical level of diversity is guaranteed [14,15]. In order to generate a diverse set of initial solutions, we create x solutions using a constructive heuristic and $P - x$ solutions in a random way. The constructive heuristic schedules the nurses in a random sequence taking both preference costs and penalty cost of violating the coverage constraints into account. This greedy heuristic is conceived as a minimum cost flow problem which represents all shifts on all days to which a particular nurse can be assigned to. Since not all case-specific constraints (e.g. the maximum number of assignments) can be modelled in the network, it has been implemented by a k-shortest path approach. Based on this initial population, a subset of the population elements are designated to be reference solutions. This reference set contains b_1 high quality solutions ($Refset_1$) and b_2 diverse solutions ($Refset_2$). The construction of $Refset_1$ starts with the selection of the best b_1 solutions in terms of solution quality out of the P initial solutions. In order to select the diverse solutions ($Refset_2$), the minimum distance between all remaining $P - b_1$ solutions and the b_1 solutions is calculated based on the adjacency degree of [16]. In pursuit of diversity, the b_2 solutions with maximal distance will be selected for membership of $Refset_2$ while all other $P - b_1 - b_2$ solutions are disregarded.

The Solution Generation Method. After the initialization phase, scatter search operates on this reference set by combining pairs of reference solutions in a

controlled, structured way. Two elements of the reference set are chosen in a systematic way to produce points both inside and outside the convex regions spanned by the reference solutions. [14] suggest to create new solutions out of all two-element subsets. Choosing the two reference solutions out of the same cluster stimulates intensification, while choosing them from different clusters stimulates diversification. Hence, in our scatter search, the solution method consists of the evaluation of all $b_1 \times b_1$, $b_1 \times b_2$ and $b_2 \times b_2$ combinations in a random sequence.

The Solution Combination Method. A new solution point is the result of a linear combination of two population elements in the reference set. The process of generating linear combinations of a set of reference solutions may be characterized as generating paths between solutions [17]. A path between solutions will generally yield new solutions that share a significant subset of attributes contained in both parent solutions, which can differ according to the path selected. The moves introduce attributes contributed by a guiding solution and/or reduce the distance between the initiating and the guiding solution. The goal is to capture the assignments that frequently or significantly occur in high quality solutions, and then to introduce some of these compatible assignments into other solutions that are generated by a heuristic combination mechanism.

Our specific combination method relies on problem-specific information of both the initiating and the guiding solution schedule to create a new schedule, and takes four criteria into account. Two of these criteria incorporate objective function related data into account, as follows:

- Preference costs: Since the overall objective is to minimize the nurses' aversion towards the constructed work timetable, the day/shift preference cost expressed by each nurse is an important determinant in constructing high-quality new solution points.
- Coverage information: In order to minimize the penalty cost of violating the minimal coverage constraints, the algorithm penalizes those shifts where the coverage constraints are violated. In doing so, the solution combination method biases the initiating solution towards a (more) feasible solution.

The other two criteria incorporate information to maintain the "good" characteristics of both the initiating and the guiding schedule, as follows:

