Centro de Investigação Operacional

Solving a Bi-objective Nurse Rerostering Problem by Using a Utopic Pareto Genetic Heuristic

Margarida Vaz Pato and Margarida Moz

Solving a Bi-objective Nurse Rerostering Problem by Using a Utopic Pareto Genetic Heuristic

Margarida Vaz Pato and Margarida Moz

Instituto Superior de Economia e Gestão
Universidade Técnica de Lisboa
Centro de Investigação Operacional, Universidade de Lisboa

address: Margarida Vaz Pato
Instituto Superior de Economia e Gestão
Departamento de Matemática
Rua do Quelhas, 6
1200-781 Lisboa Portugal

fax: +351-21-3922781

e-mail: mpato@iseg.utl.pt

Abstract

Nurse rerostering arises when at least one nurse announces that she will be unable to undertake the tasks previously assigned to her. The problem amounts to building a new roster that satisfies the hard constraints already met by the current one and, as much as possible, fulfils two groups of soft constraints which define the two objectives to be attained. A bi-objective genetic heuristic was designed on the basis of a population of individuals characterised by pairs of chromosomes, whose fitness complies with the Pareto ranking of the respective decoded solution. It includes an elitist policy, as well as a new utopic strategy, introduced for purposes of diversification. The computational experiments produced promising results for the practical application of this approach to real life instances arising from a public hospital in Lisbon.

Key words: nurse scheduling, rerostering, bi-objective heuristics, genetic algorithms.

Introduction

The rerostering problem is a scheduling-type problem that frequently arises in hospital units where no reserve pool of nurses exists. Here, the roster for the nurses of a given unit must often be adapted in view of the scheduled nurses' unexpected absences. However, like many other rescheduling problems, this rerostering problem has received little attention in OR literature.

As witnessed in hospital units in Portugal, the nurse rerostering problem has been studied by using two single-objective multi-commodity flow models and tackled by mixed binary linear formulations or genetic heuristics in Moz and Pato (2003, 2004, 2005a). The abovementioned single-objective problem consists in minimising the violations related to a group of soft constraints – by forcing a nil dissimilarity between the current and the new roster – subject to the hard constraints imposed by Portuguese Law, working contracts and institutional requirements. This being so, the solutions to such optimisation problem will tend to satisfy other soft constraints whenever satisfied by the current roster. However, in practice, there is a trade-off between similarity of both rosters and compliance with other soft constraints. Hence, to handle this trade-off, a bi-objective problem was formulated and the respective solutions were obtained from a goal programming model (Moz and Pato (2005b)).

In the literature, a reference was found to a nurse rerostering problem in Tien and Kamiyama (1982). However, it concerns a situation where a reserve pool of nurses exists. More recently, other rescheduling problems have been analysed by Cumming, Paechter and Rankin (2000) and Petrovic, Beddoe and Vanden Berghe (2002). Nevertheless, all of these studies have little in common with the present nurse rerostering issue, though all have been identified as being more difficult than the corresponding scheduling issues. In fact, the methodologies usually applied to the rostering problem cannot be adapted to rerostering that easily. For a recent survey of methods for rostering see Burke et al. (2004), Cheang, Lim and Rodrigues (2003) or Ernst et al. (2004).

Due to the high complexity of this problem, whose single-objective version has been classified in Moz and Pato (2005a) as NP-hard (Garey and Jonhson (1979)), the authors

adopted a heuristic methodology. This strategy was also dictated by the need for a solving engine requiring modest human and computing resources, to be incorporated in a decision support system and run in each hospital unit. The choice of a genetic heuristic was prompted by the favourable experience obtained from genetic heuristics by several authors working with difficult bi-objective scheduling-type problems, such as Carrasco and Pato (2001) and Burke et al. (2001). In this context one should also consult a recent survey of Silva, Burke and Petrovic (2004) on multiobjective heuristics for scheduling problems, which includes nurse rostering.

The Utopic Pareto Genetic heuristic described in this paper was developed by introducing the bi-objective criterion into the fitness function, in a Pareto fashion, and running it, while maintaining all the other features considered in the abovementioned genetic heuristic for a single-objective case (Moz and Pato (2005a)). In an attempt to enhance the behaviour of the heuristic, an elitist policy was adapted from the one due to Gandibleux (2000), as well as a new utopic strategy developed in order to induce diversity in the population and attain a low level of at least one of the objectives. The bi-objective genetic heuristic was computationally tested, using real life rerostering instances taken from a surgical unit of a Lisbon public hospital.

Section 1 of this paper presents the nurse rerostering problem itself and the respective bi-objective model, whereas section 2 is devoted to the presentation of the bi-objective genetic heuristic and section 3 to the computational experiments. Finally, in section 4 some comments are made.

1. The Nurse Rerostering Problem

There follows a brief description of the problem, which may be found, in a more detailed version, in Moz and Pato (2005b). The problem arises when, during the rostering period, one or more nurses announce that they will be unable to perform tasks assigned to them in the current roster.

