

A metaheuristics approach to the nurse rostering problem

Arthur Tórigo Gómez, José Vicente Canto dos Santos

Universidade do Vale do Rio dos Sinos. Av. Unisinos, 950, Cristo Rei, 93022-000, São Leopoldo, RS, Brazil
atgomezbr@gmail.com, jvcanto@unisinos.br

Maira Regina Poltosi

Laboratório Nacional de Computação Científica. Rua Getulio Vargas, 333, Quitandinha, 25651-075, Petropolis, RJ, Brazil
maira@lncc.br

Abstract. Health care providers are affected by problems of personnel costs. Usually, the generation of rosters is a hand-made and time-consuming task and does not always comply with the legislation and the internal rules. The article presents an approach to roster generation for nursing technicians according to legal and internal restrictions and in a satisfactory period of time. It is also designed to give employees a higher level of satisfaction concerning their day off preferences and a fair distribution of unpopular shifts. The article's proposal is to develop a hybrid system formed by a Tabu Search metaheuristic combined with a genetic algorithm. Experiments were carried out with artificial test cases based on real data. The results obtained were satisfactory, showing the feasibility of the solution in all tests performed.

Key words: rostering problem, tabu search, genetic algorithm, hybrid systems.

Introduction

Productivity problems and personnel turnover affect the costs of health care providers. The control of costs should not compromise the service quality. Currently, the failure to use computational tools for nurse rostering leads most of the hospitals and clinics to handle this task manually. This is a time-consuming task, since the schedules must be in accordance with the legislation, follow the organization's internal rules and take into account most of the employees' preferences.

The literature on the rostering problem is large, as one can see in Burke *et al.* (2004a) and Ernst *et al.* (2004a, 2004b). Variations in research are observed in the areas of application and in the definition of the problem, as well as in the employed solution methods: Jaumard (Jaumard *et al.*, 1998), Chun (Chun *et al.*, 2000), Barboza (Barboza *et al.*, 2003), Meisels and Schaefer (2003), and Özcan (2005). Currently, in the literature one can find real cases of the rostering problem treated with metaheuristics: Simulated Annealing as in Dowling (Dowling *et al.*, 1997), Tabu Search (TS) as in Dowsland (Dowsland, 1998) and Burke (Burke *et al.*, 1998), Genetic Algorithms (GA) as in Burke (Burke *et al.*, 2001),

Özcan (Özcan, 2005), Yeh (2007) and Tsai (Tsai and Li, 2009), and Iterated Local Search. Due to the large number of variables and constraints involved, it is classified as a NP-Hard optimization problem (Burke *et al.*, 2003; Osogami and Imai, 2000). Modelling the nurse rostering problem is not a trivial task, due to the large number of constraints to be considered and the complexity of the objective function that seeks, in general, to maximize the overall satisfaction of nurses or to reduce hospital costs by setting the schedule of nurses.

This paper proposes a model for generating rosters for nursing technicians to be utilized in a Brazilian hospital. The hospital's operational rules as well as the legislation's constraints should be considered. To increase the productivity and to give the employees a higher level of satisfaction, their day off preferences are considered and a fair distribution of unpopular shifts (Saturdays, Sundays and holidays) is sought.

Problem Definition

The administration of personnel includes the management of the workforce, i.e. the allocation and distribution of tasks in accordance

with the internal and external requirements of an organization. The elaboration of the work schedules corresponds to the stage of the allocation of personnel to periods of time and, sometimes, to places. To provide a 24-hour, 7 days a week service, the health facilities adopt shifts to generate the schedules. A monthly schedule of nursing presents the distribution of an infirmity team, considering every day of the month and the work shifts. Usually there are morning and afternoon shifts of 6 hours and a night shift of 12 hours. In the schedule, also the absences are registered, such as rests, vacations and leaves. The monthly schedule has the structure of a two-dimensional matrix, where each line corresponds to a technician's schedule, and an individual schedule, and the columns represent the days of the period that is being considered. In the generation of the monthly schedule, in order to obtain good quality results, the labor legislation, the institution's norms, the rules and the dynamics of the infirmity should be considered. A humanized schedule is also a desirable goal. These aspects determine the restrictions that the schedule should take into account. In this study they are:

- Minimum covering: the rests should be planned in such a way as to guarantee the necessary minimum number of professionals in attendance, in accordance with the demand of each ward/day/shift;
- Days off: to each technician should be given at least the number of days off required by law. The total of granted days off cannot exceed the number of obligatory rests, plus the number of additional days off that the technician can be entitled.
- Availability: if on certain day the technician is on leave or vacation, they cannot be assigned for work.

