

## **ATC work shift scheduling using multistart simulated annealing and regular expressions**

**F. Tello, A. Mateos, A. Jiménez-Martín, J.A. Fernández del Pozo**

Departamento de Inteligencia Artificial, Universidad Politécnica de Madrid

Campus de Montegancedo S/N, Boadilla del Monte 28660, Spain

{faustino.tello, alfonso.mateos, antonio.jimenez, juan.fdezpozo.salamanca}@upm.es

web-page: <http://www.dia.fi.upm.es/>

### **ABSTRACT**

In this paper, we propose a new approach for solving the air traffic controller (ATC) work shift scheduling problem, which minimizes the number of ATCs required to cover a given airspace sectoring, while satisfying a set of ATC labor conditions. This optimization problem belongs to the class of timetabling problems. The size and complexity of these combinatorial problems make them hard or even impossible to solve with exact methods.

In the proposed approach, initial feasible solutions are first built using a heuristic based on optimized templates, and then multistart simulated annealing is used to reach optimal solutions. In the search process, we use regular expressions to check the feasibility of the generated solutions. This provides high testing speed and modularity for a clear and maintainable implementation of the optimization model. Once the optimal ATC number is reached in one or more solutions, they are used as the initial solutions for a new optimization process aimed at balancing the ATC workloads.

The proposed approach is illustrated using a real example, and the optimal solution reached outperforms the reference solution, i.e. a real solution derived from the currently used tools based on templates. Indeed, one less ATC is needed to cover the airspace sectoring, and the ATC workloads are more balanced.

**Keywords:** Work Shift Scheduling Problem, Air Traffic Control, Simulated Annealing, Regular Expressions.

## INTRODUCTION

The core of air traffic controller (ATC) activity is to facilitate the airspace and airport surface traffic flow under their responsibility, while avoiding collisions between aircrafts. To satisfy this essential safety constraint, they must detect and solve possible conflicts among trajectories.

The airspace is divided into sectors. These sectors are operated by two ATC working positions (executive and planner). All the sectors open at any one time must cover all the airspace. This is referred to as sectorization. The sectorization changes throughout the day depending on the aircraft traffic. A higher volume of air traffic involves opening more sectors and, consequently, more ATCs are necessary.

The sectorization required to handle the traffic estimations for a period can be designed beforehand. Therefore, a very important problem in air traffic control is to determine the minimum number of ATCs necessary to cover a sectorization structure for a given period, denoted as airspace sectoring, while satisfying certain strong constraints accounting for ATC labor conditions and real-time requirements for tactical decision level at the air control room.

This optimization problem belongs to the class of timetabling problems, which require the assignment of times and resources to events, considering sets of required and desirable constraints. The size and complexity (constraints) of these combinatorial problems make them hard or even impossible to solve with exact methods, like linear or constraint programming; or population-based metaheuristics, like genetic algorithms or particle swarm optimization.

Different problem-solving approaches have been proposed in the literature to deal with timetabling problems [1]. Regarding the timetabling problem in the context of ATM, an overview of available results related to ATC shift scheduling is presented in [2]. There are already some software tools for generating ATC schedules [3], and their advantages and drawbacks have already been recognized [4]. Some are in-house tools, and details are not available for most of them. Therefore, it is necessary to build a decision support system to solving the ATC work shift scheduling problem, which minimizes the number of ATCs required to cover a given airspace sectoring, while satisfying a set of ATC labor conditions.

In this paper, we propose a novel methodology, wherein a heuristic is used to build initial feasible solutions and then multistart simulated annealing (SA) is used to reach an optimal solution. Regular expressions (Regex) [5] are used to check the feasibility of the visited solutions in the search process. The benefits of using Regex are high testing speed and modularity for a clear and maintainable implementation of the optimization model.

## PROBLEM DESCRIPTION

One of the core tasks of ATCs is to avoid collisions among aircrafts. But there are two types of ATC. The executive ATC talks to the aircraft and gives instructions to the pilots to avoid conflict situations, whereas the planner ATC is responsible for anticipating possible conflicts and communicating with the executive ATC to solve the situation before it happens.