This scheduling-type problem is highly constrained, in view of labour contract rules in Portugal, as well as rules peculiar to the case under study. It should, however, be noted that nowadays most of the nurses employed at public or non-public Portuguese hospitals operate under similar conditions. Some of these rules are imperative, the so-called hard

constraints, whereas others, the soft constraints, should be satisfied as much as possible. Such regulations may be summarised by the following constraints:

– (hard i) each nurse has an eight-hour working day in a single shift, referred to as a duty or a work task;

– (hard ii) some nurses have an average 35 hour weekly workload, while that of others is 42 hours;

– (hard iii) nurses must rest at least 16 hours between two consecutive shifts; this means that some shift sequences are not feasible (e.g. work on an evening shift should not be followed by work on a night shift or on a day shift);

– (hard iv) nurses must enjoy a minimum number of days' leave in every seven day sequence, depending on their weekly workload (two days-off for those with an average weekly workload of 35 hours, and at least one day-off for those with a 42-hour week);

– (hard v) nurses who should not be working on certain shifts and/or on certain days, for contractual reasons or due to announced absences, may not be assigned to perform tasks pertaining to those shifts and/or days;

– (soft i) some nurses should not work on the night shift on consecutive days;

– (soft ii) nurses should be assigned a pre-determined number of work tasks during the rerostering period (from the first day of absences to the end of the rostering period), in keeping with the respective weekly workload;

– (soft iii) the preferences of nurses for some sequences of tasks should be met (in the rerostering context, nurses prefer not to alter the previously assigned work tasks).

As for (soft ii), it is worth noting that in a rostering period of four weeks, a 42-hour weekly workload for a nurse amounts to 168 hours' work in 21 shifts, whereas a 35-hour weekly workload amounts to 140 hours, that is, 17.5 shifts. This being so, the head nurse, who currently devises the rostering for the unit, tries to assign to that nurse 17 shifts in one period and 18 shifts in the next period.

Figure 1 displays a seven-day roster for a hypothetical small hospital unit with a fixed set of five nurses, where nurse 3 announces an absence on the night shift of day 5 of the seven-day period.

day	1	2	3	4	5	6	7	# duties	
nurse 1	D	E	O	N	O	D	E	5	
nurse 2	O	N	O	D	E	O	N	4	
nurse 3	O	D	E	O	N̶	O	D	4	
nurse 4	E	O	N	O	D	E	O	4	
nurse 5	N	O	D	E	O	N	O	4	
#N	1	1	1	1	1	1	1		
#D	1	1	1	1	1	1	1		
#E	1	1	1	1	1	1	1		

9 duties:

$$\underbrace{\overset{1}{D}\ \overset{2}{E}\ \overset{3}{N}}_{\text{day 5}}\ \underbrace{\overset{4}{D}\ \overset{5}{E}\ \overset{6}{N}}_{\text{day 6}}\ \underbrace{\overset{7}{D}\ \overset{8}{E}\ \overset{9}{N}}_{\text{day 7}}$$
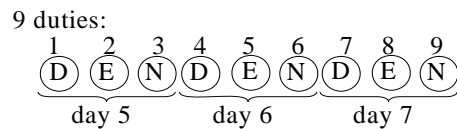
Figure 1. The rerostering problem for a hypothetical hospital unit.

The current roster, requiring one nurse for each day shift, is here represented in precisely the same way as in the hospital unit. The symbols D, E, N and O stand for day, evening, night shifts and day-off, respectively. Note that the cross on the night shift cell (N) of day 5 is assumed to have been marked by the head nurse of the unit immediately after announcement of the unexpected absence by nurse 3.

In this case, the rerostering problem consists of rebuilding the roster by reassigning the nine duties, listed at the bottom of Figure 1, on days 5, 6 and 7. The rebuilding procedure amounts to 15 tasks: nine duties and six days-off. Of course, these 15 tasks from day 5 to day 7 will have to be performed by the five nurses.

For instance, one solution for the rerostering issue, shown in Figure 2, may be to assign nurse 1 to that night shift (N) and a day-off (O) on the next day. Nurse 3 will work on the day shift (D) of day 6, while all the others perform the same tasks. In fact, nurse 1 may perform the night shift on day 5, since she was assigned to a night shift on the previous day and will get the day-off on day 6, accounting for the two days-off in each seven-day period.

In the general case, the present bi-objective model deals with soft constraints referred to as (soft i) as if they were hard constraints. Therefore, the two objectives to attain concern (soft ii), and (soft iii) constraints.

day	1	2	3	4	5	6	7	# duties
nurse 1	D	E	O	N	N	O	E	5
nurse 2	O	N	O	D	E	O	N	4
nurse 3	O	D	E	O	O	D	D	4
nurse 4	E	O	N	O	D	E	O	4
nurse 5	N	O	D	E	O	N	O	4
#N	1	1	1	1	1	1	1	
#D	1	1	1	1	1	1	1	
#E	1	1	1	1	1	1	1	

Figure 2. A solution to the illustrative rerostering problem.

In this context, for the rerostering issue addressed, one must, for all the nurses of the unit, determine a roster which satisfies the hard constraints plus (soft i), and is guided by the two scheduling objectives:

– (objective 1) to obtain a minimum gap between the number of scheduled duties and the number of duties each nurse should perform during the period;

– (objective 2) to minimise dissimilarity regarding the previously announced roster for the same period.

2. Utopic Pareto Genetic Heuristic

Genetic heuristics have been studied for difficult multi-objective optimisation problems since the pioneering work of Schaffer (1985). From the various heuristics of this type our option favoured a Pareto genetic approach (see, for a survey, Zitzler, Laumanns and Bleuler (2004)).

