J Sched (2007) 10: 245–254
DOI 10.1007/s10951-007-0024-x

Rostering in a rail passenger carrier

Mikel Lezaun · Gloria Pérez · Eduardo Sáinz de la Maza

Published online: 20 July 2007

Abstract In this paper we present an applied study, commissioned by the regional rail passenger carrier EuskoTren, into how the annual workload of drivers can be allocated in an egalitarian fashion. The allocation must meet the constraints arising from working conditions and the preferences of employees, as reflected in collective bargaining agreements. The workload varies over the five periods, into which the year is divided, and according to the day of the week. Moreover, not all morning, evening and night shifts are of equal duration. Reduced services on public holidays are also considered. The solution to the problem proposed is obtained in four linked steps, at each of which a binary programming problem is solved using commercial software. Step one is to build five lists of weekly multi-shift patterns, two of them rotating, that contain all the shifts in the week. Step two consists of the partially rotating annual assignment of patterns to drivers, step three involves the extraction of shifts by reduction of services on public holidays, and step four incorporates the durations in hours into the shifts already assigned. The final solution obtained is quite satisfactory: all drivers are assigned a similar number of morning, evening and night shifts and Sundays off, and they work practically the same number of days and hours per year. The results obtained, the adaptability of the system to new requirements and the computation time used are fully satisfactory to the firm, which has decided to implement the model.

Keywords Manpower planning · Crew rostering · Scheduling · Binary programming

M. Lezaun (✉) · G. Pérez · E. Sáinz de la Maza
Departamento de Matemática Aplicada, Estadística e Investigación Operativa, Facultad de Ciencia y Tecnología, Universidad del País Vasco, Apdo. 644, 48080 Bilbao, Spain
e-mail: mikel.lezaun@ehu.es

1 Introduction

The general rostering problem is how to assign shifts and rest days to employees, so that the predicted workload is met, taking into account the constraints deriving from the type of work and the preferences of workers. This is a long-standing problem and one that has many variants. In many industrial and public service firms work goes on for many hours per day every day of the year. In such cases work is typically split into morning, evening and night shifts, and the shifts are allocated week by week on a rotating basis. To that end, an ordered set of weekly patterns is designed that includes rest days, which are assigned to workers, consecutively. In other words, what is known as a "rotating schedule" must be drawn up. How work is organized varies greatly depending on the activity and philosophy of each firm. In particular, it depends on whether the workload varies from one day to another during the week, or on a seasonal basis, on whether the workforce is fixed or casual employees can be hired, e.g., to cover vacations, and on whether all employees are on a rotary schedule or some are assigned permanently to certain shifts or rest days, etc.

Shift work is always uncomfortable and a nuisance. In some organizations shift work is one of the basic causes of discontentment and complaints among workers. A well-designed system of alternating shifts and spacing out free weekends to ensure that work is distributed fairly (a hard problem to solve), therefore, often becomes a central issue in the negotiation of collective agreements. As Laporte (1999) indicates, this makes designing good schedules more of an art than a science.

A wide variety of rotating schedules exists. Esclapés (2000) provides an overview and an extensive bibliography on rostering problems. The paper by Ernst et al. (2004a) presents a review of staff scheduling and rostering. Burke et al. (2004) present the state of the art of nurse rostering. Laporte (1999) centres on rotating schedules. Ernst et al. (2004b) provide an annotated bibliography of personnel scheduling and rostering. However, although basic principles and some rules exist to govern the design of these schedules, in practice each situation has its own specific problems that must be solved. See, for instance, Azmat et al. (2004), Beaumont (1997), Caprara et al. (1997, 1998, 1999), Ernst et al. (2000), Isken (2004), Muslija et al. (2002), Tharmmaphornphilas and Norman (2004), and Townsend (1988). Laporte and Pesant (2004) give a classification of the main constraints governing the design of multi-shift rotating schedules and develop a new algorithm that can cover a wide variety of constraints. Their extensive bibliography lists articles featuring case studies that involve the different types of constraint.

