Combining Metaheuristic Algorithms to Solve a Scheduling Problem

Mª Belén Vaquerizo García[1], Bruno Baruque Zanón[1], Emilio Corchado Rodríguez[2]

[1] Computer Languages and Systems Area, University of Burgos, Spain
belvagar@ubu.es, bbaruque@ubu.es
[2] Depart. de Informática y Automática, University of Salamanca, Spain
escorchado@usal.es

Abstract. The Labour Scheduling problem in the context of any transport company is a complex optimization problem of that belongs to the class of NP-Hard problems. In these cases, it is not viable to try to find an exact solution and therefore, they require methods that assure the optimal management of the available resources in the tracing of the work calendars under the most suitable criteria of economy of times and costs. The main purpose of this research is to propose an efficient method to determine optimal shifts in a generic transport company, using bio-inspired methods. This method employs a two-step approach to obtain a solution. In a first stage, a Grasp algorithm is used to generate a viable solution. Then in a second stage, this preliminary solution is tuned, in order to obtain an optimal one, by using a Scatter Search algorithm.

Keywords: Bus Driver Scheduling Problem; Evolutionary Algorithm; N-P Hard Problems; Grasp Algorithm; Scatter Search Algorithm

1 Introduction

Personnel scheduling is the problem of assigning staff members to shifts or duties over a scheduling period (typically a week or a month), so that certain constraints, organizational and personnel related, are satisfied [1].

This process normally consists of two different stages:

1. Staffing: Estimation of the number of drivers needed to cover the needs of working hours.
2. Scheduling: Construction of calendars of work to cover the estimation of drivers obtained in the stage of the staffing.

This study presents a method for solving this problem considering the limited time-availability of workers, among other constraints. Using the particular case of the organization of a transport company, the final objective of this work is the generation of calendars of shifts of work for drivers of a generic company, so the method used should be as flexible and adaptable as possible for any change or incident. These type of scheduling problems have been previously solved by means of metaheuristics algorithms, and among them, the Scatter Search algorithm [2, 3]. This study considers also a Scatter Search Algorithm to solve the problem while it also requires to adapt the algorithm to the characteristics of the problem to be solved. The remaining of this study is structured as follows: Section 2 summarizes previous efforts towards the solution of the presented problem. Section 3 describes the analysis of the problem to be solved. In Section 4, the method proposed for its resolution is presented. Section 5 includes the analysis of the results obtained. And, finally, Section 6 describes the conclusions and the future lines for this problem.

2 Previous Work

There are many different research works in the literature for solving the bus driver scheduling problem. Some of them [4, 5] use the set-covering formulation and Grasp, Simulated Annealing, Tabu Search or Genetic Algorithms to achieve a solution. Others [6, 7] present new mathematical models that represent all the complexity of the problem; considering in this way, different mathematical programming formulations for it.

Other works [8, 9, 10] apply different partitioning methods and set partitioning/covering model, one after the other, to solve huge instances at once. A common approach to deal with these huge instances is to split them into several smaller ones. Others [11] can solve huge instances without splitting, and combining Lagrangian heuristics, column generation and fixing techniques.

Some [12] consider a set of trips to be covered, and the goal consists in finding a driver-vehicle schedule that serves the maximum workload and optimizes several economic objectives while satisfying a set of imperative constraints. Other [13] propose a hybrid solution method, using an iterative heuristic to derive a series of small refined sub-problem instances fed into an existing efficient set covering/partitioning ILP model.

Other [2, 3] propose a Scatter Search Algorithm to solve this type of problem, because this algorithm is competitive and superior to other algorithms on most instances, especially in large-sized problems.

This work differs from the works examined in the literature in the way of constructing the final calendar, which minimizes the cost of the shifts programmed in the temporary horizon previously considered and satisfies the required demand. To do so, in a first step the construction of a preliminary solution is performed in a guided way by means of two constructive algorithms, obtaining a good starting point and satisfying a set of hard constraints. Later on, the Scatter Search Algorithm is applied to this initial solution, in order to obtain the final solution by improving the preliminary one.