The major difference between the basic genetic heuristic, devised for the bi-objective rerostering problem referred to in the Introduction, and the one developed for the single-objective case in Moz and Pato (2005a), lies in the fitness function, which now considers the two objectives of the problem under study. In addition, the enhanced bi-objective version – Utopic Pareto Genetic heuristic – includes an elitist policy and a utopic diversification strategy. It should be stressed that knowledge of the rerostering problem and of previous experiments favoured the abovementioned bi-objective genetic heuristic, which achieves a compromise between computing effort and effectiveness.

As a matter of fact, like many others within the scheduling domain, the problem addressed has a combinatorial structure but possesses a highly constrained nature. Moreover, the solutions produced by the heuristic are assumed to be used directly in the

hospital unit. Consequently, all the hard constraints must be satisfied, i.e., the solution must be feasible. The need to produce feasible solutions, together with the difficulty in obtaining such solutions, due to the intricate rules of work contracts, led the authors to consider employing roulette wheel selection instead of the binary tournament strategy. In fact, under binary tournament, two solutions from the entire population are randomly chosen and, in the presence of many unfeasible solutions, the probability of selecting two solutions that are both unfeasible is high.

On the other hand, the previously mentioned experiments included the merging of the individuals corresponding to the Pareto frontier of a generation in the population for the next. This led to premature convergence and poor quality results, which explains the need for a specially devised heuristic – one which could, throughout all generations, maintain several Pareto rank levels, within which the genetic operators effectively act. In the most common multi-objective genetic algorithms, such as SPEA2 and NSGA-II (Deb et al. (2002) and Zitzler, Laumanns and Thiele (2001)), the preference for the individuals of the first Pareto ranks to pass to the next generation excludes those of higher ranks, thus reducing the diversity required for rerostering application.

There follows a brief description of the main components of this approach, whose pseudo-code is given in Table 1.

Step 1 (initialisation)
Generate n_pop individuals for generation 1 by running a single-objective genetic algorithm
Generate the utopic individual and include it in the population for generation 1
$t \leftarrow 1$
Step 2 (evaluation)
Assign a bi-objective fitness value to each individual of the population in generation t
Update the two lexicographic optima
Step 3 (iterative step)
Perform selection to create the mating pool and crossover on randomly chosen pairs of individuals
Submit the individuals to mutation
$t \leftarrow t+1$
Include the utopic individual in the population for generation $t+1$ and apply elitism
Step 4 (stopping criterion)
IF $t > max\_gen$, THEN STOP; OTHERWISE, GO TO step 2

Table 1. Utopic Pareto Genetic heuristic.

2.1 Encoding and Initial Population

As mentioned in the previous section, the rerostering process involves a list of nurses pertaining to the unit, besides a list of work tasks to be performed during the rerostering period. In the real case under study, this period extends from the first day of absences to the end, day 28. This being so, the list of work tasks, besides the list of nurses, are regarded as chromosomes. Thus, each individual of the population, which corresponds to a unique solution of the rerostering problem (if feasible, a new roster), is characterised by a pair of permutation chromosomes: one is a list of work tasks and the other a list of nurses. Figure 3 shows this encoding scheme.
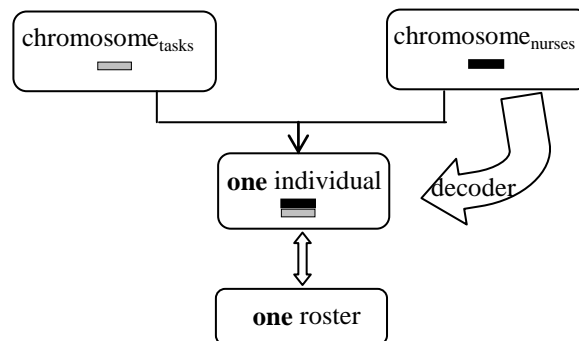

Figure 3. Encoding scheme.

As for decoding the two chromosomes, the exercise is performed in a natural way by considering the list of work tasks and the list of nurses, both of which are represented by integer vectors. Hence, in order to obtain a solution for the rerostering problem from one individual, which may be a new roster, one must take the respective pair of chromosomes and run the decoder, which amounts to a constructive heuristic acting in accordance with the description given in Table 2. Note that indirect encodings – albeit in a different way from the latter, and applied to individuals characterised by a unique chromosome – were also devised by Davis (1991) within a genetic heuristic for colouring and, more recently, by Aickelin and Dowsland (2004) for nurse rostering.

In short, the constructive heuristic includes a main step in which each non-assigned task is given to a nurse. One such assignment is performed according to the scheduling rules, which are ordered hierarchically, and imposes satisfaction of the constraints.

Step 1 (initialisation)
All the work tasks are randomly ordered and considered as not assigned
Step 2
Each non-assigned task of the list is assigned to a nurse according to hierarchically ordered rules
Step 3
IF it is not possible to assign a task, say *A*, to any nurse, THEN the procedure reverts back to the last task already assigned, say *B*, swaps the position of tasks *A* and *B* and tries to assign *B* by satisfying the hierarchical rules
Step 4 (stopping criterion)
IF all the tasks are assigned or a task could not be assigned, THEN STOP; OTHERWISE, GO TO step 2

Table 2. Constructive heuristic.

There are *n_pop*=400 individuals, each assigned to a pair of chromosomes, in the population in all generations. To create the population for generation 1, the single-objective version of the genetic heuristic (minimising dissimilarity, i.e., objective 2 in the bi-objective model) runs during a pre-defined number of iterations. Its output is the initial population for the bi-objective heuristic, including with high probability some solutions which comply with the hard constraints.