In Lezaun et al. (2006), the authors study a similar, though simpler, problem, in which shifts are assigned without taking the number of hours into account.

In this article we present an applied study commissioned by EuskoTren with a view to improving the assignment of shifts and rest days to drivers and making it more egalitarian. Drivers have no hierarchical structure: they all do the same job in the same way. Their work is divided into morning, evening and night shifts of differing duration, and varies according to the days of the week and the five periods into which the year is divided. On public holidays services are reduced to the level provided on Sundays in the period, into which they fall. Weekends (Saturdays and Sundays) are not treated as different from the remaining days of the week. The workforce of drivers is sized to cover all the shifts, rest days and vacations of the year. Working conditions are agreed between the management of the firm and the representatives of the workers, and are covered in the constraints described below. These constraints are numerous and varied, and include the main families, described by Laporte and Pesant (2004). Work must, of course, be distributed equitably so that at the end of the year all drivers have been assigned approximately the same workload.

The solution to the problem proposed is obtained in four linked steps, at each of which a binary programming problem is solved that is built from the solution of the preceding step. First of all drivers' vacations need to be fixed. Vacations are assigned to each driver in blocks of consecutive weeks distributed throughout the year, so that in all cases several drivers will be assigned the same block of vacations.

The first step in the solution is to draw up five lists, one for each period of the year, with as many weekly multi-shift patterns as there are drivers not on vacation. Each list will contain all the shifts to be worked in a given week. The duration of shifts, in hours, is not specified at this first stage: only the type, i.e., morning, evening and night. The lists of patterns corresponding to periods with several consecutive weeks (winter and summer) must be ordered in such a way that a single driver can work them consecutively. Moreover, all the lists must take into account the requirements arising from the obligations and entitlements of the workers in regard to working conditions. The second step consists of the annual assignment of the weekly patterns to drivers, using the relevant list of patterns for each period. In winter and summer, weekly patterns are assigned to each driver week by week in increasing order on the list until the driver enters a vacation period. When a driver returns to work after vacations he/she is assigned one of the patterns that should be assigned to a driver beginning a vacation period, and then begins the consecutive assignment again. Since for every group of drivers, returning from vacation, there is another group starting vacation, there is some degree of freedom in assignments after vacations. Advantage can be taken of this to ensure that the annual workload obtained for each driver is similar. In the third step, the shifts corresponding to reductions of services on public holidays are eliminated from the previous solution, so that the similarity between workloads of different drivers becomes greater. Finally, the solution to the third step is used as the basis for assigning durations, in hours, to shifts. Not all shifts have the same duration, so this assignment takes into account certain homogeneity criteria, so that at the end of the year all drivers have worked a very similar number of hours.

The four resulting problems are solved using commercial software, e.g., the LINGO package by Lindo Systems Inc. (2003). The final solution obtained is quite satisfactory to all the parties involved, and EuskoTren has, therefore, decided to implement the procedure explained here and extend it to other groups who work shifts.

2 Problem description

EuskoTren drivers are grouped into different "residences". Shifts at each residence are allocated independently of the others. The allocation criteria applied must, of course, be the same at all the residences. To establish the idea, in this article we will deal only with the most complex residence, which comprises 51 drivers. All drivers have assigned nine weeks' vacation, divided into three blocks of three consecutive weeks. So there are nine drivers on vacation each week, except in the first week, when there are none. In the first week nine drivers are assigned to reserve or supplementary duties.

The time frame for planning is one year or, more specifically, 52 weeks, running from Monday to Sunday. The year is divided into five periods: winter, summer, San Sebastián week, Semana Grande ["Festival Week"] 1 and Semana Grande 2. The numbers of shift types in each period are shown in Tables 1 to 5. These tables only specify, whether the morning, evening or night shift is worked. However, not all shifts are of the same length: their duration varies between 6 and 9 hours. Moreover, there are public holidays, and on those days the hours worked are reduced to the level of a Sunday in the relevant period.