The real number of resources is considered from the beginning, so the constructive algorithms can include the satisfaction of the major number of restrictions as a main goal. In this way, the set of solutions on which will be managed by the Scatter Search algorithm is constructed in guided way with two constructive algorithms, observing in the first phase all the hard restrictions and in the second phase the major possible number of soft restrictions. This does not affect significantly the component of diversification of the Scatter Search algorithm trying to improve the final solution.

To the knowledge of the authors, none of the previously presented methods include this way of generating a solution: first generating a valid one and then refining it, simplifying significantly the second phase, as non-valid solutions will no longer be considered in this last one. In the rest of methods detailed, non-valid solutions can be generated during the functioning of the evolutionary models; that have to be discarded later on. In this sense, the proposed solution tries to favour the intensification over the solution, trying to improve a valid one; rather than the diversification over it, trying to obtain a novel valid solution.

3 Analysis of the Problem to Solve

Bus driver scheduling problem is one of the most important planning decision problems that transportation companies must solve and that appear as an extremely complex part of the general transportation planning system [14].

The objective is to have a method to be used as an automatic tool to produce driver schedules, generating real and useful schedules that can be implemented without many manual adjustments or modifications. For this purpose, the formation of these shifts must consider a group of rules that are specific to each organization. These rules are usually derived from other national and local regulations, being obligatory or not.

Typically, there are constraints in the total time worked, in the total extension of the shift (duration between the start and the end of the shift), etc. So the problem involves several constraints related to labour and company rules and therefore can also present different evaluation criteria and objectives.

On the other hand, it must consider three different aspects: it should try to maximize the satisfaction of the drivers (Labour Agreement), optimizing at the same time the resources of personnel of the companies, by minimizing their costs (Personal Costs), and trying to cover the maximum demand required (Demand of Work).

3.1 Information Requirements

The information obtained about the needs of the real-life problem provides the initial parameters for the creation of the calendar. First, it is necessary to establish the main restrictions to be considered in this problem [13, 12].

So, there are various rules, preferences and requests to comply with when allocating shifts. These constraints on the problem can be divided into two groups. In the category of hard constraints there are those that must always be satisfied. Some of the hard constraints considered in this work are: shift type requirements, maximum number of assignments, maximum number of consecutive days, etc.

In contrast, it is also possible to consider a high number of soft constraints on the personnel schedules. This kind of soft constraints must be preferably satisfied, but violations can be accepted to a certain extent. It is highly exceptional in practice to find a schedule that satisfies all the soft constraints. These constraints have an associated penalty if not are satisfied, and this value would be added to the final value of the objective function for the solution obtained. Some of the soft constraints considered in this work are: minimum number of consecutive days, maximum number of consecutive free days, minimum number of consecutive free days, etc.

3.2 Driver Scheduling Problem Modelling

In this study, this is considered a real-life problem, with multiple pieces of information as data entries and multiple restrictions to satisfy. The solution must satisfy two opposite points of view: that of the company and that of its employees. The goal of the company is generating the work calendars with the minimum cost possible, whereas the goal of the worker is to obtain the major level of service possible respecting the highest number of soft restrictions as possible (which include their collective agreement or work conditions). Besides, an equitable distribution of the calendar of work with the rest of partners in the distribution of shifts of work is necessary. This requires achieving a level of commitment between both opposite goals.

For the transportation company, every driver has assigned a fixed cost per month. The main purpose of this study is to minimize the cost of the shifts programmed in the temporary horizon considered (day, week, month, etc.), with the restriction of having enough number of drivers in all periods of times to satisfy the demand required.

In the first stages of the solution, this problem is formulated as a minimization problem whose objective is to determine the minimum number of driver shifts necessary to cover the estimated demand in each line, subject to a variety of rules and regulations that must be enforced. Once a minimum number of shifts is calculated, the problem changes in order to determine the estimated cost of each of the potential solutions.

Finally, the satisfaction of the drivers must be maximized by means of an equitable assignment of working hours, and considering their preferences (normally they are considered as soft constraints).

