PLANNING BUS DRIVER ROSTERS

Marta Mesquita[1], Margarida Moz[2], Ana Paias[3] and Margarida Pato[2]
[1] CIO and ISA-UTL, Tapada da Ajuda, 1349-017 Lisboa, Portugal
[2] CIO and ISEG-UTL, Rua do Quelhas 6, 1200-781 Lisboa, Portugal
[3] CIO and DEIO-FCUL, Bloco C6, Piso 4, 1749-016 Lisboa, Portugal

Keywords: Bus driver rostering, Mixed integer linear programming, Branch-and-bound techniques.

Abstract: This paper proposes a methodology for planning bus driver rosters with days off patterns in public transit companies. The problem is modeled as a mixed integer linear programming problem which is solved with special devised branch-and-bound techniques by a standard MILP solver. The new methodology was tested on instances of two companies operating in Portugal. Two types of days off rules giving rise to rosters with specific days off patterns are compared. The computational experiment shows promising results which suggest that the proposed framework can be used as a tool to evaluate and discuss different days off patterns within public transit companies.

1 INTRODUCTION

In urban public transit companies, bus driver rostering is the problem of assigning drivers to vehicle schedules, while satisfying labor law, contracts and internal regulations. A vehicle schedule is the sequence of timetabled trips to be performed by a vehicle during a day. Scheduling for both the vehicles and the drivers must be defined for a given time horizon. The solution quality of these problems has a great impact on transit companies' operating costs. For example, according to the "Annual report & accounts 2009" of CARRIS SA, the main public transit company in Lisbon, expenditure on staff is 62.5% of operating costs which, in turn, is 71% of current costs. Hence, this is an area where savings are urgent.

Due to the computational complexity of these problems, they are usually solved separately, on a sequential basis. First, a vehicle scheduling problem is solved for each day of the time horizon building the daily schedules for the vehicles that cover the demand for urban transport. Then, also for each day, crew duties are defined to cover vehicle schedules, satisfying daily labor constraints – the crew scheduling problem. Afterwards, using this information, a rostering problem is solved to assign the anonymous daily crew duties to specific company drivers, thus defining their sequences of work days and days off for the whole time horizon.

Different mathematical formulations and solution approaches have been proposed for the rostering problem in several transport contexts. An extensive survey may be found in (Ernst, Jiang, Krishnamoorthy, Nott and Sier 2004). Multilayer network models have been proposed in (Carraresi and Gallo, 2004) and (Moz, Respício and Pato, 2009) for bus driver rostering, (Aringhieri and Cordone, 2004) for refuse collection staff and (Cappanera and Gallo, 2004) for air crews. Set covering/partitioning models have been considered by Catanas and Paixão (1995) for bus driver rostering and Freling, Lentink and Wagelmans (2004) for railway and air crews. Recently, Hartog, Huisman, Abbink, and Kroon (2009) presented an assignment model with additional constraints and developed a decision support system for crew rostering at NS (Netherlands railways) and Nurmi, Kyngäs, and Post (2011) proposed a population-based local search algorithm to schedule drivers in a Finnish bus transit company.

This paper proposes a mathematical formulation and a computational framework to solve rostering problems with days off patterns in public transit companies. The paper is organized as follows: in the next section we give some definitions and notation; in section 3, we present a mathematical formulation for the problem along with a brief description of the solution approach; computational results concerning two different roster patterns are reported and

Mesquita M., Moz M., Paias A. and Pato M..
PLANNING BUS DRIVER ROSTERS.
DOI: 10.5220/0003757504150420
In *Proceedings of the 1st International Conference on Operations Research and Enterprise Systems* (ICORES-2012), pages 415-420
ISBN: 978-989-8425-97-3

discussed in section 4; finally, in section 5, some conclusions are drawn.

2 DEFINITIONS AND NOTATION

The bus driver rostering problem, DRP, consists of assigning a set of M drivers to daily crew duties that operate the vehicles during a given planning horizon H. In this paper, we consider a planning horizon of 7 weeks, 49 days. A crew duty is a daily working period that respects labor law, union contracts and internal rules of the company such as maximum/minimum spread (time elapsed between the beginning and end of a crew duty), maximum working time without a break, break duration.

The sequence of crew duties and days off, one per day, assigned to a particular driver during the planning horizon is called a line of work. The set of lines of work, covering all crew duties, assigned to the drivers of the company is the roster. A roster must satisfy a set of constraints related with labor union contracts as well as internal rules of the company. These constraints concern the minimum number of days off per week, specific days off per week, minimum number of Sundays off in the planning horizon, minimum number of consecutive days off, maximum number of consecutive workdays and minimum number of rest hours between consecutive crew duties.