Soft Constraints

The soft constraints represent a group of factors that are susceptible to relaxation. It is highly exceptional, in practice, to find a roster that meets all the soft constraints (Burke *et al.*, 2004b), as they are frequently conflicting (Gascon *et al.*, 2000). These are the soft constraints considered in this work:

- Requests: dates requested for days off;
- Fairness: distributing duties on unpopular (Saturdays, Sundays and holidays) days in an equal way among the employees;
- Weekend: each employee must have at least one day off on a Saturday, as well

as one day off on a Sunday or holiday, providing a weekend off, not necessarily on consecutive days;

- Additional days off: hours worked besides the workload and holidays spent at work are converted into additional days off. Maximum of additional days off must be granted for each employee. Each employee must be given the maximum of additional days off;
- Interval: observing to the maximum the interval between the days off determined by law. In practice some relaxation is possible in this period;
- Ideal covering: it is preferable to work with the largest possible number of allocated personnel, but there is an ideal minimum number of personnel to be allocated to each ward / shift/day.

Methodology

This article involves the definition of a mathematical model, the development of a computational tool and the performance of validation tests. Next, it presents the mathematical model, the architecture of the developed program and how the metaheuristics Tabu Search and Genetic Algorithm were applied.

Mathematical Model

The rostering problems are considered problems of combinatorial optimization. In this model, each decision variable (DV) represents a penalty for the desirable but not observed restriction. The objective function (OF) evaluates the quality of the solution indicating how the desirable restrictions are being met, looking for the smallest sum of penalties. Each DV is given a weight that indicates the contribution of this variable to the objective function. The objective function evaluates the quality of the obtained solutions. The aim is to minimize the penalties resulting from violations of soft constraints. The symbols used in the mathematical formulation are presented in Table 1.

The binary variable x_{ijk} takes on the value 1 if the technician i is allocated to the shift j , on the day k , and the value 0 in the case of day off. X represents a solution as the set of all x_{ijk} .

The equation (1) that represents the objective function is composed of seven decision variables, represented by functions, and

The following formulation describes the proposed model:

Table 1. List of mathematical symbols.

X	Represents the solution as the set of all decision variables x_{ijk} .
$f_r(X)$	DV Request: penalty for not granted request for specific days off.
$f_f(X)$	DV Fairness: the standard deviation of the duties distributed on Saturdays, Sundays and holidays.
$f_w(X)$	DV Weekend: penalty for each employee who has not had their request met.
$f_a(X)$	DV Additional days off: penalty for each additional gap of each technician not granted in the schedule.
$f_i(X)$	DV Interval: penalty for each interval longer than the legal limit, but still within the allowed relaxation.
$f_{i+}(X)$	DV Interval +: Penalty for each interval that exceeds the legal limit allowed in addition to relaxation.
$f_c(X)$	DV Coverage: represents the penalty for instances of coverage below the minimum ideal.
Rn_{jk}	The matrix that contains the technicians' minimum need for the shift of the type j on the day k .
D_{ijk}	The binary matrix that represents the availability of the technician i in the shift j on the day k . It amounts to 1 if there is availability, 0 if there is none.
O_i	Vector that contains the number of obligatory days off for technician i .
A_i	Vector that contains the number of additional days off for technician i .

Minimize

$$F(X) = P_r \cdot f_r(X) + P_f \cdot f_f(X) + P_w \cdot f_w(X) + P_a \cdot f_a(X) + P_i \cdot f_i(X) + P_{i+} \cdot f_{i+}(X) + P_c \cdot f_c(X) \quad (1)$$

Subject to

$$Rn_{jk} \leq \sum_{i=1}^n x_{ijk}, (j=1, \dots, t; k=1, \dots, p); \quad (2)$$

$$x_{ijk} \leq D_{ijk}, (i=1, \dots, n; j=1, \dots, t; k=1, \dots, p); \quad (3)$$

$$\sum_{j=1}^t \sum_{k=1}^p (1 - x_{ijk}) \geq O_i, (i=1, \dots, n); \quad (4)$$

$$\left(\sum_{j=1}^t \sum_{k=1}^p (1 - x_{ijk}) \right) - O_i \leq A_i, (i=1, \dots, n); \quad (5)$$

$$x_{ijk} \in \{0, 1\}, (i=1, \dots, n; j=1, \dots, t; k=1, \dots, p); \quad (6)$$

multiplied by their respective weights. The hard constraints are guaranteed by the formulations of the equations (2) to (6). The restriction (2) guarantees the requirements of minimum covering. The technician's availability is determined by the restriction corresponding to equation (3). The restriction (4) checks the number of obligatory days off granted. The equation (5) guarantees that the concession of days off does not surpass the number of obligatory days off plus the number of additional ones. Finally, the restriction (6) defines the binary variable x_{ijk} .

