

# Nurse Rostering with Soft Constraints

## *Evidence from Chilean Mid-size Health Care Centers*

Ricardo Soto<sup>1,2</sup>, Broderick Crawford<sup>1,3</sup>, Rodrigo Bertrand<sup>1</sup> and Eric Monfroy<sup>4</sup>

<sup>1</sup>*Pontificia Universidad Católica de Valparaíso, Valparaíso, Chile*

<sup>2</sup>*Universidad Autónoma de Chile, Santiago, Chile*

<sup>3</sup>*Universidad Finis Terrae, Santiago, Chile*

<sup>4</sup>*CNRS, LINA, Université de Nantes, Nantes, France*

**Keywords:** Rostering, Constraint Programming, Heuristic Search.

**Abstract:** Nurse rostering deals with the shifts arrangements of nursing staff in the daily operation of health care centers. The design of suitable rosters for nurses is known to be particularly complex due to the number of interrelated requirements that must be considered. The literature reports a wide list of works devoted to solve such a problem. The techniques used range from classic methods such as linear programming to more modern incomplete methods such as evolutionary computing. However, most works are centered on the performance of techniques for solving well-known instances of nurse rostering. In this paper, we focus on a real case study of nurse rostering. The solution is devoted to a set of mid-size Chilean hospitals that use a very uncommon shift pattern due to proper country legal regulations. We present a new model involving hard and soft constraints that can be applied generically to any Chilean health care center.

## 1 INTRODUCTION

Nurse rostering consists in producing a schedule of shift assignments for a given period of time satisfying a set of constraints. The problem is known to be challenging mainly because the number of constraints that must be satisfied. Some constraint involved are related to the hospital management policies, government regulations, minimum area allocation, different required nurse skills, as well as the fairness among nurses. Different commercial and generic solutions have been proposed to tackle this problem. However they usually fail to satisfy the given requirements due to the wide variety of constraints from one scenario to another. Health centers therefore try to design tailored solutions in order to satisfy their specific policies and operation mechanisms.

This paper extends previous work (Soto et al., 2013) done on the study of nurse rostering for mid-size Chilean health care centers. We take as sample three clinics and four hospital that collect the main features of mid-size Chilean health care centers:

- Size: 30~50 nurses, 5~10 chief nurses, and receive 1000~2000 patients per month.

- Organization: Urgency, Medical Center, Operating Rooms, Intensive-care Unit, and Hospitalization.
- Shift pattern: the “fourth shift” is used, which is a very uncommon working shift system.
- The main administrative procedures are computationally handled. However, nurse rostering is usually done manually.

Nurse rostering has been largely studied from the seventies. Classic and exact methods such as linear, integer, and goal programming were early used to solve such a problem (Semet et al., 1998; Miller et al., 1976; Warner and Prawda, 1972). From the nineties, metaheuristics have taken an important place in the rostering research. Different solutions have been proposed, some examples use tabu search (Dowland, 1998; Burke et al., 1999), simulated annealing (Thomson, 1996), genetic algorithms (Aickelin and Dowland, 2001; Aickelin and Dowland, 2004; Bai et al., 2010; Maenhout and M. Vanhoucke, 2011), scatter search (Burke et al., 2010), electromagnetic metaheuristics (Maenhout and Vanhoucke, 2007), and variable neighborhood search (Burke et al., 2008; Lü and Hao, 2012).

In the context of complete search, constraint programming has been also strongly involved in nurse rostering, some examples can be seen in (Abdennadher and Schlenker, 1999; Bourdais et al., 2003; Pizarro et al., 2011) and (Métivier et al., 2009). Hybrids including constraint programming and local search, variable neighborhood search, and tabu search can be seen respectively in (Qu and He, 2009; Qu and He, 2008), and (Li et al., 2003).

As illustrated, a considerable number of relevant studies can be found in the topic. However most work is focused on the efficient solving of well-known instances, and the work devoted to real cases of nurse rostering is quite limited. This paper presents the study of a real case of nurse rostering in mid-size Chilean health care centers. The problem is modeled as a Constraint Satisfaction Problem (CSP) involving soft and hard constraints and solved with state-of-the-art Constraint Programming (CP) techniques via the JaCoP solver. We believe that this work will be useful to increase the experience in real cases of nurse rostering as well as to future NRP researchers.

This paper is structured as follows: Section 2 gives an overview of CP and related solving techniques. A real case of nurse rostering based on the study of mid-size health care centers is described and modeled in Section 3. The experiments are presented in Section 4. Finally, we conclude and give some directions for future work.