2.2 Bi-objective Fitness

The fitness of each individual must reflect the bi-objective nature of the problem. Hence, the option favoured that of calculating the fitness according to the position of the individual in relation to the Pareto frontier of the current population, as in Goldberg (1989) and NSGA-based algorithms (Deb et al. (2002)).

As mentioned above, in a specific iteration of the genetic algorithm, the decoder – constructive heuristic – acts on each pair of chromosomes of an individual, thus producing a solution for the rerostering problem. For each solution, the values of both objective functions (the deviation from the required number of duties and the number of shift swaps in the nurses' schedules) are calculated, and a point in the objective space is defined.

After representing the points corresponding to all the individuals obtained from the decoder in the objective space, those corresponding to the Pareto frontier are assigned to

Pareto rank 1 and are given a fitness value equal to 1. The next step consists in eliminating those points and, where the remaining points of the objective space are concerned, determining the next Pareto frontier, thus defining the rank 2 frontier. The respective individuals receive the value 1/2 for fitness. And the process continues, always giving fitness values to the individuals that are equal to the inverse of the respective frontier rank.

Figure 4 presents a small example for a population of 10 points in the objective space, plus the point associated with the utopic individual, in a particular generation *t*. The utopic individual, with a fitness value equal to 1, is represented by a different 'crowned' symbol. Its origin and motivation will be explained in the next subsection. In this example, the frontiers were built up to rank 3, whose individuals have a fitness value equal to 1/3.
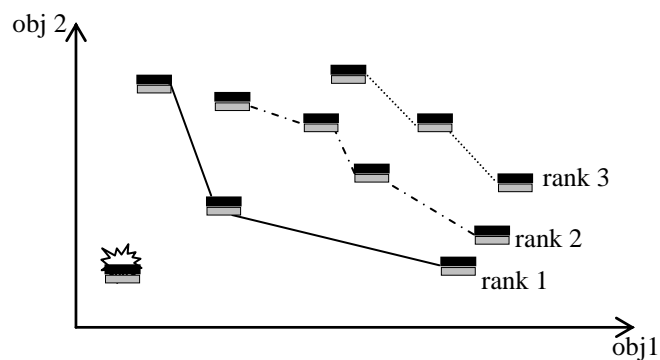

Figure 4. Illustration of the Pareto ranks.

It is important to stress that the decoder heuristic may lead to an unfeasible solution, in other words, a roster that does not satisfy all the hard constraints it should satisfy. This being so, then the respective objective function values will be penalised.

2.3 Operators, Elitism and Utopia

As for the operators, let us begin with the selection operator, which is based on the roulette wheel procedure, in other words, the probability of an individual entering the mating pool being directly proportional to its fitness. From the population of each generation, the selection operator selects *n_pop* individuals for the mating pool.

The Utopic Pareto Genetic heuristic embeds both elitism and a utopic strategy. Elitism consists in forcing two individuals into the population of every generation. These correspond to the two lexicographic optima of the current generation.

In addition to these individuals, a third one is always introduced in the population – the utopic individual – corresponding to a false solution. It is generated in the initial phase of the heuristic by running the single-objective genetic heuristic, referred to at the beginning of section 2, for a relaxation of the problem. The single-objective is, precisely, objective 2 and the relaxed constraints are hard (ii) to hard (v). At the end, the utopic individual is the best fitted one of the last generation, provided it is unique. Should it not be the case, the one with the lowest value for objective 1 is chosen among the best fitted ones. As a result, it is usually an unfeasible solution and attains a low dissimilarity. However, as a rule, this is not so for objective 1, as the two objectives are potentially conflicting.

In every generation of the Utopic Pareto Genetic heuristic, the utopic individual corresponds to a point in the objective space that is below the rank 1 frontier, but its fitness value is always set at equal to 1. Thus this individual will have a high probability of being selected by the roulette wheel for the mating pool. Solutions created from the utopic individual by crossover and mutation are therefore expected to have a low value for at least objective 2 and to present a small number of hard constraints violated, as objective 2 enforces a roster similar to the current one that already satisfies all the hard constraints.

The genetic operators of crossover and mutation act in the mating pool, thus creating the population for the next generation. Individuals in the mating pool are matched at random, and the crossover probability is equal to 60%. The PMX type crossover operates independently over the pair of chromosomes for tasks and over the pair of chromosomes for nurses, from a pair of matched individuals. This is followed by substitution of the parents by the offspring in the mating pool. Figure 5 illustrates the recombination of two individuals through PMX adapted to this double chromosome encoding.
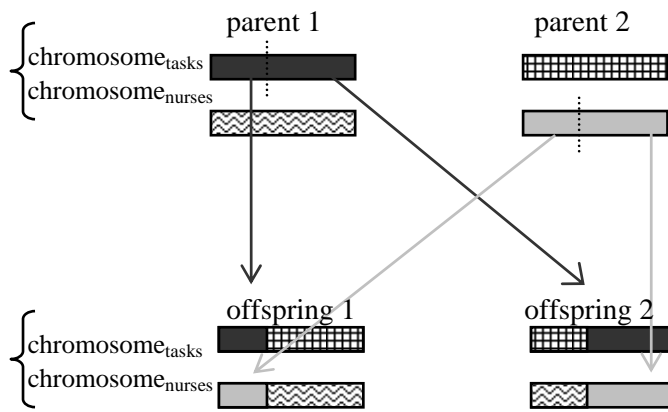
Figure 5. Illustration of the crossover operator.