Different policies may be followed in a company, or in different companies, to build the roster. Some groups of drivers are scheduled in a cyclic basis so that all drivers in a group are assigned to the same type of work and rest periods. In order to be able to perform all crew duties within cyclic rostering, drivers in the same group usually share the same characteristics, namely seniority, same bus and route knowledge. In this paper we deal with a group of drivers whose contracts allow more flexibility on the rosters. These drivers work according to a pre-defined days off pattern where they get the same type of rest periods but not necessarily the same type of crew duties.

Each days off pattern is *a priori* defined and includes one, or more than one, days off schedules. Each days off schedule is a template for a line of work that fixes the days off and the working days to be filled with crew duties. Each days off schedule satisfies, *a priori*, a subset of the above mentioned constraints: the minimum number of days off per week is 1; the minimum number of Sundays off in the planning horizon is 2; the minimum number of consecutive days off is 2; the maximum number of consecutive workdays is 6. The remaining constraints will be explicitly considered in the mathematical model presented in the next section.

The 0-1 matrix in Table 1 gives an example of a days off pattern where each day off is denoted by 0 and each workday by 1. This pattern covers a planning horizon of 7 weeks and includes 7 days off schedules. Schematically, each days off schedule starts in row 1 of any column, a Monday, and consists of 7 consecutive columns being the last day (Sunday) of column i followed by the first day (Monday) of column i+1. Note that, column 1 follows column 7. For example, a driver assigned to the days off schedule that starts with column 6 works and rests according to columns 6, 7, 1, 2, 3, 4 and 5 during weeks 1, 2, 3, 4, 5, 6 and 7, respectively.

Table 1: Example of days off pattern.

	1	2	3	4	5	6	7
Mon	0	1	1	1	1	1	0
Tue	0	0	1	1	1	1	1
Wed	1	0	0	1	1	1	1
Thu	1	1	0	0	1	1	1
Fri	1	1	1	0	0	1	1
Sat	1	1	1	1	0	0	1
Sun	1	1	1	1	0	0	1

The days off pattern described in Table 1 is followed by a group of drivers from a public transit company in the city of Lisbon. According to it, all drivers have 4 consecutive rest periods of 2 days off and 2 consecutive rest periods of 3 days off which include Saturday and Sunday. Moreover, during the planning horizon all drivers rest two Mondays, two Tuesdays,.., and two Sundays. Consequently, all drivers share the same type of rest periods and days off, and a roster built with these schedules has a cyclic nature in what concerns the days off.

In the next sections we present a methodology to solve the rostering bus driver problem with pre-defined days off pattern. As an additional tool, the underlying computational framework may be used to compare rosters built under different days off patterns regarding the rostering problem objectives: minimizing the number of drivers assigned to work and evenly distribute the workload among the drivers during the planning horizon.

3 MATHEMATICAL MODEL

Each daily crew duty has to be assigned to a driver that works according to one of the schedules included in the days off pattern in use. Let S be the set of days off schedules and let L^h be the set of crew duties to be performed on day $h \in H$. According to the crew duty starting time, L^h is partitioned into L_E^h, set of early crew duties starting before 3:30 p.m. and L_A^h, set of late crew duties starting after 3:30 p.m. According to the crew duty spread L^h is partitioned into L_T^h, set of short duties with a maximum spread of 5 hours (without lunch); L_N^h, set of normal duties with spread $\in [5,9]$ hours and L_O^h, set of long duties with spread $\in]9, 10.75]$ hours (with overtime).

The DRP can be formulated as an assignment/covering problem with additional constraints, as stated in (Mesquita, Moz, Paias and Pato, 2011) for the integrated vehicle-crew-roster problem.

The mathematical model includes three types of decision variables. Let $y_\ell^{mh} = 1$, if driver *m* performs crew duty ℓ on day *h*, or 0 otherwise. Let $x_s^m = 1$, if driver *m* is assigned to schedule *s*, or 0 otherwise. A cost r^m, related with driver *m* salary, is associated with variables x_s^m. The objective function is devised to minimize the number of drivers assigned to work as well as to evenly distribute the workload among the drivers. That is, the undesirable types of daily crew duties - short and long - must be equitably partitioned among the lines of work assigned to drivers. To balance the workload we define a third type of decision variables η_T and η_O, which represent, respectively, the maximum number of short and long crew duties assigned to a driver during *H*. Penalties λ_T and λ_O are associated with η_T and η_O, respectively.