- Critical shifts of the initiating solution: In directing the initiating solution towards the guiding solution, the algorithm prevents the removal of critical shifts from the initiating solution, which, in case of removal, would lead to an additional violation of the coverage constraints. In doing so, the algorithm aims at the construction of a new solution point that does not encounter any (additional) violations of the coverage constraints.
- Bias to the guiding solution: The algorithm guides the initiating solution to the assignments of the guiding solution, in order to decrease the distance between the two schedules by introducing attributes of the guiding solution.
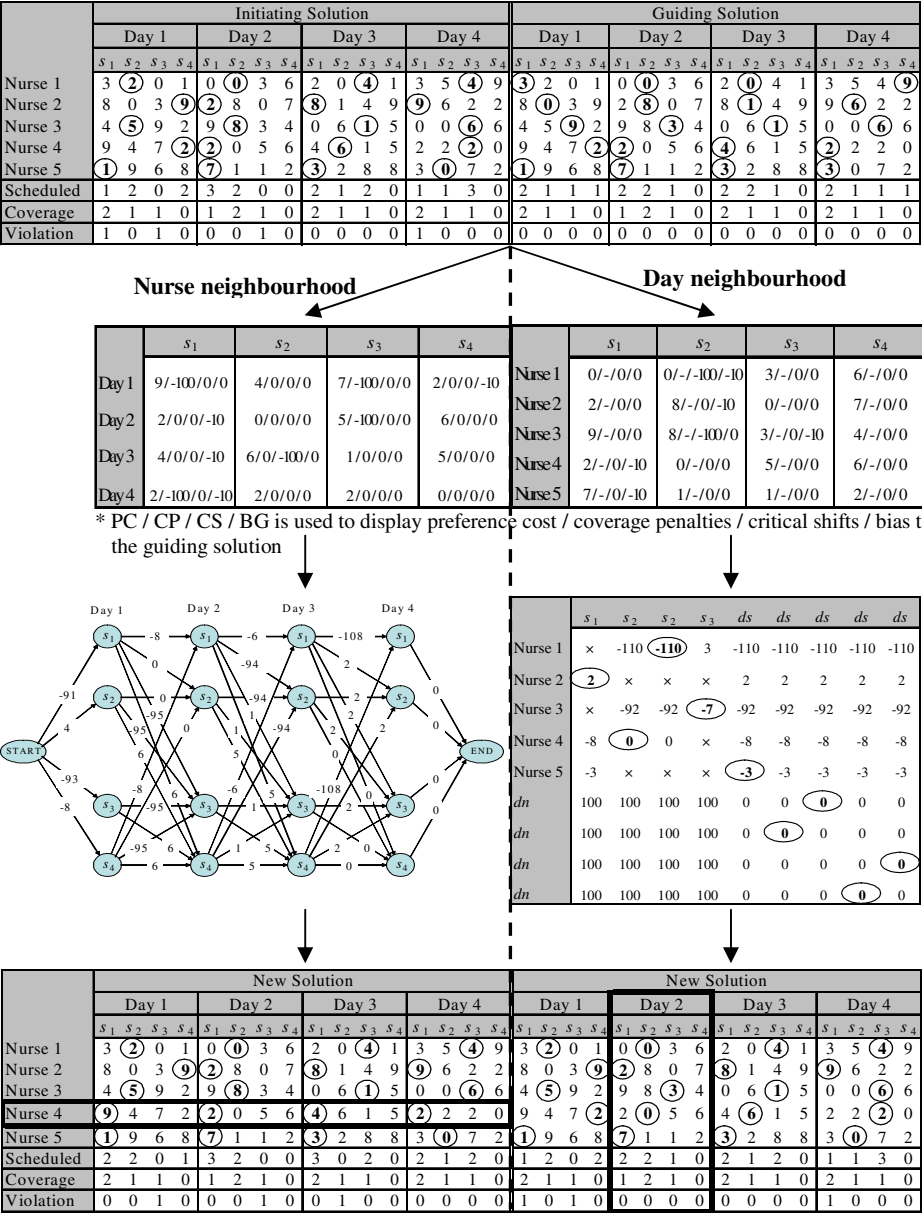
These four elements will be carefully taken into account for each move from an initiating solution to a guiding solution. Since the re-linking process of two solutions out of the reference set can be based on more than one neighbourhood [15], our algorithm makes use of two types of neighbourhood moves: a nurse neighbourhood move or a day neighbourhood move.

In the *nurse neighbourhood move*, the schedule of a single nurse of the initiating solution is directed towards the schedule of the corresponding nurse in the guiding solution. Therefore, the algorithm relies on a k-shortest path approach to optimize the schedule for a particular nurse taking into account a weighted average of the four aforementioned elements. The scheduling of a single nurse at minimum cost over the complete scheduling horizon can be considered as a minimum cost flow problem and can be solved by any shortest path algorithm. Moreover, since not all case-specific constraints can be incorporated in a shortest path algorithm, a k-shortest path approach is implemented where the outcome of this algorithm (i.e. a nurse schedule) should be checked whether it is feasible or not with respect to all these constraints. If the outcome is not feasible, a 2nd shortest path will be generated and checked for feasibility. This process continues until the shortest feasible pattern (i.e. the kth shortest path) for the nurse is found. The graph used for our algorithm consists of #days*#shifts nodes (plus two extra dummy nodes representing the start and end of the network) representing the daily shift assignments for the nurse under study. An arc (a, b) is drawn to connect node a representing a possible shift assignment on day j to node b representing a shift assignment on day $j + 1$. The distances between nodes are made up of a weighted average of the four abovementioned elements. In calculating the new schedule for a particular nurse, we rely on the algorithm of [18].