## 2 CONSTRAINT PROGRAMMING

Constraint Programming (CP) is a programming paradigm devoted to the efficient solving of constraint satisfaction and optimization problems. The main features of this paradigm are inherited from different and relevant domains of computer science such as operational research, artificial intelligence, and programming languages. In CP, a problem is formally stated as a Constraint Satisfaction Problem (CSP), which is defined by a triple  $N = \langle X, D, C \rangle$ :

- $X$  is a finite sequence of integer variables  $X = (x_1, \dots, x_n)$ .
- $D$  is the corresponding set of domains for  $X$ , that is,  $D = D(x_1) \times \dots \times D(x_n)$ , where  $D(x_i) \subset \mathbb{Z}$  is the finite set of values that variable  $x_i$  can take.
- $C$  is a set of constraints  $C = \{c_1, \dots, c_e\}$ , where variables in  $X(c_j)$  are in  $X$ .

A solution to a CSP is an assignment  $\{x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n\}$  such that  $a_i \in d_i$  for

$i = 1, \dots, n$  and the set  $C$  is satisfied.

### 2.1 CSP Solving

In CP, a CSP is commonly solved by using a forward checking algorithm that can be seen as a combination of a backtracking procedure with a filtering technique. The backtracking algorithm is responsible for exploring the combinatorial space while the filtering technique attempt to reduce it by deleting those values that do not lead to any solution. This is carried out in a process called constraint propagation where a local consistency is enforced. A local consistency is indeed a property that the problem must satisfy (a detailed explanation about constraint propagation can be found in (C. Bessière, 2006)).

---

#### Algorithm 1.

---

**Input:**  $C, \mathcal{D}$

- 1 **While**  $\neg$ success or failure **do**
- 2   **Variable\_Selection**( $\mathcal{D}$ )
- 3   **Value\_Selection**( $\mathcal{D}$ )
- 4   **Propagate** $_C(\mathcal{D})$
- 5   **If** empty\_domain\_in\_future\_var
- 6     **Shallow\_Backtrack**()
- 7   **If** empty\_domain\_in\_current\_var
- 8     **Backtrack**()
- 9 **End While**

---

Algorithm 1 illustrates a classic procedure for solving CSPs. It receives as input the set of constraints  $C$  and the sequence of domains  $D$  of the problem. Then, a while loop encloses a set of actions to be performed until a solution is reached or no solution is encountered. The first action to be performed is to select a variable and a value from its domain. This selection allows to build the potential solution to be verified whether is feasible. At line 4, the propagation procedure is triggered, which tries to filter the domains. Finally, two procedures are responsible for backtracking. The backtracking come back to the most recent viable state, and the shallow backtrack tries the following remaining value from the domain of the variable currently verified.

## 3 THE NRP MODEL

Mid-size health care centers in Chile employs a particular type of shift assignment called the “fourth shift” system. This system is quite uncommon do not

fitting with the common 8 hours-shift system used in most hospital and research papers. The “fourth shift” system considers two shifts of 12 hours per day as denoted in the following:

- Day Shift (D): starts at 8:00 AM and ends at 8:00 PM.
- Night Shift (N): starts at 8:00 PM and ends at 8:00 AM.

The “fourth shift” is used for nurses and paramedics, for chief nurses a different system applies to be presented in section 3.3. The complete set of health care centers uses almost the same constraints, obviously the preferences may vary from one center to another but this can be adapted with a slight modification on the preference constraints. In the following, we illustrate the constraints for nurses and chief nurses. The constraints for paramedics are omitted since they are analogous to the ones of nurses.

### 3.1 Constraints for Nurses

The “fourth shift” considers 12 hours shifts ordered in the following sequence:

Day 1: D, Day 2: N, Day 3: Off, Day 4: Off

where D corresponds to the day shift and N to the night shift. We consider a complete planning of 28 days (see Figure 1) to cover the lowest common multiple of 4 (the number of shifts within a cycle) and 7 (the number of week days). Then, taking into account a set of nurses  $\{1, \dots, nurses\}$ , a set of days  $\{1, \dots, 28\}$ , and a set of variables  $V_{i,j} \in \{0, 1, 2\}$ , where 0 denotes day off, 1 denotes day shift, and 2 denotes night shift, the constraints are modeled as follows:

- A day shift must be followed by a night shift which in turn is followed by an off day, for  $i \in \{1, \dots, nurses\} \wedge j \in \{1, \dots, 27\}$

$$(V_{i,j} = 1 \Rightarrow V_{i,j+1} = 2) \wedge \\ (V_{i,j} = 2 \Rightarrow V_{i,j+1} = 0)$$

- A night shift must be followed by two off days, which in turn must be followed by a day shift for  $i \in \{1, \dots, nurses\} \wedge j \in \{1, \dots, 26\}$

$$(V_{i,j} = 2) \Rightarrow (V_{i,j+1} = 0 \wedge V_{i,j+2} = 0) \wedge \\ (V_{i,j} = 0 \wedge V_{i,j+1} = 0) \Rightarrow (V_{i,j+2} = 1)$$

Now, in order to guarantee the required nurse allocation and off nurses per day the occurrences global constraint is included. The global constraint  $occurrences(Value, Vars, N)$  ensures that *Value* occurs *N* times in the *Vars* list. Let  $V_j$  denote the set of variables  $\{V_{1,j}, \dots, V_{nurses,j}\}$  for  $j \in \{1, \dots, 28\}$ . Then, the required constraints are as follows:

- $nurses/2$  nurses must be off per day  
 $occurrences(0, V_j, nurses/2)$
- $nurses/4$  day shifts per day  
 $occurrences(1, V_j, nurses/4)$
- $nurses/4$  night shifts per day  
 $occurrences(2, V_j, nurses/4)$

### 3.2 Soft Constraints

Soft constraints allow to model the general preferences of staff nurses. For instance, in the studied health-care centers, there exist nurses holding a senior position able to suggest some preferences on the schedule. As common CP solvers do not provide primitives to directly handle soft constraints, we model the soft constraints by using reified constraints. A reified constraint  $reified(c_k, B_k)$  assigns 1 (true) to the variable  $B_k$  if the constraint  $c_k$  is satisfied; and it assigns 0 (false) otherwise. Then, the idea is to maximize the sum of  $B_k$  variables as follows, assuming  $k \in \{1, \dots, p\}$ , where  $p$  is the number of preferences.

$$\text{maximize } \sum_{k=1}^p B_k$$

Then, the constraint satisfaction problem is naturally transformed into an optimization problem. As an example, let us consider three preferences:

- $c_1$ : Nurse 5 prefers not to be off on day 1
- $c_2$ : 2 and 4 prefer to start with a day shift
- $c_3$ : 11 prefers to be off the third weekend of the cycle

Those constraints can be modeled as follows:

- $c_1 : V_{5,1} \neq 0$
- $c_2 : V_{2,1} = 1 \wedge V_{4,1} = 1$
- $c_3 : V_{11,20} = 0 \wedge V_{11,21} = 0$

Then, we impose three reified constraints and the corresponding objective function.

$$reified(c_1, B_1) \wedge \\ reified(c_2, B_2) \wedge \\ reified(c_3, B_3) \\ \text{maximize } \sum_{k=1}^3 B_k$$

	1	2	3	...	26	27	28						
1	0	0	1	2	0	0	...	1	2	0	0	1	2
2	0	0	1	2	0	0	...	1	2	0	0	1	2
	1	2	0	0	1	2	...	0	0	1	2	0	0
⋮	⋮												
nurses	0	1	2	0	0	1	...	2	0	0	1	2	0
	2	0	0	1	2	0	...	0	0	1	2	0	0

1	2	0	0	1	2	0	...	0	0	1	2	0	0
2	2	0	0	1	2	0	...	0	0	1	2	0	0
	1	2	0	0	1	2	...	1	2	0	0	1	2
⋮	⋮												
chief_nurses	1	2	0	0	1	2	...	1	2	0	0	1	2
	1	1	0	2	0	2	...	1	1	0	2	0	2

Figure 1: Example of roster generation.

### 3.3 Chief Nurses Constraints

Most of health care centers need also chief nurses, which are responsible for management and coordination of specific nurse tasks. However, chief nurses usually do not follow the same pattern that regular nurses. They must take either 4 D per week or 3 N per week. Then in order to balance the total amount of nurses per day, for a chief nurse  $i \in \{1, \dots, chief\_nurses\}$  and a day  $j \in \{1, \dots, 28\}$  the following set of constraints are required.