In order to divide the problem into sub-problems by reducing the search space (Burke *et al.*, 2004b), and to obtain computational gains and better quality of the final solution, the schedules are developed separately by ward/shift since the teams are specific to each ward and shift. The monthly roster has the structure of a two-dimensional matrix, where each line corresponds to a technician's schedule and the columns represent the days of the planning period. The problem consists of optimally assigning a working shift or a day off to each employee on each day, according to several requirements (Bellanti *et al.*, 2004).

Architecture

The process begins with the generation of an initial schedule using informed parameters: planning period (month/year), number of technicians, minimum and ideal coverings, allowed relaxations and weights of the OF variables. This schedule will be the initial solution for Tabu Search (TS), which is the method suggested by Glover (1989, 1990; Glover and Laguna, 1997).

The resulting schedule obtained for the TS is submitted to the Genetic Algorithm, as a diversification strategy. This architecture is based on the work of Burke (Burke *et al.*, 1998). Burke (Burke *et al.*, 2001) suggests the application of TS with diversification heuristics and, after one more diversification stage, with GA. According to Burke (Burke *et al.*, 2004b), by using this TS/GA procedure efficiency can be increased. This cycle ends when the informed number of diversifications is reached or when GA does not get to alter the schedule. Figure 1 illustrates this dynamics. The objective is to obtain just a feasible schedule, where only the hard constraints are observed.

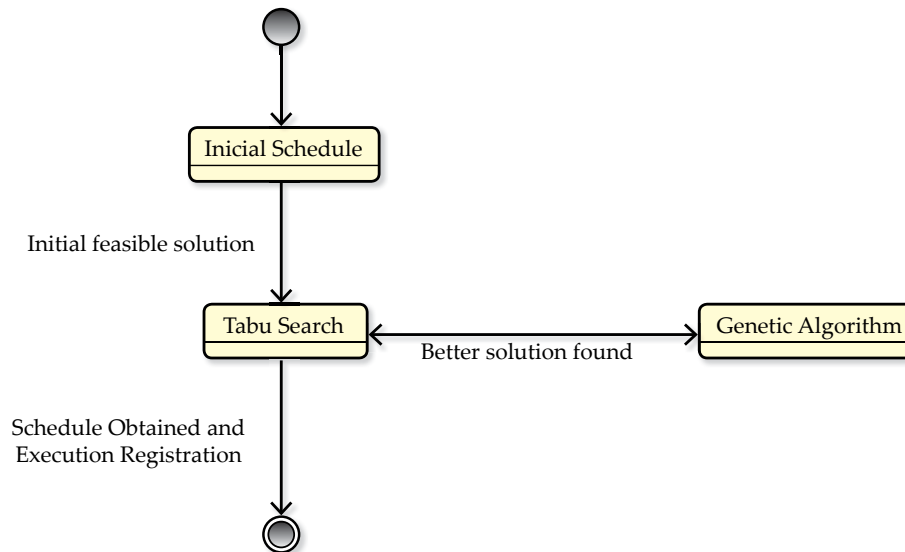


Figure 1. UML State Diagram: TS and GA Metaheuristics.

At the end of the process, the system presents the best schedule obtained, showing where it is suitable, every day, whether the technicians will be in attendance or in rest. They are also exposed to the percentile of improvement obtained in the objective function in relation to the initial schedule and the value of each portion that composes the objective function, among other results.

Tabu Search

Tabu Search was proposed by Glover (1989) and its concepts were detailed by Glover and Laguna (1997). It's a technique to solve optimization combinatorial problems (Glover, 1989) that consists of an iterative procedure to generate neighborhoods, avoiding getting stuck in locals optimums. The searching process of the Tabu Search to find best solutions is realized through an aggressive exploration (Glover and Laguna, 1997), choosing the best movement for each iteration, regardless of whether this movement improve the value of the current solution.

In the Tabu Search development, intensification and diversification strategies are alternated through the Tabu attributes analysis. Diversification strategies direct the search to new regions, aiming to reach a whole search space, while the intensification strategies reinforce the search in the neighborhood of a historically good solution (Glover and Laguna, 1997). The stopping criterion makes it possible to stop the search. The stopping

criterion can be defined as the interaction in which the best results were found or as the maximum number of iterations without an improvement in the value of the objective function. The Tabu list is a structure that contains some solution attributes that are considered Tabu. The objective of this list is to forbid the use of some solutions during a defined period of time.

In the proposed model, for each iteration of Tabu Search (Glover, 1989, 1990; Glover and Laguna, 1997) two neighborhoods are examined. The first neighborhood acts as a small diversification schedule since its movements alternate schedules of a technician/day, between day off and duty. As in the construction of the initial solution, only the obligatory days off of the period are observed.