As for mutation, one opted in favour of simply swapping the position of two randomly chosen genes of the chromosomes of each individual that has just been selected for mutation from the mating pool. The mutation probability is equal to 0.1%.

3. Computational results

The computational experiments were based on real data taken from a unit of a Lisbon public hospital. In this unit the number of nurses is 19, which is a standard dimension for this kind of service. As the planning period duration is 28 days, the rerostering period may range from one to 28 days, depending on the day on which the announcement of an absence occurs.

Four main groups of instances were tested, according to the week of the first day of absences, thus resulting in different lengths for the rerostering period. In the first group, comprising 8 instances, the first day of absences lies in the last week of the rostering period, so the rerostering period covers at the most one week. However, in the second group of 7 instances, the duration of the rerostering period is, at the most, two weeks, yielding problems of a higher dimension. The third and fourth groups, both with 8 instances each, correspond to absences in the second and in the first week, respectively. Details regarding these and other similar real instances are given in Moz (2003).

Two heuristics were implemented: the Utopic Pareto Genetic heuristic and a basic bi-objective version without elitism and utopic strategy, the latter having been tested to assess the effect of these features. The algorithms were coded in Pascal (Borland Delphi

5.0) and ran on a 2.6 Ghz and 256 Mb RAM INTEL Pentium IV processor. The two algorithms were applied five times to each one of the abovementioned 31 instances.

Previous experiments with these bi-objective genetic heuristics suggested that the stopping criterion should be at *max_gen*=2000 generations, since running the algorithm for more iterations would not lead to further improvement in solutions. As for the auxiliary single-objective genetic heuristic used to create the initial population, it was run for 400 generations.

Results from the basic bi-objective heuristic are displayed in Table 3 and those from the version with utopia and elitism in Table 4, whereas Table 5 presents results for both. Note that columns 4 to 10 in Tables 3 and 4 and columns 4 and 5 in Table 5 refer to average values obtained from, at the most, five runs of the algorithms for each instance. The average values do not always take into account five runs as the basic bi-objective genetic heuristic (Table 3) was unable to find a feasible solution in a number of runs, and never found a feasible solution for the instance III.3_19. The last two columns of Table 5 present the average computing times in seconds.

In Tables 3 and 4, the second and third columns present the optimum values of the single-objective models. The next four columns (4, 5, 6 and 7) of these two tables display the average minimum and maximum values taken from the Pareto frontier of the last generation obtained from the genetic heuristics. Column 8, in the same tables, refers to the average number of different ranks in the population of the last generation, the so-called wave metric, whereas columns 9 and 10 show the average spread metric, once more for the Pareto frontier of the last generation – Euclidean distance between the two lexicographic points – and the average Pareto ratio metric (e.g. Collete and Siarry (2005)).

It is worth noting that several metrics have been proposed in the literature to assess the behaviour of multiobjective heuristics (see, Deb (2001) and Fonseca et al. (2005)). However, no more than simple metrics were implemented, i.e., absolute metrics to evaluate each one of the heuristics and a relative metric to compare the behaviour of the two. This decision was motivated by one specific feature of the rerostering problem and its practical instances under study: in most of the cases the bi-objective genetic

heuristics produced only a small number of non-dominated points in the last generation. One suspects that this effect is due to the existence of only a few points in the exact Pareto frontier.

1	2	3	4	5	6	7	8	9	10
	single-objective optimum		basic bi-objective genetic heuristic				- absolute metrics (average)		
instances	obj1	obj2	obj1 min	obj1 max	obj2 min	obj2 max	wave	Pareto ratio	spread
I.1_19	2	3	2,00	2,00	3,00	3,00	2,00	0,50	0,00
I.2_19	1	2	2,00	2,00	2,00	2,00	1,00	1,00	0,00
I.3_19	3	9	3,60	5,00	9,00	11,00	3,80	0,42	2,40
I.4_19	0	2	0,40	1,00	2,00	3,00	4,00	0,42	0,77
I.5_19	5	20	5,67	5,67	21,33	21,33	1,67	0,78	0,00
I.6_19	2	8	2,40	3,00	9,00	10,00	3,60	0,30	1,45
I.7_19	7	20	7,00	7,00	22,00	22,00	1,67	0,67	0,00
I.8_19	0	2	0,00	1,00	2,00	3,00	3,40	0,32	1,41
II.1_19	0	1	0,00	0,00	1,00	1,00	3,40	0,33	0,00
II.2_19	0	0	0,00	0,00	0,00	0,00	1,00	1,00	0,00
II.3_19	0	5	0,80	1,00	5,00	5,80	3,80	0,36	0,82
II.4_19	1	12	3,00	3,00	13,00	13,00	4,20	0,15	0,00
II.5_19	0	6	1,00	1,00	6,00	6,00	4,00	0,18	0,00
II.6_19	1	17	2,00	3,00	19,00	20,25	4,50	0,41	1,50
II.8_19	0	5	0,00	1,00	5,00	6,00	3,80	0,35	1,41
III.1_19	1	7	2,00	2,00	7,00	7,00	4,60	0,16	0,00
III.2_19	0	12	2,00	2,00	14,00	14,00	3,00	0,33	0,00
III.3_19	1	13	-	-	-	-	-	-	-
III.4_19	0	7	1,00	3,00	7,00	9,60	3,80	0,35	3,31
III.5_19	5	27	5,00	5,50	28,25	29,25	4,00	0,23	0,35
III.6_19	4	27	5,00	5,00	30,50	30,50	2,50	0,41	0,00
III.7_19	3	19	3,40	3,40	19,80	19,80	2,60	0,46	0,00
III.8_19	2	11	3,60	5,60	11,40	13,60	3,40	0,51	2,83
IV.1_19	0	9	1,60	3,00	9,00	10,40	3,80	0,31	1,98
IV.2_19	0	12	1,40	3,00	12,00	13,60	3,40	0,35	2,26
IV.3_19	0	10	0,20	1,80	11,80	14,00	4,60	0,31	2,40
IV.4_19	3	34	5,50	5,50	38,50	38,50	4,50	0,17	0,00
IV.5_19	4	18	5,40	6,60	20,20	23,40	3,00	0,60	1,59
IV.6_19	3	23	5,50	5,50	27,00	27,00	3,50	0,25	0,00
IV.7_19	0	9	2,20	5,20	9,00	10,80	5,00	0,26	2,55
IV.8_19	1	10	2,80	4,80	10,60	13,20	5,00	0,29	3,22
average	1,58	11,61	2,55	3,25	12,51	13,40	3,42	0,41	1,01