As cited before, the sectorization changes throughout the day depending on the aircraft traffic. The sectorizations needed to handle the traffic estimations for a time period (usually a day) can be designed beforehand. This is denoted as airspace sectoring. Thus, the airspace sectoring contains the sectors open over a twenty-four hour period.

Figure 1 shows an example of an airspace sectoring for Madrid Path1 in Spain. Each interval is associated with a configuration (1A,2A,4A...), where the integer value represents the number of open sectors and the letter refers to the sector configuration. Note that the 24-hour period has been divided into night (N)/morning (M)/afternoon (A) periods in Figure 1.

Besides, five different ATC shifts are considered: long morning (LMS) (5:40-14:00h.), morning (MS) (6:20-14:00h.), afternoon (AS) (14:00-21:20h.), long afternoon (LAS) (14:00-22:20h.) and night (NS) (21:20-6:20h.).

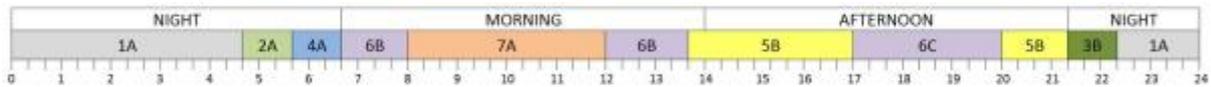


Figure 1: Madrid Path1 airspace sectoring

Moreover, a number of constraints accounting for ATC labor conditions have to be taken into account. In Spain, all these conditions were compiled and published in the Official State gazette (Boletín Oficial del Estado, BOE), Royal Decree 1001/2010, and Law 9/2010, regulating the provision of air traffic services. ATC labor conditions are as follows:

1. ATCs can cover a position (as executive or planner) in only one sector at any time.
2. Sectors must be covered by two ATCs (an executive and a planner ATC) during the time they are open.
3. ATCs must rest during 25% of the working shift in day shifts (MS, LMS, AS and LAS). A 33% resting time is established for the night shift (NS).
4. ATCs cannot work more than 2 consecutive hours.
5. Each rest period should last at least half an hour.
6. ATCs must remain in the same sector and position for at least half an hour (minimum time in a position).
7. Each ATC can work at most in two different sectors in the respective shift.
8. ATCs should occupy the planner (executive) position approximately 60% (40%) of the time.
9. ATCs cannot work more than 8 hours a day.
10. ATCs are assigned to one shift and cannot work in any other shift.
11. A sector change is not allowed without resting unless there is an emergency. ATCs can change to an affine (sector  $s_1$  and  $s_2$  are affine if  $s_1 \cap s_2 \neq \emptyset$ ) sector without resting, but are required to work in the new sector for at least 15 minutes. Otherwise, ATCs need to rest to perform the sector change.

The objective of the problem is to determine the minimum number of ATCs necessary to cover a given airspace sectoring while satisfying the above ATC labor conditions. Moreover, it would be interesting to balance the ATC workload, avoiding big differences among ATCs. It was necessary to solve an instance of the problem at most in 15 minutes.

## PROBLEM SOLVING METHODOLOGY

The proposed methodology is based on three elements, a heuristic to build initial feasible solutions, the use of Regex to check the feasibility of the solutions and a multistart SA to achieve optimal solutions.

Before describing the above elements in detail, let us first examine how the proposed methodology represents the solutions. We model the time discretely using a matrix containing 288 columns, i.e., the number of time slots under consideration. We consider time slots of 5 minutes since 5 is the greatest common divisor of 15 (11<sup>th</sup> ATC labor condition) and 20 (minimum period open a sector). Each row is associated with an ATC. The number of rows is established when applying the heuristic to build an initial solution. Note that initial solutions may have different numbers of ATCs.