The DRP can be stated as the following MILP:

$$\text{Min} \sum_{m \in M} \sum_{s \in S} r^m x_s^m + \lambda_T \eta_T + \lambda_O \eta_O \qquad (1)$$

$$\sum_{m \in M} y_\ell^{mh} = 1, \quad \ell \in L^h,\ h \in H \qquad (2)$$

$$\sum_{s \in S} x_s^m \le 1, \quad m \in M \qquad (3)$$

$$\sum_{\ell \in L^h} y_\ell^{mh} - \sum_{s \in S} a_s^h x_s^m \le 0, \quad m \in M, h \in H \qquad (4)$$

$$\sum_{\ell \in L_A^h} y_\ell^{mh} + \sum_{\ell \in L_E^{h-1}} y_\ell^{m(h-1)} \le 1, m \in M, h \in H - \{1\} \qquad (5)$$

$$\sum_{\ell \in L_E^h} y_\ell^{mh} + \sum_{\ell \in L_A^{h-1}} y_\ell^{m(h-1)} \le 1, m \in M, h \in H - \{1\} \qquad (6)$$

$$\sum_{h \in H} \sum_{\ell \in L_t^h} y_\ell^{mh} - \eta_t \le 0, \quad m \in M, t \in \{T, O\} \qquad (7)$$

$$y_\ell^{mh} \in \{0,1\}, \quad \ell \in L^h, m \in M, h \in H \qquad (8)$$

$$x_s^m \in \{0,1\}, \quad s \in S, m \in M \qquad (9)$$

$$\eta_T, \eta_O \ge 0 \qquad (10)$$

Constraints (2) guarantee that each crew duty is assigned to one and only one driver. Constraints (3) ensure that a driver is assigned to one schedule of set *S* or is available for other services in the company. Inequalities (4), where parameter $a_s^h = 1$ if *h* is a workday on schedule *s* or $a_s^h = 0$ if *h* is a day off on schedule *s*, link variables y_ℓ^{mh} and x_s^m. These inequalities establish that a driver assigned to a duty ℓ on day *h* works according to a schedule *s* where *h* is a workday. Constraints (5) prevent undesirable sequences of crew duties in which, on consecutive days, a driver performs an early duty followed by a late duty. Constraints (6) ensure the implementation of labor laws with regard to minimum rest periods between consecutive working days. That is, (6) forbids the assignment of a driver to an early duty in the day after he performed a late duty. Both (5) and (6) impose a day off period between different crew duty types. Inequalities (7) define the variables η_T and η_O which determine the maximum number of short/long duties assigned to a driver.

Branch-and-bound techniques are used to obtain optimal/near optimal solutions for DRP. Due to the combinatorial nature of the problem only small real instances can be solved directly with a software package. To reduce the size of the instances under resolution different branching strategies combined with variable fixing have been tested and compared.

The mathematical model (1) to (10) includes two sets of integer decision variables. Variables *x* define the assignment of drivers to the days off schedules thus establishing, for each driver, the sequence of rest periods. Variables *y* define the assignment of drivers to the crew duties thus determining the sequence of crew duties that each driver has to perform along the planning horizon. These two sets of integer variables suggest different branching strategies according to the subset of variables branching dichotomy is based on. The first branching rule (R1) looks to the linear programming relaxation solution and fixes to 1 decision variables $x_s^m > 0.75, \quad \forall s \in S, m \in M$ and decision variables $y_\ell^{mh} > 0.999, \quad \forall \ell \in L^h,\ h \in H, m \in M$. The second rule (R2) performs branching over the subset of variables y_ℓ^{mh}, $\forall \ell \in L^h$, $h \in H, m \in M$, fixing to 1 variables $y_\ell^{mh} > 0.85$ at the end of the root node. The third rule (R3) performs branching over the two subsets of variables y_ℓ^{mh}, $\forall \ell \in L^h$, $h \in H, m \in M$ and $\sum_{s \in S} x_s^m$, $\forall m \in M$, fixing to 1 variables $y_\ell^{mh} > 0.95$ at the end of the root node.

Computational results are shown in the next section.

4 COMPUTATIONAL RESULTS

We have compared two days off patterns denoted by PI and PII. Pattern PI was described in section 2, Table 1. Pattern PII, described in Table 2, includes two different sets of days off schedules. One is the set of days off schedules defined by PI. The other set includes a single schedule, denoted by s_8, containing 7 rest periods, which always occur on Saturday and Sunday. That is, during the 7 weeks of the planning horizon, a driver assigned to s_8 will have his rest periods always on Saturday and Sunday. PII arises to counterbalance the lower demand, in what concerns the number of crew duties to cover during weekends. Within pattern PII, the group of drivers is partitioned into two sub-groups: one sub-group will work according to the days off schedules, $s_1,\ldots,s_7$, defined by PI and the other sub-group according to s_8. Pattern PII is not a cyclic pattern since drivers assigned to different subgroups no longer have the same type of rest periods.