So, trying to satisfy the maximum required demand, the mathematic model [15] considered in this study is presented as follows:

$$Min \sum_{j=1}^{m} c_j x_j \tag{1}$$

where

$$\sum_{j=1}^{m} a_{j,i} x_j \geq r_i \qquad x_j \geq 0 \,;\; x_j \in \mathrm{Z} \qquad (i=1 \ldots h)$$

h = Number of periods of time considered (normally hours)
m = Number of shifts allowed or possible shifts

$$\mathrm{a}_{\mathrm{j,i}} = \begin{cases} 1 & \text{if the period i is included in the shift j} \\ 0 & \text{otherwise} \end{cases} \qquad (i = 1 \ldots\ h) \quad (j = 1 \ldots m)$$

c_j = Cost of having a driver working during the shift j ; *(j = 1 ... m)*

r_i = Level needed of drivers in the period *i ;* *(i = 1 ... h)*
x_j = Number of drivers being employed at the shift *j ;* *(j* = 1 ... *m)*

To obtain the total cost of the solution considered, it is necessary to add the penalties for the soft restrictions not covered to the value obtained by this objective function.

In computational complexity theory, this type of problem belongs to the class of NP-complete problems [16]. Thus, it is assumed that there is no efficient algorithm for solving it and, in the worst case, the running time for an algorithm for this problem depends exponentially on the number of drivers, categories, shifts and so on, so that even some instances with only dozens of drivers cannot be solved exactly.

4 Proposed Method

The idea of this research is to construct an initial solution in a guided way, considering all the input information available and all the hard and soft restrictions previously defined. Once this initial process is finished, the solution obtained is evaluated to test if it is a feasible solution to try to improve it on a subsequent phase.

Three different sources of input information of are considered: the Labour Agreement, the Personnel Costs, and the Demand of Work to be satisfied; being the last one the most important. Once these three sets of data are defined, the satisfaction of the drivers must be maximized by means of an equitable assignment of working hours, and considering their preferences (normally they are considered as soft constraints). The two phases of construction of the calendar are initially applied, and the yielded solution is hopefully improved through a Scatter Search Algorithm. This last one is selected because the Scatter Search framework is one of the most flexible algorithms, allowing the development of alternative implementations with varying degrees of sophistication.

According to this, Fig. 1 shows the process of obtaining the final solution.

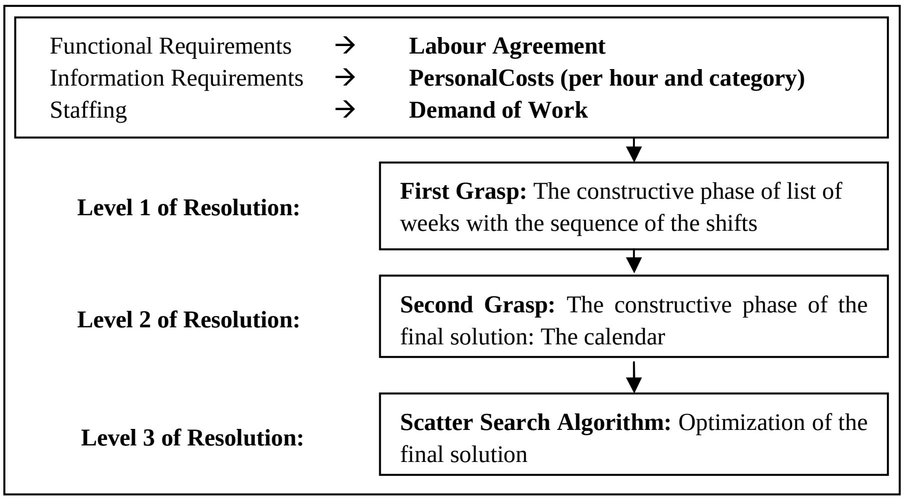
Functional Requirements → Labour Agreement
Information Requirements → PersonalCosts (per hour and category)
Staffing → Demand of Work
Level 1 of Resolution:
First Grasp: The constructive phase of list of weeks with the sequence of the shifts
Level 2 of Resolution:
Second Grasp: The constructive phase of the final solution: The calendar
Level 3 of Resolution:
Scatter Search Algorithm: Optimization of the final solution

Fig. 1. The Process of obtaining the final Solution with two Constructive Algorithms and a Scatter Search Algorithm

The construction of the solution consists of three different phases:

An initial constructive phase obtaining a list of weeks satisfying the conditions related to the sequence of shifts and the hard constraints.

As a first step, it is necessary to generate the list of all the possible weeks that can be obtained for a single driver. This is obtained by using a standard Grasp (greedy randomized adaptive search procedure) algorithm.