In the *day neighbourhood move*, the roster of a single day of the initiating solution is directed towards the roster of the corresponding day of the guiding solution, given the assignments of the nurses on all other days in the initiating solution. To that purpose, we transform a single day roster to a linear assignment problem (LAP) matrix, and solve it by means of the Hungarian method [19]. In constructing this matrix, we duplicate each shift column such that each shift has a number of columns that is equal to its coverage requirements. Moreover, we add dummy nurses and/or dummy shifts to allow under- or over-coverage of the coverage requirements. The number of extra dummy nurses equals the total daily nurse requirements, and penalizes under-coverage when a required shift column has to be assigned to a dummy nurse. The number of extra dummy shifts equals the total number of (non-dummy) nurses and allows over-coverage of the coverage requirements when nurses are assigned to dummy shifts. The cost of assigning non-dummy nurses to one of the dummy shifts is equal to the minimum cost of the (feasible) shifts a nurse can be assigned to. The LAP matrix contains costs associated with the four criteria the solution combination mechanism is based on. Furthermore, the LAP matrix excludes certain shift assignments to cope with the case-specific constraints, taking into account the fixed assignments of all other days of the current solution.

This solution combination method can be best illustrated on an example NSP instance with 5 nurses and a scheduling period of 4 days. We assume that each day consists of three working shifts (e.g. early (s_1), day (s_2), night (s_3)) and a free shift (s_4). The nurses' preferences as well as the minimal coverage requirements are displayed in the top table of figure 1. Since "s_4" is used to refer to a free shift, its daily coverage requirements equal zero. We assume some additional case specific constraints as follows: the number of assignments varies between a minimal value of 3 and a maximal value of 4. The consecutive working shifts vary between a

minimal value of 2 and a maximal value of 4. The assignment of nurses to maximal one shift per day and the succession constraints are inherent to continuous personnel scheduling. The latter constraint implies forbidden successive assignments between s_3 and s_1, s_3 and s_2 and s_2 and s_1.

Initiating Solution / Guiding Solution

	Initiating Solution																Guiding Solution															
	Day 1				Day 2				Day 3				Day 4				Day 1				Day 2				Day 3				Day 4			
	s_1	s_2	s_3	s_4	s_1	s_2	s_3	s_4	s_1	s_2	s_3	s_4	s_1	s_2	s_3	s_4	s_1	s_2	s_3	s_4	s_1	s_2	s_3	s_4	s_1	s_2	s_3	s_4	s_1	s_2	s_3	s_4
Nurse 1	3	(2)	0	1	0	(0)	3	6	2	0	(4)	1	3	5	(4)	9	(3)	2	0	1	0	(0)	3	6	2	(0)	4	1	3	5	4	(9)
Nurse 2	8	0	3	(9)	(2)	8	0	7	(8)	1	4	9	(9)	6	2	2	8	(0)	3	9	2	(8)	0	7	8	(1)	4	9	9	(6)	2	2
Nurse 3	4	(5)	9	2	9	(8)	3	4	0	6	(1)	5	0	0	(6)	6	4	5	(9)	2	9	8	(3)	4	0	6	(1)	5	0	0	(6)	6
Nurse 4	9	4	7	(2)	(2)	0	5	6	4	(6)	1	5	2	2	(2)	0	9	4	7	(2)	(2)	0	5	6	(4)	6	1	5	(2)	2	2	0
Nurse 5	(1)	9	6	8	(7)	1	1	2	(3)	2	8	8	3	(0)	7	2	(1)	9	6	8	(7)	1	1	2	(3)	2	8	8	(3)	0	7	2
Scheduled	1	2	0	2	3	2	0	0	2	1	2	0	1	1	3	0	2	1	1	1	2	2	1	0	2	2	1	0	2	1	1	1
Coverage	2	1	1	0	1	2	1	0	2	1	1	0	2	1	1	0	2	1	1	0	1	2	1	0	2	1	1	0	2	1	1	0
Violation	1	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Nurse neighbourhood / **Day neighbourhood**

Nurse neighbourhood	s_1	s_2	s_3	s_4
Day 1	9/-100/0/0	4/0/0/0	7/-100/0/0	2/0/0/-10
Day 2	2/0/0/-10	0/0/0/0	5/-100/0/0	6/0/0/0
Day 3	4/0/0/-10	6/0/-100/0	1/0/0/0	5/0/0/0
Day 4	2/-100/0/-10	2/0/0/0	2/0/0/0	0/0/0/0

Day neighbourhood	s_1	s_2	s_3	s_4
Nurse 1	0/-/0/0	0/-/-100/-10	3/-/0/0	6/-/0/0
Nurse 2	2/-/0/0	8/-/0/-10	0/-/0/0	7/-/0/0
Nurse 3	9/-/0/0	8/-/-100/0	3/-/0/-10	4/-/0/0
Nurse 4	2/-/0/-10	0/-/0/0	5/-/0/0	6/-/0/0
Nurse 5	7/-/0/-10	1/-/0/0	1/-/0/0	2/-/0/0