Table 3. Computational results – basic bi-objective genetic heuristic.

As for the average extreme values, one finds that the heuristic with utopia and elitism is better in terms of achievement of the minimum values for the two objectives (columns 4 to 7 of Tables 3 and 4). In fact, in the case of the Utopic Pareto Genetic heuristic, the gap between the average minima for objective 1 (last line of column 4 in Table 4) and the corresponding single-objective average optimum value (column 2) is 0.18. An identical comparison for objective 2 and the same heuristic shows a gap of 0.53. Moreover, it is interesting to observe that, for each objective, the difference between the maximum value and the single-objective optimum is significant. It thus reveals a greater diversity attained with this elitist and utopic enhanced heuristic.

The wave metric for both heuristics, on average equal to 3.42 and 3.46, respectively, indicates equivalent heuristic behaviour.

1	2	3	4	5	6	7	8	9	10
	single-objective optimum		Utopic Pareto Genetic heuristic - absolute metrics (average)						
instances	obj1	obj2	obj1 min	obj1 max	obj2 min	obj2 max	wave	Pareto ratio	spread
I.1_19	2	3	2,00	2,00	3,00	3,00	2,40	0,43	0,00
I.2_19	1	2	1,00	2,00	2,00	3,00	3,00	0,48	1,41
I.3_19	3	9	3,00	4,80	9,00	11,40	4,40	0,32	3,07
I.4_19	0	2	0,00	1,00	2,00	3,00	4,20	0,30	1,41
I.5_19	5	20	5,20	6,00	20,60	23,60	1,20	0,90	3,11
I.6_19	2	8	2,00	3,80	9,00	11,20	4,80	0,27	2,54
I.7_19	7	20	7,00	7,00	22,60	22,60	1,80	0,67	0,00
I.8_19	0	2	0,00	1,00	2,00	3,00	4,40	0,28	1,41
II.1_19	0	1	0,00	0,00	1,00	1,00	3,80	0,17	0,00
II.2_19	0	0	0,00	0,00	0,00	0,00	1,00	1,00	0,00
II.3_19	0	5	0,00	1,00	5,00	10,20	3,60	0,38	5,30
II.4_19	1	12	1,60	4,80	12,00	19,20	6,60	0,31	6,64
II.5_19	0	6	0,00	1,20	6,00	7,00	6,00	0,22	1,58
II.6_19	1	17	1,00	3,40	17,00	22,60	3,80	0,49	5,87
II.8_19	0	5	0,00	1,00	5,00	6,40	6,40	0,21	1,76
III.1_19	1	7	2,00	2,00	7,00	7,00	4,20	0,16	0,00
III.2_19	0	12	1,20	3,00	12,00	17,40	2,20	0,74	1,70
III.3_19	1	13	1,20	3,00	13,20	20,00	1,80	0,85	7,05
III.4_19	0	7	0,00	3,00	7,00	10,00	4,80	0,29	4,24
III.5_19	5	27	4,20	6,00	28,20	33,00	1,00	1,00	5,16
III.6_19	4	27	4,40	6,80	30,20	35,40	1,20	0,93	7,07
III.7_19	3	19	3,00	4,00	19,00	20,80	1,20	0,93	2,09
III.8_19	2	11	2,00	6,00	11,00	13,80	5,20	0,39	7,92
IV.1_19	0	9	0,00	3,00	9,00	16,80	5,40	0,40	8,30
IV.2_19	0	12	0,20	3,00	12,40	18,00	2,40	0,70	7,00
IV.3_19	0	10	0,00	2,00	10,00	15,20	5,40	0,30	5,70
IV.4_19	3	34	4,00	6,20	37,60	43,20	1,40	0,90	6,00
IV.5_19	4	18	4,00	6,40	18,00	24,40	1,80	0,70	6,80
IV.6_19	3	23	4,00	6,20	26,40	33,20	1,20	0,90	7,20
IV.7_19	0	9	0,60	4,00	9,00	16,00	5,00	0,50	7,80
IV.8_19	1	10	1,00	4,80	10,00	17,60	5,80	0,40	8,50
average	1,58	11,61	1,76	3,50	12,14	15,77	3,46	0,53	4,08

Table 4. Computational results – Utopic Pareto Genetic heuristic.