- Let  $X_j$  denote the set of variables  $\{X_{1,j}, \dots, X_{chief\_nurses,j}\}$  for  $j \in \{1, \dots, 28\}$ . If the occurrence of N is less than D for a determined day the chief nurse take a night shift. Then, for  $j \in \{1, \dots, 28\}$ :

$$\begin{aligned}
 & (\text{occurrences}(1, V_j, N) \wedge \\
 & \text{occurrences}(2, V_j, D) \wedge (N < D) \\
 & \wedge \text{occurrences}(2, X_j, S)) \Rightarrow (S = D - N)
 \end{aligned}$$

- If the occurrence of D is less than N for a determined day, the chief nurse take a day shift.

$$\begin{aligned}
 & (\text{occurrences}(1, V_j, N) \wedge \\
 & \text{occurrences}(2, V_j, D) \wedge (N > D) \\
 & \wedge \text{occurrences}(1, X_j, S)) \Rightarrow (S = N - D)
 \end{aligned}$$

- Four day shifts or three night shifts per week, and two days off, for  $j \in \{1, \dots, days\}$

$$\begin{aligned}
 & (\text{occurrences}(1, X_j, 4) \vee \\
 & \text{occurrences}(2, X_j, 3)) \wedge \\
 & \text{occurrences}(0, X_j, 2)
 \end{aligned}$$

### 4 EXPERIMENTS

In this section we describe the experimental evaluation of the presented approach. We have tested 10 instances of the problem by considering different number of nurses (see Table 1). Experiments have been performed on a 2.0 Ghz Intel Core2Duo T7250 with 2GB RAM computer running Windows 7 and the presented model has been implemented in the JaCoP solver. For each instance we provide the following information from left to right.

- The number of nurses of the instance.
- The number of constraints of the instance.
- The number of visited nodes of the search tree during the solving process.
- The number of decision taken during the searching.
- The maximum depth of the tree reached.
- The maximum depth of the tree reached.
- The solving time.

The results illustrate an explosion of indicators when the number of nurses increases (see Fig. 2, 3, 4, and 5). The visited nodes, decisions, and solving time exhibit an obvious exponential growth as the problem belongs to the NP-Hard class. Despite of this exponential growth, the proposed solution satisfies the requirement of mid-size health care centers. Rosters are automatically generated in a reasonable amount of time (about 22 minutes for 40 nurses) instead of a manual generation as usually done in this kind of health centers.

Table 1: Statistics of solving process.

Nurses	Constraints	Nodes	Decisions	Depth	Solving time (ms)
4	433	12	11	7	31
8	729	34	29	14	32
12	1065	86	61	22	47
16	1381	306	177	31	94
20	1697	1422	741	40	343
24	2013	7378	3725	49	1623
28	2329	39734	19909	58	8299
32	2645	217186	108641	67	47471
36	2961	1196942	598525	76	243361
40	3277	6635010	3317565	85	1365273

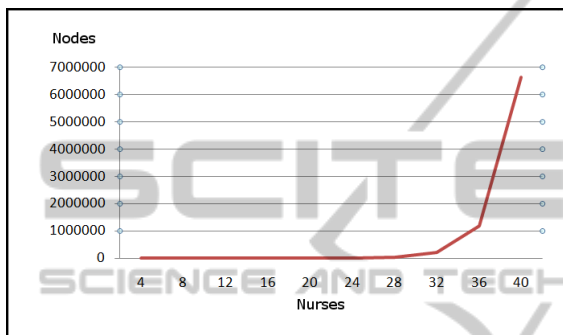


Figure 2: Visited nodes during the solving process of different instances of the problem.

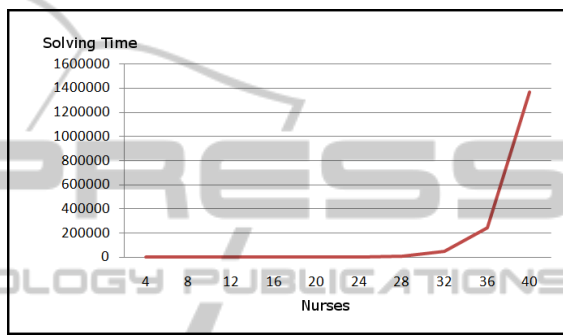


Figure 4: Solving times for different instances of the problem.