This movement is performed only for technicians with the right days off obtained. Figure 2 illustrates the diversification movements of the first neighborhood.

At each iteration TS is evaluated, for all the technicians, if the number of days off granted is lower than the number of days off required plus the number of days off obtained, i.e., if the technician is entitled to more days off, it is checked which day should be the day off in order to minimize the objective function. Afterwards, it is checked whether the number of days off granted to the particular technician is greater than the number of days off required. If this is the case, there is an assessment of the impact on the objective function caused by the replacement of one day off for

	Days				
Name	16	17	18	19	20
Tech. 1					off
Tech. 2					
Tech. 3		off		off	
Working Tech. number	3	2	3	2	2

(a) Before the movement

	Days				
Name	16	17	18	19	20
Tech. 1	off				off
Tech. 2					
Tech. 3		off		off	
Working Tech. number	2	2	3	2	2

(b) After the movement

Figure 2. First Neighborhood: diversification movement.

work day. Therefore, between these two possibilities, giving a day off or taking a day off, the movement that most minimizes the value of the objective function is performed in the schedule. But the feasibility requirements must always be met: the minimum days off required, the minimum coverage, the number of days off granted may not exceed the sum of days off required and days off obtained for each technician.

The second neighborhood is generated by movements of exchange (swap). These movements evaluate all exchanges between the different schedules (day off or duty) of each technician in different days, according to the availability of the technician and the schedule that result in a viable solution. The movements seek to find the coverage requirements and the number of days off granted to each technician, in such a way that the number of days off is not lower than the number of days off required or greater than the number of days off required plus the additional gaps. At each iteration, the movement performed in the schedule is the one that results in a lower value in the objective function.

The Tabu List, which contains the movements considered forbidden to prevent the search from returning to a solution already assessed, will contain the two types of movements. So in this Tabu application there are two Tabu Lists, one for the generation of each neighborhood.

Genetic Algorithm

The term genetic algorithm was first used by John Holland (1992), whose book *Adaptation in Natural and Artificial Systems* of 1992 was instrumental in creating a field of research and application that at present is much wider than the original GA. In this model, Genetic Algorithm is applied as a diversification policy that is more aggressive than the diversification movements accomplished in Tabu Search. While the movements implemented for Tabu Search alter a technician/day schedule, the Genetic Algorithm changes several schedules (technician/day) aiming to reach areas of the search space of solutions that the Tabu Search movements does not reach. The efficiency of Tabu Search can be increased by applying diversification strategies (Burke *et al.*, 2004b). From the schedule resulting from Tabu Search, the initial population is selected, and each technician's schedule is a potential candidate to individual of this population. The individuals with more than one requested day off not granted or with the weekend constraint not observed are selected.

Each individual schedule is a chromosome and the day is a gene. In order to avoid an excessive time of processing, the performance of the Genetic Algorithm is restricted to a search for improvements in the individual schedules that are not considering the restrictions considered more important, one for the general objective of the schedule and others for the legal aspects.

The population size parameter limits the maximum number of individuals that will be selected. The Genetic Algorithm continues if at least two individuals are selected. For each iteration, two parents of the initial population that have not crossed among themselves, called Parent 1 and Parent 2, are selected. An incidence matrix is maintained to manage the crossings already accomplished. The parameter number of generations indicates how many crossings should be accomplished. For each generation, a new random crossover point is generated and the parents are combined to generate children.

After the generations, it is checked whether any child can replace Parent 1 with a decrease in the objective function value. If this is the case, the parent is replaced by the child that promotes the largest reduction in the objective function value and the parent is eliminated from the initial population; otherwise the parent

stays for future crossings. This same substitution process is repeated for the Parent 2. All of the generated children are discarded and the algorithm returns to select two parents from the initial population.

The crossing process ends when there is no longer a minimum of two individuals in the initial population that have not crossed among themselves.

Computational Experiments

The experiments were made in a computer with an Intel 1.66 GHz Duo Core processor and 1 Gb of RAM, with the Windows XP Professional 2000 Service Pack 2 operating system at normal processing speed. The application was developed in a thread, just using one processor. As cases of tests the experiments used generators of random cases of test as in Özcan (2005), artificial cases based on real data as in Dowsland (1989) or data of real cases as in Tsai and Li (2009), Burke *et al.* (1998, 2001, 2003) and Osogami and Imai (2000). In this work, a case of artificial test was used based on real data collected at a large Brazilian hospital. The following conditions were applied in all of the experiments: 20 technicians, minimum coverage of 14 technicians and allowing the relaxation in the coverage of one technician. The size of the Tabu List was 10 and the maximum number of iterations in Tabu Search without improvement (*nbmax*) was 50, the interval between rests was a maximum of six days and a minimum of four days. To obtain a non-tendentious solution (NTS), in which each decision variable contributes in the same way as the others to the objective function, the weights of the decision variables of the objective function were gauged, normalizing the values that each portion can take on. In this stage, the best result was obtained with *nbmax* equal to 50 and Tabu List containing 30 movements, and an improvement of 70,88% was obtained. Starting from this calibration, experiments were accomplished activating the diversification policy for GA, to evaluate the behavior of the objective function value and the time of processing. Due to the random generation of the cut points, each test was executed 10 times and the arithmetic average of the obtained results was considered.