* PC / CP / CS / BG is used to display preference cost / coverage penalties / critical shifts / bias to the guiding solution

	s_1	s_2	s_2	s_3	ds	ds	ds	ds	ds
Nurse 1	×	-110	(-110)	3	-110	-110	-110	-110	-110
Nurse 2	(2)	×	×	×	2	2	2	2	2
Nurse 3	×	-92	-92	(-7)	-92	-92	-92	-92	-92
Nurse 4	-8	(0)	0	×	-8	-8	-8	-8	-8
Nurse 5	-3	×	×	×	(-3)	-3	-3	-3	-3
dn	100	100	100	100	0	0	(0)	0	0
dn	100	100	100	100	0	(0)	0	0	0
dn	100	100	100	100	0	0	0	0	(0)
dn	100	100	100	100	0	0	0	(0)	0

New Solution / New Solution

	New Solution																New Solution															
	Day 1				Day 2				Day 3				Day 4				Day 1				Day 2				Day 3				Day 4			
	s_1	s_2	s_3	s_4	s_1	s_2	s_3	s_4	s_1	s_2	s_3	s_4	s_1	s_2	s_3	s_4	s_1	s_2	s_3	s_4	s_1	s_2	s_3	s_4	s_1	s_2	s_3	s_4	s_1	s_2	s_3	s_4
Nurse 1	3	(2)	0	1	0	(0)	3	6	2	0	(4)	1	3	5	(4)	9	3	(2)	0	1	0	(0)	3	6	2	0	(4)	1	3	5	(4)	9
Nurse 2	8	0	3	(9)	(2)	8	0	7	(8)	1	4	9	(9)	6	2	2	8	0	3	(9)	(2)	8	0	7	(8)	1	4	9	(9)	6	2	2
Nurse 3	4	(5)	9	2	9	(8)	3	4	0	6	(1)	5	0	0	(6)	6	4	(5)	9	2	9	8	(3)	4	0	6	(1)	5	0	0	(6)	6
Nurse 4	(9)	4	7	2	(2)	0	5	6	(4)	6	1	5	2	2	(2)	0	9	4	7	(2)	2	(0)	5	6	4	(6)	1	5	2	2	(2)	0
Nurse 5	(1)	9	6	8	(7)	1	1	2	(3)	2	8	8	3	(0)	7	2	(1)	9	6	8	(7)	1	1	2	(3)	2	8	8	3	(0)	7	2
Scheduled	2	2	0	1	3	2	0	0	3	0	2	0	2	1	2	0	1	2	0	2	2	2	1	0	2	1	2	0	1	1	3	0
Coverage	2	1	1	0	1	2	1	0	2	1	1	0	2	1	1	0	2	1	1	0	1	2	1	0	2	1	1	0	2	1	1	0
Violation	0	0	1	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0	0	1	0	0	0

Fig. 1. The Solution Combination Method in nurse and day neighbourhood space

The left top table is assumed to be the initiating solution from $Refset_2$ with a total preference cost of 81 and four coverage violations (the specific assignments have been encircled). Since the algorithm penalizes each coverage violation with a penalty cost of 100, the total solution quality of the initiating solution equals 481. The right table is assumed to be the guiding solution from $Refset_1$ with a total preference cost of 70 and no coverage penalties. The adjacency degree measures the distance between schedules as the sum of zeros (identical day/shift assignment) and ones (different day/shift assignment), which leads to a total distance of 12.

In the remainder of this section, we illustrate the nurse neighbourhood move for nurse 4 (left part) and the day neighbourhood move for day 2 (right part) on our two parent solution schedules.

Nurse neighbourhood move: The left table below the parent solutions displays the calculations of the four elements of this section, i.e. the nurse's preference costs, the coverage penalties, the critical shifts and the assigned shifts of the guiding solution. The coverage penalty is set at 100, while the assigned shifts of the guiding solution have a negative cost of 10. The critical shifts are found for those assigned shifts of the initial schedule of nurse 4 when the difference between the number of scheduled nurses (row 'Scheduled') and the assignments of nurse 4 is lower than the minimum required number of nurses (row 'Coverage').

The sum of all costs in the table results in the cost values on the corresponding arcs of the network representing the scheduling of the nurse over the complete scheduling horizon. The graph counts 4 * 4 nodes and a start and an end dummy node. The shortest path in the network is $s_3 - s_3 - s_4 - s_1$ with a distance of -291. However, the path is infeasible since the constraint of minimal 2 consecutive working days is violated. Based on the same argument, the 2nd shortest path, i.e. $s_1 - s_3 - s_4 - s_1$ with a distance of -289, is infeasible. The next shortest path is $s_1 - s_1 - s_1 - s_1$ with a distance of -213. This path is feasible for all case-specific constraints and leads to the newly constructed schedule at the left bottom part of the figure. The new solution point has a total solution quality of 386. The total preference cost has increased from 81 to 86, whereas the number of coverage violations has decreased from 4 to 3. The distance between the new initiating solution point and the guiding solution point has also decreased from 12 to 11.