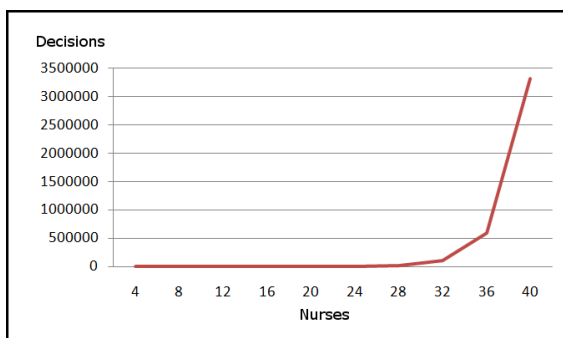


Figure 3: Decisions taken during the solving process of different instances of the problem.

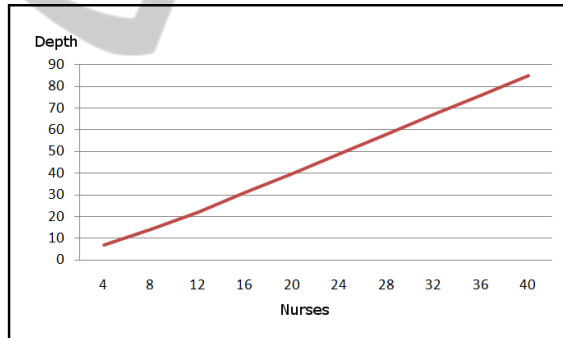


Figure 5: Depth of the tree reached during the solving process of different instances of the problem.

## 5 CONCLUSIONS AND FUTURE WORK

In this paper, we have addressed a study of nurse rostering for a set of Chilean mid-size health care centers. We have presented a model that handles a very uncommon shift pattern and the nurse preferences via soft constraints. Soft constraints have been modeled by using reified constraints in conjunction with an objective function in order to maximize the number of preferences satisfied. In this way, the model becomes an optimization problem that has been solved

in the JaCoP solver. We illustrated experimental results where the optimizer is able to solve the problem in a reasonable amount of time (about 22 minutes considering 40 nurses).

The solution introduced here is ongoing work, and it can clearly be extended by considering bigger health care centers and more complex soft constraints. Another interesting research direction to pursue is about the integration of autonomous search in the solving process, which in many cases has demonstrated excellent results (Crawford et al., 2013; Monfroy et al., 2013; Crawford et al., 2012; Soto et al., 2012; Crawford et al., 2011b; Crawford et al., 2011c; Crawford et al., 2011b; Crawford et al., 2011a).