Non-tendentious Solution

To obtain a NTS the weights were gauged to normalize the values of each decision vari-

able. Altering the values of the decision variables' weights is possible in order to annul or to privilege the contribution of the decision variable in the objective function value. In order to evaluate the behavior of the other variables when one of them is privileged, the model was executed 6 times for each variable. The value of the NTS weight of the privileged variable was multiplied by 1, 2, 5, 10, 50 and 100 at each execution. The most significant behaviors observed were these:

- When privileging the variables Additional days off and Requests, the variable Ideal covering is harmed;
- Favoring the variable Fairness harms the variable Weekend and Additional days off. As the variable Ideal covering has an inversely proportional behavior to the variable Additional days off, it is favored within the variable Fairness.

The influence of Monthly Variation

The problem discussed here refers to the distribution of the days off. The number of days off required by law is the number of Sundays plus the number of holidays that occur in every planning period. Thus, it is possible to notice the influence of the planning period on the schedule. To study this, we calculated an index called "degree of difficulty" (DD) for each planning period. It was calculated as a proportion of the number of days off by the number of "no days off" (a number of days in a planning period minus the number of days off of the planning period). The more Sundays and holidays there during the period, more days off should be distributed and the more difficult it is to respect the covering constraints. Another factor is the number of days of the planning period, which can be from 28 to 31 days. The longer the period, the easier it is to distribute the days off and the larger is the solution research space. The amount of Saturdays in a month, besides Sundays and holidays in the period, also exercises a certain influence. A larger number of Sundays and holidays hinders the meeting of the covering constraint. On the other hand, the larger the number of Saturdays, Sundays and holidays, the easier it is to deal with the Weekend and Fairness constraints. The behavior of the objective function obtained in relation to the degree of difficulty is presented in Figure 3. The line refers to the degree of difficulty associated to the schedule on the right.

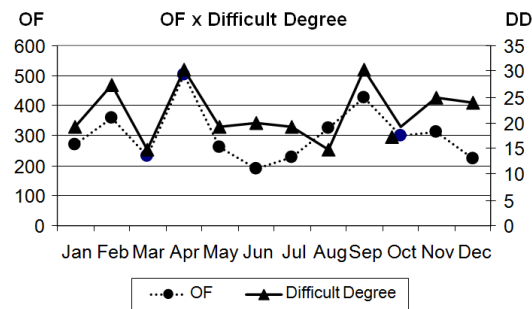


Figure 3. Objective function and behavior of the degree of difficulty.

The correlation of the objective function (OF) to the degree of difficulty is 0.72. Although the correlations of some variables with the degree of difficulty are inverse, the tendency is a directly proportional relationship. For the other variables, the correlations located in the interval from -0.7 to 0.7 are considered less significant. The correlation between the degree of difficulty and the Additional days off variable is of 0.9354, and the correlation between the degree of difficulty and the variable Ideal covering is of 0.9129. So, the correlations between the variables *Requests* and *Fairness* and the degree of difficulty show an inversely proportional behavior, and are of -0.7434 and -0.8286, respectively. It may be observed that the number of Saturdays, Sundays and the variable holidays of each month is different from one year to another, and the degree of difficulty may vary too.

Experiments with TS and GA

The following conditions were applied in all of the experiments: 20 technicians, but one technician was unavailable for the whole period due to a leave and three professionals were on vacation for fifteen days. The ideal covering is 14 technicians, allowing the relaxation of one technician. The planning period used was January 2007 and the population size was of 20. Each technician's specific information, such as days off requested, last day off of the previous period, number of additional days off and availability are supplied by the test case, simulating data that, in practice, would be obtained from a database. In previous experiments designed to calibrate the TS, the best results were obtained using a Tabu List size equal to 30 and 50 iterations without an improvement in the objective function value. In this case, the improvement of the objective function obtained

was 70,89%. Tables 2 and 3 show the calibration process considering the variation in the Taboo List size and the number of iterations without improvement in the objective function value (nbmax).