As the output is composed of a high number of elements, a subsequent step must be carried out to reduce the size of its elements, using a first group of restrictions. This includes all the hard restrictions that are selected by the user at the beginning of the execution.

The next step is to examine which weeks fulfil the sequence of shifts conditions to be joined with other previously given, without violating some of the previous restrictions. This will save computation time in a later phase.

The size chosen for the reduced list is the number of drivers working for the company studied.

A subsequent constructive phase (from the results obtained in the previous phase), considering the soft constraints and obtaining the final solution: a valid calendar.

This stage is also composed by two different parts, again implemented by means of a Grasp algorithm. The first one is the constructive phase of the calendar. In this phase, an initial group of *n* weeks is extracted randomly from the results of the previous main step. In the second stage, a post-processing phase is performed over the result obtained in this way in the previous part, examining if the list chosen fulfils a second block of restrictions composed of the soft restrictions determined by the user. In case that these restrictions are satisfied, such list is stored. Otherwise, one of the weeks that form this solution is rejected, substituting it by other one obtained from the post-processing phase. Again, it is analysed to check if it fulfils the requirements. This way, possible combinations are generated and rejected in an iterative way.

Finally, a study of improvement of the obtained solution is carried out, looking for some other combinations in the environment of the solution that can improve the solution considered.

Finally, the optimization of the preliminary solution is constructed using a Scatter Search Algorithm.

Scatter Search is a population and evolutionary method. In contrast to other evolutionary methods like genetic algorithms, scatter search is mostly based on systematic designs and methods with the purpose of creating new solutions. It uses strategies for search diversification and intensification that have proved effective in a variety of optimization problems.

This study makes use of this algorithm, because it has demonstrated to solve efficiently this type of complex problem, even for instances of the problem of great size. It operates with a set of reference of the population. In SS it is possible to generate a very significant number of combinations with few individuals. SS systematicly introduces diversity in the set reference.

To apply a Scatter Algorithm [14, 17, 18] to this Scheduling problem, first the problem must be represented as a genome. To make sure that a certain genome is a feasible solution, it must be checked if it obeys the precedence constraints previously indicated. The fitness function of a solution considers the number of penalties and the value of each one of the restrictions previously considered, as well as the number of restrictions not met by this solution.

The algorithm proceeds to initialize a population of solutions considering the solution generated by the second constructive phase as an entry. All the solutions in the population are generated from that initial solution, and the evolution happens in generations.

In each generation, the fitness of every individual in the population is evaluated, multiple individuals are selected in a guided way from the current population (based on their fitness), and modified (recombined and possibly mutated) to form a new population. The new population is then used in the next iteration of the algorithm. Thus, each generation consists on updating the population of individuals, using for it a new population obtained by recombination of the characteristics of some selected individuals.

The algorithm ends when, either a maximum number of generations have been produced, or a satisfactory fitness level has been reached for the population. Among the different final solutions obtained, the solution that is closer to the ideal value or that is in the environment of the ideal solution will be selected.

5 Analysis of Results

This section includes a description of a very simple generic problem, along with the solution found. The calendar generated in the application for every driver is a succession of days, and the temporary horizon can be a week, a month, a semester or a year.

The constructive phases have been used to identify the lists of feasible weeks, yielding a list of shifts for each driver. Afterwards, the final annual calendar has been constructed from these feasible lists of shifts in the 4 weeks.

The following is only an explanatory example, considering only 4 drivers. For every day of the week, working place and shift, a cost is defined by hour and a cost of overtime. The restrictions considered in this example are showed in Table 1.