## REFERENCES

- Abdennadher, S. and Schlenker, H. (1999). Nurse Scheduling using Constraint Logic Programming. In *Proceedings of AAAI/IAAI*, pages 838–843.
- Aickelin, U. and Dowsland, K. (2001). Exploiting Problem Structure in a Genetic Algorithm Approach to a Nurse Rostering Problem. *Journal of Scheduling*, 3(3):139–153.
- Aickelin, U. and Dowsland, K. A. (2004). An Indirect Genetic Algorithm for a Nurse-Scheduling Problem. *Computers & OR*, 31(5):761–778.
- Bai, R., Burke, E., Kendall, G., Li, J., and McCollum, B. (2010). A hybrid evolutionary approach to the nurse rostering problem. *IEEE Trans. Evolutionary Computation*, 14(4):580–590.
- Bourdais, S., Galinier, P., and Pesant, G. (2003). Hibiscus: A constraint programming application to staff scheduling in health care. In *Proceedings of CP*, volume 2833 of *LNCS*, pages 153–167. Springer.
- Burke, E., Causmaecker, P. D., and Berghe, G. V. (1999). A Hybrid Tabu Search Algorithm for the Nurse Rostering Problem. In *Proceedings of SEAL*, volume 1585 of *LNCS*, pages 187–194. Springer.
- Burke, E., Curtois, T., Post, G., Qu, R., and Veltman, B. (2008). A hybrid heuristic ordering and variable neighbourhood search for the nurse rostering problem. *European Journal of Operational Research*, 188(2):330–341.
- Burke, E., Curtois, T., Qu, R., and Berghe, G. V. (2010). A Scatter Search for the Nurse Rostering Problem. *Journal of Operational Research Society*, 61:1667–1679.
- C. Bessière (2006). *Handbook of Constraint Programming*, chapter Constraint Propagation, pages 29–84. Elsevier.
- Crawford, B., Castro, C., Monfroy, E., Soto, R., Palma, W., and Paredes, F. (2012). A hyperheuristic approach for guiding enumeration in constraint solving. In *Proceedings of EVOLVE*, volume 175 of *Advances in Intelligent Systems and Computing*, pages 171–188. Springer.
- Crawford, B., Soto, R., Castro, C., and Monfroy, E. (2011a). Extensible cp-based autonomous search. In *Proceedings of HCI International*, volume 173 of *CCIS*, pages 561–565. Springer.
- Crawford, B., Soto, R., Castro, C., and Monfroy, E. (2011b). A hyperheuristic approach for dynamic enumeration strategy selection in constraint satisfaction. In *Proceedings of the 4th International Workshop on the Interplay Between Natural and Artificial Computation (IWINAC)*, volume 6687 of *LNCS*, pages 295–304. Springer.
- Crawford, B., Soto, R., Monfroy, E., Palma, W., Castro, C., and Paredes, F. (2013). Parameter tuning of a choice-function based hyperheuristic using particle swarm optimization. *Expert Systems with Applications*, 40(5):1690–1695.
- Crawford, B., Soto, R., Montecinos, M., Castro, C., and Monfroy, E. (2011c). A framework for autonomous search in the ecl<sup>i</sup>ps<sup>e</sup> solver. In *Proceedings of the 24th International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE)*, volume 6703 of *LNCS*, pages 79–84. Springer.
- Dowsland, K. (1998). Nurse Scheduling with Tabu Search and Strategic Oscillation. *European Journal of Operational Research*, 106:393–407.
- Li, H., Lim, A., and Rodrigues, B. (2003). A hybrid ai approach for nurse rostering problem. In *Proceedings of SAC*, pages 730–735. ACM Press.
- Lü, Z. and Hao, J.-K. (2012). Adaptive neighborhood search for nurse rostering. *European Journal of Operational Research*, 218(3):865–876.
- Maenhout, B. and Vanhoucke, M. (2011). An evolutionary approach for the nurse rostering problem. *Computers & OR*, 38(10):1400–1411.
- Maenhout, B. and Vanhoucke, M. (2007). An electromagnetic meta-heuristic for the nurse scheduling problem. *J. Heuristics*, 13(4):359–385.
- Métivier, J.-P., Boizumault, P., and Loudni, S. (2009). Solving Nurse Rostering Problems Using Soft Global Constraints. In *Proceedings of CP*, volume 5732 of *LNCS*, pages 73–87. Springer.
- Miller, H., William, P., and Gustave, J. (1976). Nurse Scheduling Using Mathematical Programming. *Operations Research*, 24(5):857–870.
- Monfroy, E., Castro, C., Crawford, B., Soto, R., Paredes, F., and Figueroa, C. (2013). A reactive and hybrid constraint solver. *Journal of Experimental and Theoretical Artificial Intelligence*, 25(1):1–22.
- Pizarro, R., Rivera, G., Soto, R., Crawford, B., Castro, C., and Monfroy, E. (2011). Constraint-based nurse rostering for the valparaíso clinic center in chile. In *HCI International Conference 2011*, volume 174 of *CCIS*, pages 448–452. Springer.
- Qu, R. and He, F. (2008). A hybrid constraint programming approach for nurse rostering problems. In *Proceedings of SGAI*, pages 211–224.
- Qu, R. and He, F. (2009). Constraint-directed local search to nurse rostering problems. In *Proceedings of the 6th International Workshop on Local Search Techniques in Constraint Satisfaction (LSCS)*, pages 69–80.
- Semet, F., Vovor, T., and Jaumard, B. (1998). A Generalized Linear Programming Model for Nurse Scheduling. *European Journal of Operational Research*, 107:1–18.
- Soto, R., Crawford, B., Bertrand, R., and Monfroy, E. (2013). Modeling nrps with soft and reified constraints. In *Proceedings (To appear) of the 2013 AASRI Conference on Intelligent Systems and Control (ISC)*.
- Soto, R., Crawford, B., Monfroy, E., and Bustos, V. (2012). Using autonomous search for generating good enumeration strategy blends in constraint programming. In *Proceedings of the 12th International Conference on Computational Science and Its Applications (ICCSA)*, volume 7335 of *LNCS*, pages 607–617. Springer.
- Thomson, G. (1996). A Simulated Annealing Heuristic for Shift-Scheduling Using Non-Continuously Available Employees. *Computers and Operations Research*, 23:275–288.
- Warner, D. and Prawda, J. (1972). A Mathematical Programming Model for Scheduling Nursing Personnel in a Hospital. *Management Science*, 19(4):411–422.