In Table 2, we note that with a Tabu List size 20 we obtain the best results and the lowest number of iterations. The solicitations variable is favored when the interval + variable is favored with increasing Tabu List. Lists with larger sizes lead the search to other regions where better solutions might be found.

As shown in Table 3, the intensification in the search promoted by increasing the NBmax value favors only the variable interval +.

Due to the use of random cross points for analyzing the GA results, each test was executed 10 times and the arithmetical average of the results was used.

When using only one cross point, 4 children are created for each generation. The best result reached (75.13%) was almost 6.0% better than the value obtained from the objective function without using the GA. It is not necessary to have a large number of generations in order to get better results. The cross point must be a number from 1 up to 30, depending on the planning period and whether both parents are available throughout the period. In a large number of generations the cross point tends to repeat, which does not promote improvements and does not guarantee a better result, as can be seen in Figure 4. Even when executing more than 20 generations, better results were not obtained. With 150 generations a large number of TS-GA cycles were executed, but the value of the objective function obtained was not better. Besides, there is a significant correlation between an objective function value and the number of TS-GA cycles: 0.79. The line refers to the cycle's average associated to the measures scale on the right.

When using two cross points, 6 children are created for each generation, increasing the possibility to get better results. As expected, some values of the objective function obtained reached up to 80.05% and 80.55% of improvement. The number of TS-GA cycles executed increased too. Sometimes it reached 6 cycles. In these tests the correlation between OF and number of TS-GA cycles is 0.81. Figure 5 presents these results.

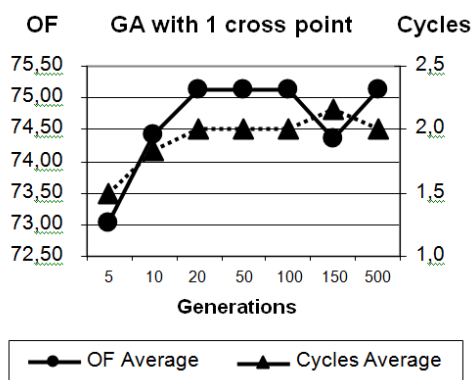
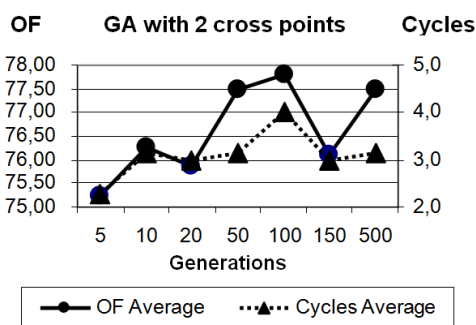
The line refers to the cycle's average associated to the measures scale on the right. It was also observed that a higher number of generations does not imply better results.

Table 2. Variation of Tabu List size.

Tabu List Size	10	30	50	100	500
Solicitation	27,45815	25,95565	25,95565	25,95565	25,95565
Distribution of shifts	27,13256	27,13256	27,13256	27,13256	27,13256
weekend	0	0	0	0	0
Interval	0	0	0	0	0
Interval +	103	111	112	112	112
Days off adquired	0	0	0	0	0
Covering	33,878	33,878	33,878	33,878	33,878
OF	191,4687	197,9662	198,9662	198,9662	198,9662
% improvement	75,01	74,16	74,03	74,03	74,03
Best iteration	104	40	40	40	40

Table 3. Variation of Nbmax.

Tabu List Size	10	30	50	100
Solicitation	27,45815	27,45815	27,45815	27,45815
Distribution of shifts	27,13256	27,13256	27,13256	27,13256
weekend	0	0	0	0
Interval	0	0	0	0
Interval +	108	103	103	103
Days off adquired	0	0	0	0
Covering	33,878	33,878	33,878	33,878
OF	196,4687	191,4687	191,4687	191,4687
% improvement	74,36	75,01	75,01	75,01
Best iteration	44	104	104	104


Figure 4. Results using one GA cross point.

Figure 5. Results using two GA cross points.

Model Validation

The model performance was tested with 20 technicians and a planning period of 28 to 31 days test case. The planning period was 14 days in Özcan (2005) and 28 days in Burke (Burke *et al.*, 1998, 2003). Both test cases were developed with 20 employees. The researched hospital does not have teams with more than 30 people per shift in a ward nowadays. The execution time was from 10 to 26 minutes, mostly 18 minutes, depending on the degree of difficulty of the planning period. This result is compatible with the times found for the 20 nurses case tested in Burke (Burke *et al.*, 1998, 2003).