Shifts are assigned according to the general principles described in the introduction. The following assignment conditions must also be met:

1. Drivers may only do one shift per calendar day.
2. Drivers must have at least one rest day per calendar week.
3. On two consecutive working days a driver may switch from morning shift to evening or night shift, and from evening shift to night shift, but not from evening shift to morning shift or from night shift to morning shift or evening shift, as drivers must have at least twelve hours' rest between two working days.
4. Drivers may not work more than seven consecutive days.
5. Rest periods must be at least 36 hours long. This means that drivers may not have sequences of "evening–rest–morning", "night–rest–morning" or "night–rest afternoon". Moreover, whenever possible, attempts will be made to ensure that drivers do not have isolated single rest days.
6. Shift changes on consecutive working days will take place in such a way that no driver works one single morning shift or evening shift.
7. Whenever possible, attempts will be made to ensure that drivers do not have isolated single working days.
8. A driver's morning, evening and night shifts and Sundays off should be distributed evenly over the year.
9. Shifts should be distributed in egalitarian fashion. At the end of the year drivers should have been assigned a similar number of working days and hours, morning, evening and night shifts and Sundays off.

3 Lists of weekly patterns

Of the 51 drivers with work planned, 42 will be active and 9 on vacation or on reserve duty each week. So for each period of the year (except Semana Grande week 2, in which there is a generalized increase in shifts for which 14 additional drivers need to be taken on, so the list is for 56 patterns) a list must be drawn up of 42 patterns per week containing all the shifts in the week. The lists for winter and summer patters must be ordered so that a single driver can work them consecutively. All the lists must take into account the requirements arising from the obligations and entitlements of the workers in regard to working conditions.

The subscript $i \in I = \{1, 2, \dots, 42\}$ indicates the pattern, $j \in J = \{1, \dots, 7\}$ indicates the day of the week and $k \in K = \{1, 2, 3\}$ indicates morning, evening or night shift, respectively. Morning shifts are denoted by M, evening shifts by E, night shifts by N, and rest days by R.

3.1 Winter

The superscript w indicates winter. We denote by $a^w_{j,k}$ the number of days of shift k that must be worked on day j in winter, as indicated in Table 1.

– Binary variables: $x^w_{i,j,k}$, $i \in I$, $j \in J$, and $k \in K$, is 1 if pattern i has day of the week j shift k, and 0, otherwise.

In winter there is a night shift only on Saturdays. We can thus set

$$x^w_{i,j,3} = 0, \quad i \in I,\ j = 1, 2, 3, 4, 5, 7.$$

– Constraints:

$$\sum_{k=1}^{3} x^w_{i,j,k} \le 1, \quad i \in I,\ j \in J; \tag{1}$$

$$\sum_{i=1}^{42} x^w_{i,j,k} = a^w_{j,k}, \quad j \in J,\ k \in K; \tag{2}$$

$$x^w_{i,j,2} + x^w_{i,j+1,1} \le 1, \quad i \in I,\ j = 1, 2, \dots, 6; \tag{3}$$

$$x^w_{i,7,2} + x^w_{i+1,1,1} \le 1, \quad i = 1, 2, \dots, 41; \tag{4}$$

$$x^w_{i,j,3} + x^w_{i,j+1,1} + x^w_{i,j+1,2} \le 1, \quad i \in I, j = 1, 2, \dots, 6; \tag{5}$$

$$x^w_{i,7,3} + x^w_{i+1,1,1} + x^w_{i+1,1,2} \le 1, \quad i = 1, 2, \dots, 41; \tag{6}$$

$$\sum_{k=1}^{3} \left(\sum_{j=3+h}^{7} x^w_{i,j,k} + \sum_{j=1}^{4+h} x^w_{i+1,j,k} \right) \le 8,$$

$$i = 1, 2, \dots, 41,\ h = 0, 1; \tag{7}$$

$$4 \le \sum_{j=1}^{7} \left(x^w_{i,j,1} + x^w_{i,j,2} + x^w_{i,j,3} \right) \le 5, \quad i \in I; \tag{8}$$