The low values of the spread corresponding to the basic bi-objective heuristic are due to the low cardinality Pareto frontiers, often with a unique solution. The higher values for the Utopic Pareto Genetic heuristic prove that the utopic and the "lexicographic" individuals attract more solutions to the frontier.

Finally, columns 2 and 3 of Table 5 give, for each instance, the relative metrics which have been computed to enable one to compare both versions on the strength of the proportion of points of the candidate Pareto frontier of one heuristic that are dominated by the corresponding points of the other. Here, the candidate Pareto frontier of a genetic heuristic denotes the set of non-dominated points obtained from the five runs, for that

specific instance. On the basis of the results obtained one can conclude that the Utopic Pareto Genetic heuristic is more successful in producing non-dominated solutions.

1	2	3	4	5
	relative metrics		time (seconds)	
	dominated	dominated	average	
instances	basic	utopic	basic	utopic
I.1_19	1,00	1,00	53,00	61,51
I.2_19	1,00	0,50	54,50	67,32
I.3_19	1,00	0,67	88,60	106,05
I.4_19	1,00	1,00	110,10	121,87
I.5_19	1,00	1,00	112,33	224,68
I.6_19	1,00	1,00	114,71	151,28
I.7_19	1,00	1,00	128,61	215,32
I.8_19	1,00	1,00	110,38	119,91
II.1_19	1,00	1,00	129,13	131,24
II.2_19	1,00	1,00	117,47	116,31
II.3_19	1,00	1,00	148,14	151,32
II.4_19	1,00	0,25	195,69	263,48
II.5_19	1,00	0,50	132,20	135,43
II.6_19	0,67	1,00	232,58	267,14
II.8_19	1,00	1,00	185,86	204,98
III.1_19	1,00	1,00	222,34	225,60
III.2_19	1,00	0,33	254,85	324,04
III.3_19	-	-	266,71	353,28
III.4_19	1,00	0,75	276,39	279,53
III.5_19	0,00	0,67	348,01	502,87
III.6_19	0,00	0,50	307,68	449,20
III.7_19	0,00	1,00	249,53	389,88
III.8_19	1,00	0,80	265,48	356,19
IV.1_19	1,00	0,75	308,80	311,40
IV.2_19	1,00	0,75	326,30	393,70
IV.3_19	1,00	0,33	394,82	441,00
IV.4_19	1,00	0,00	476,54	585,60
IV.5_19	0,67	0,00	425,55	526,90
IV.6_19	0,00	0,25	380,59	491,10
IV.7_19	0,50	0,80	406,01	379,50
IV.8_19	1,00	0,00	395,25	386,50
average	0,83	0,70	232,84	281,75

Table 5. Computational results – comparison of the two genetic heuristics.

One should add that both the Utopic Pareto Genetic heuristic and the basic version are time-consuming. The computational experiment showed that the time spent at the initialisation step, i.e. at the generation of the initial population was, on average, within 11% and 17% of the total computing time for each heuristic, the last-mentioned figures referred to in columns 4 and 5 of Table 5. However, running a bi-objective genetic

heuristic without this initialisation would achieve poorer results, as indicated by preliminary experiments.

Although it did take more computational time, the Utopic Pareto Genetic heuristic always generated several feasible solutions for the Pareto frontier of the last generation. Most of them proved to be of good quality for both objectives, and correspond to rosters that will be welcomed by the nurses and hospital directors alike.

In short, the utopic plus the elitist strategies, induced by the individuals continuously imposed on the population, produced a fairly positive effect on the bi-objective genetic heuristic behaviour.

4. Final Comments

A Utopic Pareto Genetic heuristic was tuned to handle the bi-objective nurse rerostering problem. The main differences, when compared with previous approaches, lie in the building of the initial population, by using a single-objective genetic heuristic, the inclusion of a utopic individual into the population to improve quality and diversification and, last but not least, recourse to elitism based on the approximate lexicographic optima.

A computational experiment on a set of real life instances, taken from a specialised central hospital, produced a significant advantage over the previous single-objective approach. In short, the results of this experiment suggest that the bi-objective genetic heuristic is competitive. The software required is easy to develop and dispenses with expensive computing resources. In fact, this heuristic is now being embedded in an interactive module generating different rosters provided by the individuals from the frontier of the last generation of the genetic algorithm. From these rosters, characterised by low global violations for the soft constraints, one will be chosen by the unit's head nurse and subsequently published for the rerostering period.

This method revealed another important advantage: it can, as it stands, be used for the rerostering problem that may be defined with other functions expressing the amount of violations of the soft constraints. It is common knowledge, for instance, that quadratic

functions more heavily penalise higher violations than do linear functions. It should be noted that linear, quadratic and other functions have already been devised for the single-objective rerostering problem in Moz (2003), whereas, to our knowledge, the nonlinear functions have not, as yet, been tested.

Another interesting point to explore in the sequence is the study of a model that considers the disaggregation of both objective functions per type of nurse. In fact, the current results do not appear to be balanced, insofar as they register violations of the soft constraints for some nurses and low ones for others. One such study would include an analysis of the relative incompatibility of the several objectives to be proposed.

Finally, one can infer from this study that the Utopic Pareto Genetic heuristic can easily be explored for other rescheduling problems, besides the nurse rostering problem itself.

Aknowledgements

References

Aickelin, U. and Dowsland, K.A. (2004). "An Indirect Genetic Algorithm for a Nurse Scheduling Problem", Computers & Operations Research 31(5), 761-778.