Hard Restrictions	1.	The shifts of work to be considered: 2 (M, A)
	2.	The hourly ranges indicated in the shifts. The hourly ranges cannot be overlapped between shifts: 8 hours per shift
	3.	Types of days of work considered (TC: complete time, TP: partial time…): Only TC.
	4.	The number of drivers: 4
	5.	Different categories in which the drivers are grouped: Only one category.
	6.	Different groups of days defined: Labour days, weekly days of rest and days of annual vacations
	7.	Maximum number of consecutive days: 6
Soft Restrictions	1.	Maximum number of days worked in a year: 247.
	2.	Holidays and days off in the calendar of work: 1st January, 25th December (complete day, all the shifts), 24th and 31st December (only night shifts).
	3.	Preferences of the drivers to not to be employed at a certain shift: Driver 4 prefers not to work in Afternoon Shift.
	4.	Restrictions about labour days, shift and days of rest: All drivers prefer not to work on Sunday
	5.	Maximum number of a shift type per week: 1
	6.	Maximum number of assignments per day of the week: 6
	7.	Maximum number of assignments for each shift type: 6
	8.	Maximum number of consecutive working weekends: 2

Table 1. Lists of hard and soft restrictions to be considered

Considering a section composed by only 4 drivers, the cycle of this section or department will be constituted by 4 weeks.

In the results showed next; the temporary horizon has been considered as a year, only two shifts have been considered (M = Morning, A = Afternoon, B = Break), and the schedule includes only 4 drivers. To obtain the annual solution, is necessary to extend the 4 solutions obtained until this moment to the 52 weeks of the year. The final solution satisfies all of the hard restrictions and most of soft restrictions, with the exception the one marked with 3 on Table 1.

The list of shifts to complete in 4 weeks for each of the 4 drivers considered is shown in Table 2.

	WEEK 1							WEEK 2						
Driver 1	M	M	M	M	M	B	B	A	A	A	A	A	A	B
Driver 2	A	A	A	A	A	A	B	B	M	M	M	M	M	B
Driver 3	B	M	M	M	M	M	B	A	A	A	A	A	B	B
Driver 4	A	A	A	A	A	B	B	M	M	M	M	M	B	B
	WEEK 3							WEEK 4						
Driver 1	B	M	M	M	M	M	B	A	A	A	A	A	B	B
Driver 2	A	A	A	A	A	B	B	M	M	M	M	M	B	B
Driver 3	M	M	M	M	M	B	B	A	A	A	A	A	A	B
Driver 4	A	A	A	A	A	A	B	B	M	M	M	M	M	B

Table 2. Lists of shifts to be worked by 4 drivers in 4 weeks

The solution obtained satisfies all the hard constraints and 89% of the soft constraints previously considered to generate the calendar of work; such as labour days, shift and days of rest, and some restrictions about intervals of the vacations and about days of local or national holiday.

The final solution considers the extension of these lists through the 52 weeks in the year. As Table 3 shows, the final results include a very similar number of worked days and shifts in a year for each driver, applying this pattern of solution.

Drivers\Shifts	Morning	Afternoon	Total
Driver 1	130	143	273
Driver 2	129	144	273
Driver 3	128	145	273
Driver 4	133	140	273

Table 3. Final results of the number of worked days in a year

As it can be observed in Table 3, the differences between the drivers are not relevant, and taking into account that the rest of restrictions were satisfied, it can be concluded that the application of the constructive method and optimization algorithm previously described are considered as satisfactory to solve this problem. Both the requirements of the company (minimization of the costs to cover all routes) and those of the workers (almost no differences in working hours and most of other constraints covered) have been observed.

For this solution the execution time is 4915 ms, the best solution has a cost of 138350 and the worst solution has a cost of 273725.

As additional remark, it has been observed that if the number of hard restrictions increases, the execution time reduces and the obtained solutions deteriorate significantly. This is due to the fact that there are few possibilities of finding solutions that satisfy many hard restrictions (normally if there are a lot of hard restrictions they have opposite interests). This makes the execution to stop sooner, as no improvements are made during a determined number of iterations. On the contrary, if the number of hard restrictions decreases, the time of execution increases and the obtained solutions are better.

Based on results obtained with greater instances, with this way of solving this problem of driver scheduling when the size of the problem increases, advantages of using this method become clearer, as it obtains optimal results without the computational complexity of other methods, which have to manage invalid solutions.

6 Conclusions

In this work, the Bus Driver Scheduling Problem, which is an important aspect of the Transportation Planning System, has been analysed. The presented model for obtaining the final calendar has been designed trying to achieve simplicity, solution quality and applicability, as its main characteristics.