For validation we used actual data from a public health institution in Brazil, rather than synthetic data. A real case was used to represent the situation of an actual schedule: 13 technicians, 2 absentees during the whole period, 31 day planning period, 4 Saturdays, 4 Sundays and 1 holiday. Information such as date of the last day off and number of additional days off was obtained from the schedule of the previous period. The value obtained for the objective function calculated for the

Table 4. Schedule evaluation.

	Obtained Schedule	Hospital Schedule
Requests	19.5507	3.98449
Fairness	0	32.76162
Weekend	0	57.8366
Interval	0	157.545
Interval +	10	28
Additional days off	33.3996	72.3658
Covering	33.878	59.2865
Objective Function value	96.8300	411.7800

manual schedule was of 411.7800 and the model obtained 96.8300, with the weights of the NTS. The model significantly reduced the penalties, including the days off requests, which is a very important requirement for the technicians. The execution time was 7'45". The obtained results are summarized in Table 4.

When activating the GA with one cut point, varying from 50 to 100 generations, and with two cut points, varying from 100 to 150 generations, improvements were not obtained. A significant reduction was observed in almost all the variables of decision, with the exception of the decision variable. It can be observed that the covering increased 14.29%. For a comparison with the result obtained with the NTS weights, the value resulting from the decision variable Requests was divided by 10 and the result was the value of 0.55, which represents 13,91% of the value of this decision variable in the real scale with NTS weights. In other words, the requests for rests were observed almost fully. When performing a test with this same real schedule, but privileging the decision variable Requests, and multiplying its respective NTS weights by 10, the result was a percentile of improvement of 74.45% in the objective function value in relation to the one obtained with the real schedule.

Conclusion

A model was proposed to solve the nurse rostering problem, in accordance with the operational rules of the hospitals and the constraints of the legislation and with a view to promoting a higher level of employee satisfaction with their schedules. It is an important tool for the administration of the most significant portion of health care area employees.

The model developed here combines Tabu Search with a Genetic Algorithm, exploring

diversification strategies executed by TS and promoted by the GA to obtain good results. The executed tests showed that the model is realistic and produces quality schedules in a reasonable computational time.

The TS showed an efficient performance once at each neighborhood exploration, and a high number of movements are executed, although few are feasible. The GA stage was a surprise in terms of the speed and the obtained improvement. Even using only one cross point the improvement is significant. Better results can be obtained with two cross points, but the processing time increases too, because the number of cycles TS-GA cycles executed is bigger. With one or two cross points, a bigger number of generations do not guarantee better results.

For the experiments we used test cases based on actual data. With the weights of the non-tendentious solution it was possible to obtain good results in relation to the manual schedules, considering that in practice these schedules are not even feasible. The model was validated with a real schedule and proved to be very superior.

The Nurse Rostering is a very complex combinatorial problem; administrators must define which variables must be privileged at each time. The proposed model offers this flexibility, allowing the alteration of the weights of the parameters that configure the accepted limits of relaxation.

A future task might be the realization of validations of the quality of the solutions obtained using public data sets.

References

- BARBOZA, A.O.; CARNIERI, C.; STEINER, M. T. A.; SIQUEIRA, P. H. 2003. Técnicas da Pesquisa Operacional no Problema de Horários de Atendentes em Centrais Telefônicas. *Gestão & Produção*, **10**(1):109-127. <http://dx.doi.org/10.1590/S0104-530X2003000100008>
- BELLANTI, F.; CARELLO, G.; DELLA, C.R.; TADEI, R. 2004. A Greedy-based Neighborhood Search Approach to a Nurse Rostering Problem. *European Journal of Operational Research*, **53**:28-40. [http://dx.doi.org/10.1016/S0377-2217\(03\)00096-1](http://dx.doi.org/10.1016/S0377-2217(03)00096-1)
- BURKE, E.K.; CAUSMAECKER, P.D.; BERGHE, G.V. 1998. A Hybrid Tabu Search Algorithm for the Nurse Rostering Problem in Simulated Evolution and Learning. *Lecture Notes in Artificial Intelligence*, Springer Verlag, **1585**:187-194.
- BURKE, E.K.; CAUSMAECKER, P.D.; BERGHE, G.V. 2001. A Memetic Approach to the Nurse Rostering Problem. *Applied Intelligence*, **15**:199-214. <http://dx.doi.org/10.1023/A:1011291030731>