Table 2: Days off pattern PII.

	1	2	3	4	5	6	7	s_8
Mon	0	1	1	1	1	1	0	1
Tue	0	0	1	1	1	1	1	1
Wed	1	0	0	1	1	1	1	1
Thu	1	1	0	0	1	1	1	1
Fri	1	1	1	0	0	1	1	1
Sat	1	1	1	1	0	0	1	0
Sun	1	1	1	1	0	0	1	0

The algorithms were coded in C++ and the programs ran on a PC Pentium IV 3.2 GHz. Branch-and-bound schemes were tackled with CPLEX 11.0.

We have considered 8 instances: 5 instances, denoted by L1,…,L5, derived from a bus company operating in the city of Lisbon and 3 instances, denoted by P1, P2, P3, derived from a bus company operating in the city of OPorto. Characteristics of the test instances are described in Table 3. Columns 2 and 3 show the number of short/normal/long daily crew duties to be covered, respectively, from Monday to Friday and on weekend days. Column 4 refers to the total number of (short/normal/long) crew duties to be assigned during the planning horizon.

Table 3: Data set description.

	Daily crew duties		Total crew duties
	Mon-Fri	Sat-Sun	Mon-Sun
	T / N / O	T / N / O	T / N / O
L1	1 / 8 / 8	4 / 3 / 2	91 / 322 / 308
L2	4 / 14 / 20	7 / 6 / 6	238 / 574 / 784
L3	0 / 9 / 30	1 / 4 / 11	14 / 371 / 1204
L4	0 / 8 / 26	0 / 6 / 9	0 / 364 / 1036
L5	5 / 13 / 36	11 / 8 / 8	329 / 567 / 1372
P1	3 / 9 / 16	6 / 6 / 4	189 / 399 / 616
P2	3 / 28 / 12	10 / 8 / 8	245 / 1092 / 532
P3	0 / 9 / 30	7 / 4 / 12	98 / 371 / 1218

Parameters λ_T and λ_O in (1) were set to $\lambda_T = 0.5$ and $\lambda_O = 1$, as long crew duties are more undesirable than short crew duties since contain overtime.

A time limit of 10800 seconds has been imposed as stopping criterion for solving MILP problems.

Computational experiments on the different branching strategies have shown that rule (R3) yielded the best branch-and-bound results concerning either CPU time and solution quality. On the one hand, (R1) proved to be ineffective for solving medium/large size instances due to excessive CPU times. On the other hand, both (R2) and (R3) led to feasible solutions within the time limit but (R3) gave a better solution in 11 out of 16 instances. As for the branch-and-bound CPU time, Table 4 shows, for each instance, the corresponding time, in seconds, for both (R2)/(R3) and PI/PII, excluding root node CPU. The last two rows present, respectively, the average time and the number of times branch-and-bound stopped due to the time limit.

Results reported in Table 4 strengthen the choice of (R3) to compare the quality of PI and PII solutions.

Table 4: Comparing (R2) and (R3) – CPU time.

	PI		PII	
	(R2)	(R3)	(R2)	(R3)
L1	30.4	46.9	10800	10800
L2	15.3	70.8	10800	528.6
L3	0.5	15.0	10800	571.6
L4	0.1	3.1	10800	82.5
L5	107.4	786.9	10800	10800
P1	343.1	254.0	10800	1095.3
P2	8225.9	10800	10800	10800
P3	2352.2	355.1	10800	575.0
av	1384.4	1541.5	10800	4406.6
#tl	0	1	8	3

Table 5 presents computational results for PI and PII. Column 3 shows the number of drivers assigned to the best roster attained, while in brackets the number of drivers assigned to s_8 is reported. Column 4 presents the maximum number of short/long crew duties assigned to a driver during the planning horizon. The last column displays total CPU times, in seconds, spent by the enhanced branch-and-bound algorithm (LP relaxation CPU + MILP CPU).

Table 5: Computational results for PI vs PII.