Day neighbourhood move: The right table below the parent solutions displays the calculations of the four elements in a similar way as previously, but from the second day's point-of-view. The corresponding LAP matrix contains the sum of three of these four elements, i.e. the nurse's preference costs, the total critical shift cost and the cost of the assigned shifts of the guiding solutions. The coverage penalties have been incorporated implicitly in the structure of the LAP matrix since dummy nurses (*dn*) have been inserted which penalize the under-coverage of shifts. In contrast, the incorporation of dummy shifts (*ds*) allows the over-coverage of shifts. The case-specific constraints have been embedded in the LAP matrix by excluding some assignments (denoted by crossed cells). The optimal LAP solution has been encircled and leads to the newly constructed schedule at the right bottom part of the figure. The new solution point has a total solution quality of 374. The total preference cost has decreased from 81 to 74, and the number of coverage violations has also decreased

from 4 to 3. The distance between the new initiating solution point and the guiding solution point has not changed and remains 12.

The Improvement Method. The improvement method applies heuristic processes to improve both total preference cost and coverage infeasibilities of the newly generated solution points. To that purpose, we implemented the three complementary local search algorithms of [9], each focusing on a different part of the scheduling matrix. The *pattern-based local search* aims at the optimization of the schedule for a particular nurse, given the schedules of all other nurses. The *day-based local search* optimizes the schedule for one day given the assignments of the nurses on all other days. The *schedule-based local search* aims to improve the quality of the schedule by swapping (parts of) schedules between nurses.

The Reference Update Method. After the application of the diversification and intensification process, the child solutions are added to the reference set if certain threshold values for the criteria which evaluate the merit of newly created solution points are met. A newly generated solution may become a member of the reference set either if the new solution has a better objective function value than the solution with the worst objective value in $Refset_1$ or if the diversity of the new solution with respect to the reference set is larger than the solution with the smallest distance value in $Refset_2$. In both cases the new solution replaces the worst and the ranking is updated to identify the new worst solution in terms of either quality or diversity. The reference set is dynamically updated. In contrast to a static update where the reference set is updated after combination of all generated sub-sets, a dynamic update evaluates each possible reference set entrance instantly. In this way, new "best" solutions can be combined faster and inferior solutions are eliminated faster. During the search, diversity in the reference set is maintained through the use of these artificial tiers in the reference set but also through a threshold distance depending on the problem size under study. The latter prevents the duplication of solution points in the reference set and/or the entrance of highly resembling solutions.

3   Computational Results

In this section, we present computational results for our scatter search procedure tested on the NSPLib problem instances of [12]. This testset contains 4 sub-sets with 25, 50, 75, and 100 nurses and a 7-days scheduling horizon (this so-called *diverse set* contains 4 * 7290 instances), and 2 sub-sets with 30 or 60 nurses and a 28-days scheduling horizon (this *realistic set* contains 2 * 960 instances). The nurses' preference structure and the coverage requirements of each sub-set are characterized by systematically varied levels of various NSP complexity indicators proposed in [12]. All sets have been extended by 8 mixes of case-specific constraints which appear frequently in literature [6], i.e. the minimum/maximum number of working assignments, the minimum/maximum number of assignments per shift type, the minimum/maximum consecutive working shifts, and the minimum/maximum consecutive working shifts per shift type. The testset, the sets of case-specific constraints and results can be downloaded from http://www. projectmanagement. ugent.be/nsp.php. The tests have been carried out under a stop criterion of 1,000 or

5,000 schedules (Toshiba SPA10 2.4Ghz processor). In the next sub-section, we compare different neighbourhood combination versions into detail. In section 3.2, we present best known solutions for our large dataset.

3.1 Day Neighbourhood or Nurse Neighbourhood

In order to test the performance of our two solution combination methods, the day neighbourhood (DNH) and nurse neighbourhood (NNH), we have randomly selected 576 instances from the 25- and 288 instances from the 30- nurse instances. Each neighbourhood move (day or nurse) contains a mix of the 4 elements, i.e. preference cost (PC), coverage penalty (CP), critical shift calculation (CS) and the bias to the guiding solution (BG). Both the nurse and the day neighbourhood moves will be compared with simple and straightforward moves, based on

- Complete replacement by the guiding solution (CGS): The day (DNH) or nurse (NNH) of the initiating solution will be completely replaced by the assignments of the guiding solution.
- Random cost (RAN): The cost matrix simply contains random numbers instead of the sum of PC, CP, CS and BG.
- Percentage of guiding solution (%GS): A randomly selected part of the day (DNH) or nurse (NNH) of the initiating solution will be replaced by the assignments of the guiding solution.