Its main characteristic is the partition of the main problem of calendar construction in three different stages. That way, first an initial solution is constructed and then the solution is improved to obtain the best solution possible. This guided approach avoids the pitfalls of other, more random, solutions; in which more computing time to get to a correct solution would be higher. It also favours the construction of solutions that comply with the majority of the restrictions that the problems require. As results prove, this method ensures that all hard restrictions are satisfied, while it helps to fulfil also with the majority of the soft ones.

Future lines of work include the consideration of extending its use to another Scheduling Problems, the consideration of others constraints, and the consideration of uncertain data using the Fuzzy Sets Theory to solve the Scheduling Problem.

ACKNOWLEDGMENTS
This research has been partially supported through the projects of the Spanish Ministry of Science and Innovation CIT-020000-2009-12 and TIN2010-21272-C02-01 (funded by the European Regional Development Fund).

7 References

1. Brusco MJ and Jacobs LW.: Personnel tour scheduling when starting time restrictions are present. Management Science 44: 534–547. (1998).
2. Rezanov N, Ryan D.: The train driver recovery problem. A set partitioning based model and solution method. Computers & Operations Research. Volume 37, issue 5, pages 845-856. (2010).
3. Tang J, Zhang J, Pan Z.: A Scatter Search Algorithm for solving vehicle routing problem with loading cost. Expert Systems with Applications. Volume 37, issue 6, pages 4073-4083. (2010).
4. Portugal R, Ramalhingo-Lourenco H, Paixao J.P.: Driver Scheduling Problem Modelling. Public Transp (2009) 1: 103–120 DOI 10.1007/s12469-008-0007-0. SpringerLink. (2006).

5. Ramalhingo-Lourenco H.: The Crew-Scheduling Module in the GIST System. UPF Economics Working Paper No. 547. (2001).
6. Carraresi P, Gallo G, Rousseau J.M. : Relaxation approaches to large scale bus driver scheduling problems. Transportation Research Part B: Methodological Volume 16, Issue 5, Pages 383-397. (1982).
7. Mesquita M, Paias A.: Set Partitioning/covering-based approaches for the integrated vehicle and crew scheduling problem. Computers & Operations Research. Volume 35, issue 5, pages 1562-1575. (2008).
8. Abbink, E.J.W.: Solving large scale crew scheduling problems by using iterative partitioning. 7th Workshop on Algorithmic Methods and Models for Optimization of Railways (ATMOS'07). (2007).
9. Laurent B, Hao Jin-Kao.: Simultaneous Vehicle and Crew Scheduling for Extra Urban Transports. Computers & Industrial Engineering. Volume 53, issue 3, pages 542-558. (2007).
10. Ramalhingo-Lourenco H & Portugal R.: Metaheuristics for The Bus-Driver Scheduling Problem. Economic Working Papers Series, no. 304, Universitat Pompeu Fabra. (1998).
11. Abbink, E.J.W et at el.: Solving Large Scale Crew Scheduling Problems in Practice. Research Paper. Report/Econometric Institute, Erasmus University Rotterdam, pp 1-19. Erasmus School of Economics (ESE). This publication is part of collection Econometric Institute Research Papers Published by Econometric Institute. (2010).
12. Kwan R, Kwan A.: ASK Effective search space control for large and/or complex driver scheduling problems. Annals of Operations Research. Volume 155, issue 1, pages 417-435. (2007).
13. Tavakkoli-Moghaddam R, Makui A, Mazloomi Z.: A new integrated mathematical model for a bi-objective scatter search algorithm. Journal of Manufacturing Systems. Volume 29, issue 2-3, pages 111-119. (2010).
14. De Leone R, Festa P and Marchitto E.: The Bus Driver Scheduling Problem: a new mathematical model and a GRASP approximate solution. Tech. Rep. 22, University of Napoli, Federico II. (2006).
15. Danzing G.B.: A Comment on Edie´s `Traffic Delays at Toll Booths´. Operations Research, 2 3, pp. 339-341. (1954).
16. Garey M. R. and Johnson D. S.: Computers and Intractability: A Guide to the Theory of NP-Completeness Freeman. (1979).
17. Feo T and Resende M.: Greedy randomized adaptive search procedures. Volume 6:109–133. Journal of Global Optimization. (1995).
18. Glover f, Laguna M, Martí R.: Scatter Search. In Advances in Evolutionary Computation: Theory and Applications, pages 519-537, Springer-Verlag. (2003).