Burke, E.K., De Causmaecker, P., Vanden Berghe, G. and Van Landerghem, H. (2004). "The State of the Art of Nurse Rostering", Journal of Scheduling 7(6), 411-499.

Burke, E.K., Cowling, P., De Causmaecker, P. and Vanden Berghe, G. (2001). "A Memetic Approach to the Nurse Rostering Problem", Applied Intelligence 15(3), 199-214.

Carrasco, M.P. and Pato, M.V. (2001). "A Multiobjective Genetic Algorithm for the Class/Teacher Timetabling Problem", in Practice and Theory of Automated

Timetabling III: Third International Conference PATAT 2000, selected revised papers, E. Burke and W. Erben (eds.), Springer, LNCS 2079, 3-17.

Cheang, B., Li, H., Lim, A. and Rodrigues, B. (2003). "Nurse Rostering Problems – A Bibliographic Survey", European Journal of Operational Research 151(3), 447-460.

Collette, Y. and Siarry, P. (2005). "Three New Metrics to Measure the Convergence of the Metaheuristics towards the Pareto Frontier and the Aesthetic of a Set of Solutions in Biobjective Optimization", Computers & Operations Research 32(4), 773-792.

Cumming, A., Paechter, B. and Rankin, R. (2000). "Post-Publication Timetabling", in Practice and Theory of Automated Timetabling III: Third International Conference PATAT 2000, extended abstracts, 107-108.

Davis, L. (1991). Handbook of Genetic Algorithms. Van Nostrand Reinhold, New York.

Deb, K. (2001). Multi-Objective Optimization Using Evolutionary Algorithms. Wiley, New York.

Deb, K., Pratap, A., Aggarwal, S. and Meyarivan, T. (2002). "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II", IEEE Transactions on Evolutionary Computation 6(2), 182-197.

Ernst, A.T., Jiang, H., Krishnamoorthy, M. and Sier, D. (2004). "Staff Scheduling and Rostering: a Review of Applications, Methods and Models", European Journal of Operational Research 153(1), 3-27.

Fonseca, C., Knowles, J., Thiele, L. and Zitzler, E. (2005). "A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers", Evolutionary Multi-Criterion Optimization Conference EMO 2005, Guanajuato, Mexico.

Gandibleux, X. (2000). Journée de Travail PM20 Programmatiom Mathématique Multiobjectif, Tours. November 2000.

Garey, M.R. and Johnson, D.S. (1979). Computers and Intractability: A Guide to the Theory of NP-completeness. W.F. Freeman & Company, Publishers. San Francisco.

Goldberg, D. (1989). Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, Massachusetts.

Moz, M. (2003). "Técnicas de Investigação Operacional Aplicadas a um Problema de Escalonamento de Pessoal em Contexto Hospitalar", PhD Dissertation, Instituto Superior de Economia e Gestão, Universidade Técnica de Lisboa.

Moz, M. and Pato, M.V. (2004). "Solving the Problem of Rerostering Nurse Schedules with Hard Constraints: New Multicommodity Flow Models", Annals of Operations Research 128(1-4), 179-197.

Moz, M. and Pato, M.V. (2003). "An Integer Multicommodity Flow Model Applied to the Rerostering of Nurse Schedules", Annals of Operations Research 119(1-4), 285-301.

Moz, M. and Pato, M.V. (2005a). "A Genetic Algorithm Approach to a Nurse Rerostering Problem", accepted for publication in Computers & Operations Research.

Moz, M. and Pato, M.V. (2005b). "A Bi-objective Network Flow Approach for Nurse Rerostering", in Proceedings of the INOC 2005 - International Network Optimization Conference, Luís Gouveia and Cândida Mourão (eds.), Lisbon, March 2005, B3.817.

Petrovic, S., Beddoe, G. and Vanden Berghe, G. (2002). "Storing and Adapting Repair Experiences in Personnel Rostering", in Practice and Theory of Automated Timetabling IV: Fourth International Conference PATAT 2002, selected revised papers, E. Burke and P. De Causmaecker (eds.), Springer, Lecture Notes in Computer Science 2740, 148-165.

Schaffer, J.D. (1985). "Multiple Objective Optimization with Vector Evaluated Genetic Algorithms", in Genetic Algorithms and their Applications, Proceedings of the Third

International Conference on Genetic Algorithms, J.J. Grefenstette (ed.), Lawrence Erlbaum, Hillsdale, NJ, 93-100.

Silva, J.D.L., Burke, E.K. and Petrovic, S. (2004). "An Introduction to Multiobjective Metaheuristics for Scheduling and Timetabling", in Metaheuristics for Multiobjective Optimisation, X. Gandibleux, M. Sevaux, K. Sörensen and V. T'kindt (eds.), Springer, Berlin.

Tien, J.M. and Kamiyama, A. (1982). "On Manpower Scheduling Algorithms", SIAM Review 24(3), 275-287.

Zitzler, E., Laumanns, M. and Bleuler, S. (2004). "A Tutorial on Multiobjective Optimization", in Metaheuristics for Multiobjective Optimisation, X. Gandibleux, M. Sevaux, K. Sörensen and V. T'kindt (eds.), Springer, Berlin.

Zitzler, E., Laumanns, M. and Thiele, L. (2001). "SPEA2: Improving the Strength Pareto Evolutionary Algorithm", TIK-Report 103, Computer Engineering and Networks Laboratory, Swiss Federal Institute of Technology, Zurich.