Table 1. Average solution quality (Avg_Sol) and Ranking for the various combination methods

	Combination elements				Nurse Neighbourhood				Day Neighbourhood			
					N25		N30		N25		N30	
	PC	CP	CS	BG	Avg_Sol	Ranking	Avg_Sol	Ranking	Avg_Sol	Ranking	Avg_Sol	Ranking
Solution Combination Method	×	-	-	-	310.67	17	1,498.21	9	307.52	14	1,500.20	15
	-	×	-	-	312.26	18	1,528.22	18	307.37	11	1,501.21	16
	-	-	×	-	308.43	9	1,507.74	11	307.86	18	1,498.72	13
	-	-	-	×	306.67	5	1,496.02	6	307.00	5	1,495.93	9
	×	×	-	-	310.25	14	1,515.23	13	307.57	16	1,542.96	18
	×	-	×	-	309.18	11	1,496.80	7	307.67	17	1,498.66	12
	×	**-**	**-**	**×**	**306.66**	**4**	**1,493.21**	**4**	307.00	5	1,495.16	5
	-	×	×	-	310.40	15	1,519.58	15	307.50	13	1,498.87	14
	-	×	-	×	309.37	13	1,519.87	16	307.03	7	1,494.11	3
	-	**-**	**×**	**×**	**306.57**	**3**	**1,492.17**	**3**	307.27	10	1,495.43	7
	×	×	×	-	308.39	8	1,509.86	12	307.44	12	1,497.53	11
	×	×	-	×	309.26	12	1,518.92	14	306.97	4	1,496.35	10
	×	**-**	**×**	**×**	**305.82**	**1**	**1,490.34**	**2**	307.19	8	1,495.65	8
	-	×	×	×	308.99	10	1,494.23	5	307.21	9	1,495.24	6
	×	**×**	**×**	**×**	**306.20**	**2**	**1,486.36**	**1**	306.94	3	1,493.77	2
CGS	-	-	-	-	306.68	6	1,497.63	8	306.14	1	1,492.97	1
RAN	-	-	-	-	310.56	16	1,524.50	17	307.55	15	1,504.30	17
%GS	-	-	-	-	307.01	7	1,498.63	10	306.71	2	1,494.71	4

The results displayed in table 1 can be summarized as follows. First, the day-based neighbourhood (DNH) outperforms, on the average, the nurse-based neighbourhood (NNH). Indeed, 13 (10) of the 18 tests results in a DNH cost which is lower than its corresponding NNH cost. However, the best results can be obtained with the NNH method. The best result for the N25 (N30) instances amounts to 305.82 (1,486.36) which outperforms the best known results for the DNH method (306.94 and 1,492.97,

respectively). Second, the top 4 results for the NNH have been displayed in bold, and show that the three elements, PC, CS and BG are relevant cost factors (3, 3 and 4 times used, respectively). The CP cost factor has been used only once in the top 4 results, and seems to be less important. Last, the simple CGS approach is the best approach for the DNH method, but is not able to outperform the best known results with the NNH method. The %GS has an average (good) performance for the NNH (DNH) method while the RAN method leads to rather poor results for both solution combination methods.

In section 3.2, we report best known solutions for all the data instances, based on the solution combination method NNH-PC/CS/BG for the diverse set with a 7-days scheduling horizon (N25, N50, N75 and N100) and the solution method NNH-PC/CP/CS/BG for the realistic set with a 28-days scheduling horizon (N30 and N60).

3.2   New Best Known Solutions

In order to benchmark our results and present best known state-of-the-art solutions, we have tested all 31,080 instances on all case-specific constraint files, resulting in 248,640 instances in total. We have truncated each test after a stop criterion of 1,000 and 5,000 schedules. Table 2 displays the results for the 5,000 schedule stop criterion. A similar table can be downloaded from www.projectmanagement.ugent.be/nsp.php with a 1,000 schedule stop criterion. The table displays the average solution quality, split up in the average total preference cost (Avg_Pref) and the average penalty cost (Avg_Pen) which is calculated as the average number of violations of the minimal coverage requirements times the penalty cost of 100, the required CPU time (Avg_CPU), the percentage of files for which a feasible solution has been found (%Feas), and the percentage deviation from the LP optimal solution which serves as a lower bound (%Dev_LP). The latter has been found by a simple and straightforward LP model, and has been used for similar tests by [9]. No LP lower bounds could be provided for the N30 en N60 sets, since the number of constraints exceeds the limits for the industrial LINDO optimization library, version 5.3 [20].