Each element of the matrix  $(i, j)$  represents the state of the ATC  $i$  in the time slot  $j$ . We use the value 1 to represent a resting ATC, uppercase letters [A-Z] to point out that the ATC is working as an executive ATC in the corresponding sector, whereas lowercase letters [a-z] are used for planner positions. Finally, value 0 represents that the ATC is out of shift, i.e., neither working nor resting.

Colors are used to represent sectors when displaying solutions, see Figure 2. Resting periods and out of shift periods are in blue and white, respectively. The airspace sectoring in Figure 3 (11A, 13F, 11A) accounts for six hours rather than the whole day, and it is covered by 34 ATMs.

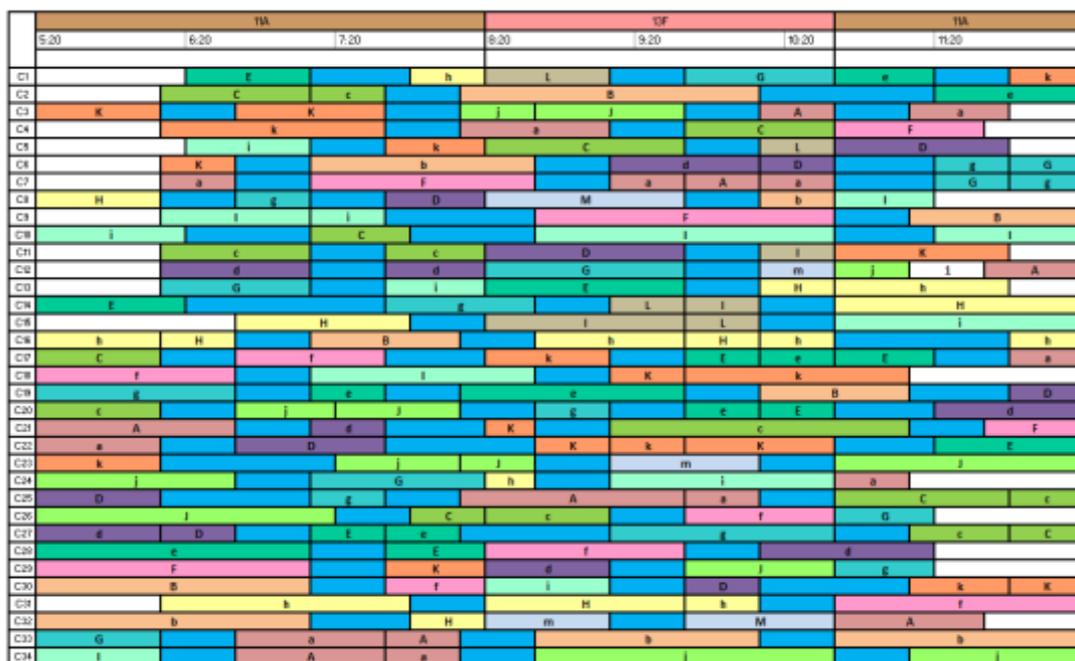


Figure 2: Solution representation

### A heuristic for the construction of initial feasible solutions

We propose a heuristic to build a set of initial solutions with different resting periods. They will be used afterwards in the multistart simulated annealing to reach an optimum solution. The proposed heuristic is based on the use of an optimized template, see Figure 3, with three ATCs covering a sector during 96 time slots (eight hours).

In the template, the duration of the working periods is always the same and are twice as long as the resting periods. Moreover, if different resting period durations are used, the heuristic will build different initial solutions, albeit with a similar structure. As the labor conditions establish that each resting period should last at least half an hour (six slots), and the minimum and maximum working periods are six and 24 slots, respectively, the duration of resting period must be between six and 12 slots. Thus, the heuristic build seven different initial solutions.

Note that some of the solutions output by the heuristic may not be feasible and, consequently, they are discarded as starting solutions for simulated annealing.

ATC 1	Working	Resting	Working	Working	Resting	Working	Working	...
ATC 2	Resting	Working	Working	Resting	Working	Working	Resting	...
ATC 3	Working	Working	Resting	Working	Working	Resting	Working	...