- BURKE, E.K.; KENDALL, G.; SOUBEIGA, E. 2003. A Tabu-Search Hyperheuristic for Timetabling and Rostering. *Journal of Heuristic*, **9**:451-470.
<http://dx.doi.org/10.1023/B:HEUR.0000012446.94732.b6>
- BURKE, E.K.; CAUSMAECKER, P.D.; BERGHE, G.V.; LANDEGHEM, H.V. 2004a. The State of the Art of Nurse Rostering. *Journal of Scheduling*, **7**(6):441-499.
<http://dx.doi.org/10.1023/B:JOSH.0000046076.75950.0b>
- BURKE, E.K.; CAUSMAECKER, P.D.; BERGHE, G.V. 2004b. Novel Metaheuristic Approaches to Nurse Rostering Problems in Belgian Hospitals. In: E.K. BURKE; P.D. CAUSMAECKER; G.V. BERGHE, *Handbook of Scheduling: Algorithms, Models and Performance Analysis*. Leuven, CRC Press, p 44.1-44.18.
- CHUN, A.H.W.; CHAN, S.H.C.; LAM, G.P.S.; TSANG, F.M.F.; WONG, J.; YEUG, D.W.M. 2000. Nurse Rostering at the Hospital Authority of Hong Kong. In: 17th NATIONAL CONFERENCE ON AAAI, 12th CONFERENCE ON IAAI, 2000, Austin, *Proceedings...* Austin, p. 951-956.
- DOWLING, D.; KRISHNAMOORTHY, M.; MACKENZIE, H.; SIER, D. 1997. Staff Rostering at a Large International Airport. *Proceedings of Operations Research*, **72**:125-147.
- DOWSLAND, K.A. 1998. Nurse Scheduling with Tabu Search and Strategic Oscillation. *European Journal of Operational Research*, **106**:393-407.
[http://dx.doi.org/10.1016/S0377-2217\(97\)00281-6](http://dx.doi.org/10.1016/S0377-2217(97)00281-6)
- ERNST, A.; JIANG, H.; KRISHNAMOORTHY, M.; SIER, D. 2004. Staffing Scheduling and Rostering: A Review of Applications, Methods and Models. *European Journal of Operations Research*, **153**:3-27.
[http://dx.doi.org/10.1016/S0377-2217\(03\)00095-X](http://dx.doi.org/10.1016/S0377-2217(03)00095-X)
- ERNST, A.; JIANG, H.; KRISHNAMOORTHY, M.; SIER, D. 2004b. An Annotated Bibliography of Personnel Scheduling and Rostering. *Proceedings of Operations Research*, **127**:21-144.
- GASCON, V.; VILLENEUVE, S.; MICHELON, P.; FERLAND, J.A. 2000. Scheduling the Flying Squad Nurses of a Hospital Using a Multi-Objective Programming Model. *Proceedings of Operations Research*, **96**:149-166.
- GLOVER, F. 1989. Tabu Search – part 1. *ORSA Journal on Computing*, **1**(3):190-206.
- GLOVER, F. 1990. Tabu Search – part 2. *ORSA Journal on Computing*, **2**(1):4-32.
- GLOVER, F.; LAGUNA, M. 1997. *Tabu Search*. Boston, Kluwer Academic Publishers, 377 p.
- HOLLAND, J.H. 1992. *Adaptation in Natural and Artificial Systems*. Ann Arbor, Michigan, University of Michigan Press, 211 p.
- JAUMARD, B.; SEMET, F.; VOVOR T. 1998. A Generalized Linear Programming Model for Nurse Scheduling. *European Journal of Operational Research*, **107**:1-18.
[http://dx.doi.org/10.1016/S0377-2217\(97\)00330-5](http://dx.doi.org/10.1016/S0377-2217(97)00330-5)
- MEISELS, A.; SCHAERF, A. 2003. Modeling and Solving Employee Timetabling Problems. *Proceedings of Mathematics and Artificial Intelligence*, **39**(1):41-59.
- OSOGAMI, T.; IMAI, H. 2000. Classification of Various Neighborhood Operations for the Nurse Scheduling Problem. *Lecture Notes in Computer Science*, **1969**:72-83.
http://dx.doi.org/10.1007/3-540-40996-3_7
- ÖZCAN, E. 2005. Memetic Algorithms for Nurse Rostering. In: 20TH INTERNATIONAL SYMPOSIUM ON COMPUTER AND INFORMATION SCIENCES, 2005, Istanbul. *Proceedings...* Istanbul, Springer-Verlag. p. 482-492.
- TSAL, C.; LI, S.H.A. 2009. A Two-stage Modeling with Genetic Algorithms for the Nurse Scheduling Problem. *Expert System with Applications*, **36**:9506-9512.
<http://dx.doi.org/10.1016/j.eswa.2008.11.049>
- YEH, W. 2007. Using Simulation Technique and Genetic Algorithm to Improve the Quality Care of a Hospital Emergency Department. *Expert System with Applications*, **32**:1073-1083.
<http://dx.doi.org/10.1016/j.eswa.2006.02.017>

Submitted on May 05, 2012.

Accepted on December 06, 2012.