In order to fine-tune a number of parameters, we have run our procedure on a small subset of all instances under different parameter settings. In order to find an appropriate balance between the diversification and intensification process, we have combined the three proposed local search heuristics into a variable neighbourhood search. For the N25 instances, all nurses are subject to the pattern-based local search and all days are subject to the day-based local search. The schedule-based local search evaluates possible swaps between whole schedules and two-day sub-schedules. In our tests, 40% of all two-day sub-schedules are swapped between the nurses for all testsets. The population size ($b_1 + b_2$) has been set to 15 (10) for the N25, N50 and N30 (N75, N100 and N60) instances, respectively. Each time, 80% of these population elements (b_1) have been put into the *Refset*$_1$. Moreover, 40% of all nurses are subject to the NNH move for N25 and N50 instances, 60% for the N75 instances, 20% for the N100 instances and all nurses for the N30 and N60 instances.

The table reveals that the gap between the obtained solutions and the LP based lower bounds is in many cases very small. For some of the cases, the gap is somewhat larger which gives an indication of the constrainedness of the problem instances. Our

scatter search algorithm is able to outperform (equalize) 40.5% and 62.06% (27.12% and 1.45%) of respectively the diverse and realistic instances solved by the electromagnetic procedure of [9] with a stop-criterion of 1,000 schedules, and 33.29% and 64.42% (45.02% and 1.74%) of respectively the diverse and realistic instances for the 5,000 schedules.

Table 2. Computational Results for the diverse and realistic dataset for 5,000 schedules

	Diverse set									
	N25					N50				
	Avg_Pref	Avg_Pen	%Feas	Avg_CPU	%Dev_LP	Avg_Pref	Avg_Pen	%Feas	Avg_CPU	%Dev_LP
Case 1	252.23	53.02	88.27%	1.61	0.19%	502.66	84.79	90.03%	5.33	0.25%
Case 2	240.86	53.02	88.27%	1.03	0.08%	480.42	84.79	90.03%	3.76	0.11%
Case 3	268.40	53.76	88.08%	2.31	1.11%	529.16	86.80	89.66%	6.38	1.17%
Case 4	250.33	53.02	88.27%	1.58	0.14%	499.12	84.79	90.03%	5.37	0.19%
Case 5	265.94	71.12	85.88%	2.26	3.63%	527.76	142.88	85.25%	6.97	3.44%
Case 6	241.86	53.02	88.27%	1.03	0.50%	482.65	84.79	90.03%	3.77	0.59%
Case 7	284.11	125.05	80.10%	5.05	5.95%	556.82	273.20	78.29%	12.61	5.33%
Case 8	259.21	71.89	85.60%	1.67	3.26%	513.02	140.01	85.50%	5.01	3.01%
	N75					N100				
	Avg_Pref	Avg_Pen	%Feas	Avg_CPU	%Dev_LP	Avg_Pref	Avg_Pen	%Feas	Avg_CPU	%Dev_LP
Case 1	762.77	150.32	88.70%	10.13	0.33%	1,223.44	166.49	90.49%	23.08	0.36%
Case 2	738.20	150.32	88.70%	10.78	0.15%	1,180.94	166.32	90.51%	21.16	0.20%
Case 3	802.67	152.43	88.37%	14.33	1.24%	1,299.15	170.40	90.01%	24.68	1.22%
Case 4	752.10	150.32	88.70%	9.83	0.23%	1,209.59	166.45	90.48%	22.09	0.28%
Case 5	802.08	202.88	86.06%	11.36	3.85%	1,280.83	260.23	86.28%	24.08	3.43%
Case 6	739.66	150.32	88.70%	8.51	0.17%	1,183.93	166.35	90.51%	21.22	0.22%
Case 7	848.60	367.17	79.48%	16.28	5.76%	1,354.75	517.49	79.49%	31.24	5.38%
Case 8	787.75	206.68	85.78%	9.42	4.30%	1,258.08	256.98	86.53%	22.11	4.11%
	Realistic set									
	N30					N60				
	Avg_Pref	Avg_Pen	%Feas	Avg_CPU	%Dev_LP	Avg_Pref	Avg_Pen	%Feas	Avg_CPU	%Dev_LP
Case 9	1,525.63	423.65	68.02%	17.37	-	3125.66	743.65	68.54%	37.99	-
Case 10	1,429.45	392.50	69.58%	7.52	-	2935.91	677.81	70.00%	24.01	-
Case 11	1,607.99	429.58	67.60%	40.68	-	3277.91	758.54	68.44%	73.00	-
Case 12	1,465.36	392.40	69.38%	8.95	-	3006.31	677.29	69.90%	26.10	-
Case 13	1,570.33	501.67	66.35%	17.52	-	3211.65	906.46	67.92%	37.66	-
Case 14	1,444.48	394.17	69.48%	7.64	-	2963.95	681.98	69.79%	24.04	-
Case 15	1,678.09	843.75	63.33%	59.68	-	3424.43	1543.33	66.04%	105.97	-
Case 16	1,541.34	498.02	66.88%	10.41	-	3154.49	882.92	68.23%	27.44	-

4   Conclusions and Future Research

In this paper, we have presented a new scatter search procedure for the well-known nurse scheduling problem. To the best of our knowledge, the literature on scatter search for the nurse scheduling problem is completely void. This framework has only been applied once on a similar problem type of labour scheduling by [13].