Figure 3: Template

## Regular expressions

A regular expression (Regex) [5] is a special text string for describing a search pattern in texts. Regexs are composed of metacharacters \, (), [], and {}; characters classes like \A (start of string), \s (white space) or \d (digit), and quantifiers \* (0 or more), + (1 or more) and ? (0 or 1). For instance, the pattern "car(s)?" matches the words "car" and "cars", and  $(\wedge+@[a-zA-Z]+\.[a-zA-Z]{2,6})$  could be used to match email addresses in a text, like Rose7@gmail.com.

In our ATC work shift scheduling problem, Regexs are used to check the feasibility of the solutions, i.e. the constraints representing the ATC labour conditions. For example, the labour condition that ATCs cannot work more than two consecutive hours can be verified using some patterns, i.e.:  $^0*([a-zA-Z]{1,24}1\{6,\}[a-zA-Z]{1,24}1\{6,\}[a-zA-Z]{1,24})(0*\$)$

This pattern accepts strings between 1 to 24 (lower and uppercase) letters (5 minutes  $\times$  24 slots = 2 hours) following at least six "1" characters.

Regexs are also used to check that ATCs work in only one shift, ATCs do not work more than 24 consecutive slots (two hours), ATCs work at least six consecutive slots in the same position, no position change is performed without resting, and all opened sectors are covered by two ATCs (executive and planner).

Other labor conditions, including constraints regarding the total resting time, affine sectors treatment and the maximum number of work positions for each ATC cannot be verified using Regexs. Their verification has been implemented in the problem-solving algorithm.

## Simulated annealing

SA [6, 7] is a trajectory-based metaheuristic which is named for and inspired by annealing in metallurgy. The basic idea of SA is as follows. An initial feasible solution is randomly generated. Then, in each iteration, a new solution is randomly generated from the neighborhood, of the solution considered in that iteration. If the new solution is better than the current one, then the algorithm moves to that solution. Otherwise, there is some probability of it moving to a worse solution. The acceptance of worse solutions makes for a broader search for the optimal solution and avoids trapping in local optima in early iterations.

The search is initially very diversified, since practically all moves are allowed. As the temperature drops, the probability of accepting a worse moves decreases, and only better moves will be accepted when it is zero.

The fitness function that we consider is maximizing  $(h_1 + 2h_2 + \dots + ch_c)/c^2$ , where  $c$  is the number of ATCs and  $h_i$  is the number of slots that the  $i$ -th ATC is working. The term  $1/c^2$  implies that when the number of ATCs decreases the fitness value is greatly improved. For a given number of ATCs, fitness values are higher for those solutions with a high workload for ATCs with higher indexes (in the last rows of the solution codification).

Thus, if we take into account the considered fitness function and the row reorganization of the visited solutions, we find that the search process tends to reallocate working periods from ATCs in the first rows to ATCs in the last rows (with the highest workload). In this way, we increase the resting periods among the first ATCs and tend to decrease the number of ATCs needed to cover the airspace sectoring.

The process for selecting a solution in the neighborhood of a given solution is as follows:

1. An ATC is selected at random. The aim is to reallocate some of this ATC's workload to another ATC.
2. A working period of the chosen ATC is selected at random. If it is composed of more than 12 slots, only the first 12 slots will be considered for reallocation, otherwise the whole working period is used.
3. We randomly select a second ATC and check if the whole working period for reallocation can be assigned to that ATC. If this is not possible, then we try with another ATC and so on. To do this, we check that the ATC under consideration is resting in the slots corresponding to the working period to be reallocated.  
If it is not possible to reallocate the considered working period to any ATC, then we reduce it by one slot (the last one) and repeat the process.  
We repeat the process until the working period is reallocated or is reduced to two slots, in which case, we go back to step 1.

## ILLUSTRATIVE EXAMPLE

We consider one airspace sectoring to illustrate the proposed methodology. There are 11 open sectors from 5:20 to 8:20, 13 open sectors from 8:20 to 10:40 and 11 from 10:40 to 11:20, see Figure 2.