		# driv	#T / #O	CPU (sec)
L1	PI	25	4 / 14	106
	PII	24 (8)	4 / 15	10805
L2	PI	54	6 / 16	85
	PII	47 (18)	9 / 20	575
L3	PI	55	1 / 22	22
	PII	47 (22)	1 / 26	599
L4	PI	49	0 / 22	18
	PII	40 (17)	0 / 35	93
L5	PI	77	5 / 18	828
	PII	68 (26)	8 / 23	10924
P1	PI	40	14 / 17	288
	PII	36 (12)	7 / 18	1112
P2	PI	61	5 / 9	10838
	PII	57 (17)	13 / 13	10853
P3	PI	59	4 / 25	394
	PII	50 (15)	4 / 28	628

From Table 5, we can see that with PII one can save on the number of drivers assigned to work. In fact, it is important to know the minimum workforce needed to operate the fleet of vehicles in order to have a pool of drivers available to replace those absent or to be assigned to other services in the company. Column 4 gives an idea of the roster quality concerning the worst scenario of short/long duties assigned to a driver in the planning horizon. Columns 3 and 4 show that the reduction on the number of drivers with PII comes with an increase in the maximum number of long crew duties assigned to a driver. This was expectable since the number of long crew duties is the same in PI and PII but in PII is divided by fewer drivers.

Although there is no guarantee of obtaining an optimal solution to DRP, one can see that the number of long crew duties is fairly distributed among de drivers. For example, concerning instance P1, a total of 616 long duties (table 3) must be assigned to 40 drivers under pattern PI and to 36 drivers under PII (table 5) which yields on average 15.4 and 17.1 for PI and PII, respectively. The solution provided by our methodology leads to a maximum of 17 and 18 long duties assigned to a driver, respectively, for PI and PII.

The last column shows that CPU times spent to solve the tested instances are quite reasonable.

5 CONCLUSIONS

Expenditures on staff have a great impact on operating costs of public transit companies. One of the main objectives of the rostering problem is to minimize such costs. However, it is important that the rosters comply with driver preferences. Some preferences concerning rest periods can be drawn *a priori* through the days off pattern. This paper proposes a computational framework that, given a pre-defined days off pattern, builds the bus driver roster for a planning horizon. The methodology has been tested over two sets of real instances derived from bus companies and proved to be an effective tool for building and comparing rosters under different days off patterns.

ACKNOWLEDGEMENTS

This work is supported by National Funding from FCT - Fundação para a Ciência e a Tecnologia, under the project: PEst-OE/MAT/UI0152.
The authors thank to CARRIS, SA for the helpful discussions.

REFERENCES

Aringhieri, R., Cordone, R., 2004. The multicommodity multilevel bottleneck assignment problem. *Electronic Notes in Discrete Mathematics*, 17, pp 35-40.

Catanas, F., Paixão, J. M. P., 1995. A new approach for the crew rostering problem. In Daduna, J., Branco, I. M., Paixão J. M. P. (eds.), *Computer-Aided Scheduling of Public Transport*. Lecture Notes in Economics and Mathematical Systems, 430. Springer, -Verlag, Berlin, pp 267-277.

CPLEX Manual (version 11.0), 2007. Using the CPLEX[R] Callable Library and CPLEX Mixed Integer Library. ILOG INC., Incline Village, Nevada, USA.

Cappanera, P., Gallo, G., 2004. A multicommodity flow approach to the crew rostering problem. *Operations Research,* 52(4), pp 583-596.

Carraresi, P., Gallo, G., 1984. A multi-level bottleneck assignment approach to the bus drivers rostering problem. *European Journal of Operational Research,* 16, pp 163-173.

Ernst, A., Jiang, H., Krishnamoorthy, M., Nott, H., Sier, D., 2001. An Integrated Optimization Model for Train Crew Management. *Annals of Operations Research,* 108, pp 211-224

Freling, R., Lentink, R., Wagelmans, A., 2004. A decision support system for crew planning in passenger transportation using a flexible branch-and-price algorithm. *Annals of Operations Research,* 127(1-4), pp 203-222.

Hartog, A., Huisman, D., Abbink, E. J. W., Kroon, L. G., 2006. Decision support for crew rostering at NS. *Public Transport* 1(2), pp 121-133.

Mesquita, M., Moz, M., Paias, A., Pato, M., 2011. An integrated vehicle-crew-roster problem with days off pattern. In Hu, B., Morasch, K., Pickl, S., Siegle, M., (eds.), *Operations Research Proceedings 2010,* Selected papers of the Annual International Conference of the German Operations Research Society. Springer-Verlag, Berlin, pp 245-250.

Moz, M., Respício, A., Pato, M. V., 2009. Bi-objective evolutionary heuristics for bus driver rostering. *Public Transport*, 1(3), pp 189-210.

Nurmi, K., Kyngäs, J., Post, G., 2011. Driver Rostering for Bus Transit Companies. *Engineering Letters*, IAENG, Hong Kong, 19(2), pp 125-132,