We have investigated the use of two types of solution combination methods, based on the combinations of sub-schedules of nurses or days. Each method calculates the attractiveness of the move based on four criteria. We have shown that the scatter search algorithm leads to promising results and hence might have a bright future in the further development of meta-heuristic optimization algorithms. We have tested our procedure on a generated problem set NSPLib, under a strict test design with a strict stop criterion to facilitate comparison between procedures.

Our main future research intention is as follows. We will aim at the development of hybrid versions of different meta-heuristics, based on knowledge and concepts presented in this and many other research papers. A skilled combination of concepts of different meta-heuristics can provide a more efficient behaviour and a higher flexibility when dealing with real-world and large-scale problems.

References

1. Thompson, G.M.: Improved implicit optimal modelling of the labour shift scheduling problem. Management Science 43 (1995) 595-607
2. Felici, G., and Gentile, C.: A polyhedral Approach for the Staff Rostering Problem. Management Science 50 (2004) 381-393
3. Ernst, A.T., Jiang, H., Krishamoorty, M., Owens, B., and Sier, D.: Staff scheduling and rostering: A review of applications, methods and models. European Journal of Operational Research 153 (2004) 3-27
4. Ernst, A.T., Jiang, H., Krishamoorty, M., Owens, B., and Sier, D.: An Annotated Bibliography of Personnel Scheduling and Rostering. Annals of Operations Research 127 (2004) 21-144
5. Glover, F., and McMillan, C.: The General Employee Scheduling Problem: An integration of MS and AI. Computers and Operations Research 13 (1986) 563-573
6. Cheang, B., Li, H., Lim, A., and Rodrigues, B.: Nurse rostering problems – a bibliographic survey. European Journal of Operational Research 151 (2003) 447-460
7. Burke, E.K., De Causmaecker, P., Vanden Berghe, G. and Van Landeghem, H.: The state of the art of nurse rostering. Journal of Scheduling 7 (2004) 441-499
8. Warner, H.W.: Scheduling Nursing Personnel According to Nursing Preference: A Mathematical Approach. Operations Research 24 (1976) 842-856
9. Maenhout, B., and Vanhoucke, M.: An Electromagnetism meta-heuristic for the nurse scheduling problem. Working paper 05/316 Ghent University (2005)
10. Osogami, T., and Imai, H.: Classification of Various Neighbourhood Operations for the Nurse Scheduling Problem. Lecture Notes in Computer Science, Vol. 1969 (2000) 72-83.
11. Glover, F: A Template for Scatter Search and Path Relinking. Lecture Notes in Computer Science, Vol. 1363 (1998) 13-54
12. Vanhoucke, M., and Maenhout, B.: Characterisation and Generation of Nurse Scheduling Problem Instances. Working paper 05/339 Ghent University (2005)
13. Casado, S., Laguna, M., and Pacheco, J.: Heuristical Labor Scheduling to Optimize Airport Passenger Flows. Journal of the Operational Research Society 56 (2005) 649-658
14. Glover, F., and Laguna, M.: Fundamentals of Scatter Search and Path Relinking. Control and Cybernetics 3 (2000) 653-684
15. Marti, R., Laguna, M., Glover, F.: Principles of scatter search. European Journal of Operational Research 169 (2006) 359-372
16. Aickelin, U.: Genetic Algorithms for Multiple-Choice Optimisation Principles. PhD, University of Wales Swansea (1999)
17. Glover, F., and Laguna, M.: Tabu Search. Kluwer Academic Publishers (1997)
18. Martins, E.Q.V., and Pascoal, M.M.B.: A new implementation of Yen's ranking loopless paths algorithm. 4OR – Quarterly Journal of the Belgian, French and Italian Operations Research Societies 1 (2003) 121-134
19. Kuhn, H.: The Hungarian method for the assignment problem. Naval Research Logistics 2 (1955) 83-97
20. Schrage, L.: LINDO: Optimization software for linear programming. LINDO Systems Inc.: Chicago, IL (1995)