First, we run the heuristic to build the initial feasible solutions. Seven initial feasible solutions are obtained using different templates. Then, multistart SA is carried out with the following parameter values: we run the algorithm proposed in [8] with different initial temperatures for adjustment using an acceptance ratio 0.9, leading to  $T_0=0.75$ . The number of iterations during which the temperature does not change is  $L = 500$  and we established  $\alpha = 0.9$ . The search stops when the fitness of the best solution does not improve by at least 0.05% during 1750 iterations.

The multistart SA reached seven solutions with the following number of ATCs {35, 34, 34, 35, 34, 34, 34}. To do this, we used a PC Intel i5-3230M CPU on 2.60GHz with 8GB of RAM running on Windows 10. It took 0.17 minutes to build the initial solutions and 3.5; 4; 9.6; 4.3; 13.2; 3.7 and 4.4 minutes, respectively, to reach the seven optimal solutions.

Note that the reference solution includes 35 ATCs, i.e. the solution yielded by the tools available before the proposed approach. The average ATC workload and the standard deviation are 280 minutes and 43.72 for the reference solution, respectively.

The optimal solution outperforms the benchmark since 34 ATCs are necessary to cover the airspace sectoring, with an average ATC workload of 288.23 minutes and a standard deviation 64.28. Looking at the standard deviation, however, we realized that the ATC workload is more balanced in the reference solution.

Using the optimal solution as the initial solution in the new optimization problem aimed at balancing the 34 ATC workloads, we obtain the solution shown in Figure 2, with a lower standard deviation, 16.17.

## CONCLUSION

We have proposed a new approach to solving the ATC work shift scheduling problem that minimizes the number of ATCs necessary to cover a given airspace sectoring while satisfying a set of ATC labor conditions according to Spanish regulations and real-time requirements. The approach consists of three elements: a heuristic to build initial feasible solutions, the use of Regex to check rapidly the feasibility of the solutions, and multistart SA to reach optimal solutions.

Once the optimal ATC number is reached, it is used as the initial solution of a new optimization process aimed at balancing the ATC workloads. The proposed approach has been illustrated using a real example, and the optimal solution reached outperforms the reference solution by one ATC and with more balanced workloads.

In this work, we have used simulated annealing for solving the corresponding optimization. However, we are currently implementing variable neighborhood search (VNS) and ant colony optimization (ACO) in combination with Regexs with the aim of comparing their performance.

**Acknowledgement.** The research reported in this paper was supported by Spanish Ministry of Economy and Competitiveness project MTM2014-56949-C3-2-R

## REFERENCES

1. J. Telhada, "Alternative MIP Formulations for an Integrated Shift Scheduling and Task Assignment Problem", *Discrete Applied Mathematics* 164, 328-343, 2014.
2. M. Arnving, B. Beermann, B. Koper, M. Maziul, U. Mellett, C. Niesing, J. Vogt, "Managing shiftwork in European ATM", Literature Review, European Organization for the Safety of Air Navigation, 2006.
3. EUROCONTROL, "Shiftwork Practices Study - ATM and Related Industries", DAP / SAF-2006/56 Brussels: EUROCONTROL, 2006.
4. EATCHIP Human Resources Team, "ATM Manpower Planning in Practice: Introduction to a Qualitative and Quantitative Staffing Methodology", HUM.ET1. ST02.2000-REP-01 Brussels: EUROCONTROL, 1998.
5. J.E. Friedl, "Mastering Regular Expressions", O'Reilly Media, Sebastopol, CA., USA 1997.
6. S. Kirkpatrick, C.D. Gelatt., M.P. Vecchi, "Optimization by Simulated Annealing", *Science* 220, 671-680, 1983.
7. V. Cerny, "Thermodynamical Approach to the Traveling Salesman Problem: An Efficient Simulation Algorithm", *Journal of Optimization Theory and Applications* 45, 41-51, 1985.
8. W. Ben-Ameur, "Computing the Initial Temperature of Simulated Annealing". *Computational Optimization and Applications* 29, 369-385, 2004.