Table 1 Number of shift types in winter

	Winter: Weeks 1, 3–23 and 36–52		
	Morning shifts	Evening shifts	Night shifts
Monday	15	17	
Tuesday	15	17	
Wednesday	15	17	
Thursday	15	17	
Friday	15	17	
Saturday	9	8	3
Sunday	9	8	

Total number of shifts: 197

$$\sum_{k=1}^{3} x^w_{i,j,k} + \sum_{k=1}^{3} x^w_{i,j+2,k} - \sum_{k=1}^{3} x^w_{i,j+1,k} \leq 1, \quad i \in I,\ j = 1, 2, \ldots, 5; \tag{9}$$

$$\sum_{k=1}^{3} x^w_{i-1,7,k} + \sum_{k=1}^{3} x^w_{i,2,k} - \sum_{k=1}^{3} x^w_{i,1,k} \leq 1, \quad i = 2, 3, \ldots, 42; \tag{10}$$

$$\sum_{k=1}^{3} x^w_{i,6,k} + \sum_{k=1}^{3} x^w_{i+1,1,k} - \sum_{k=1}^{3} x^w_{i,7,k} \leq 1, \quad i = 1, 2, \ldots, 41; \tag{11}$$

$$0 \leq x^w_{i,j,k} + x^w_{i,j+2,k} - x^w_{i,j+1,k}, \quad i \in I,\ j = 1, 2, \ldots, 5,\ k = 1, 2; \tag{12}$$

$$0 \leq x^w_{i,2,k} + x^w_{i-1,7,k} - x^w_{i,1,k}, \quad i = 2, 3, \ldots, 42,\ k = 1, 2; \tag{13}$$

$$0 \leq x^w_{i,6,k} + x^w_{i+1,1,k} - x^w_{i,7,k}, \quad i = 1, 2, \ldots, 41,\ k = 1, 2; \tag{14}$$

$$0 \leq \sum_{k=1}^{3} x^w_{i,j,k} + \sum_{k=1}^{3} x^w_{i,j+2,k} - x^w_{i,j+1,3}, \quad i \in I,\ j = 1, 2, \ldots, 5; \tag{15}$$

$$x^w_{i,6,3} + x^w_{i+1,6,3} + x^w_{i+2,6,3} + x^w_{i+3,6,3} + x^w_{i+4,6,3} + x^w_{i+5,6,3} \leq 1, \quad i = 1, 2, \ldots, 37; \tag{16}$$

$$\sum_{k=1}^{3} \left(x^w_{i,7,k} + x^w_{i+1,7,k} + x^w_{i+2,7,k} + x^w_{i+3,7,k}\right) \leq 2, \quad i = 1, 2, \ldots, 39; \tag{17}$$

$$5 \leq \sum_{j=1}^{7} \left(x^w_{i,j,1} + x^w_{i+1,j,1} + x^w_{i+2,j,1} + x^w_{i+3,j,1}\right) \leq 12, \quad i = 1, 2, \ldots, 39; \tag{18}$$

$$\sum_{j=1}^{7} \left(x^w_{i,j,2} + x^w_{i+1,j,2} + x^w_{i+2,j,2} + x^w_{i+3,j,2}\right) \geq 6, \quad i = 1, 2, \ldots, 39; \tag{19}$$

$$\sum_{j=1}^{7} \left(\sum_{k=1}^{3} \left(x^w_{i,j,k} + x^w_{i+1,j,k} + x^w_{i+2,j,k} + x^w_{i+3,j,k}\right) \right) \geq 18, \quad i = 1, 2, \ldots, 39; \tag{20}$$

$$x^w_{1,1,k} = 0, \qquad x^w_{1,2,k} = 0, \quad k = 1, 2. \tag{21}$$

Constraints (1–7) are essential. Constraints (1) and (2) ensure that each day each pattern is assigned no more than one shift, and that all shifts are assigned. Constraints (3–6) prevent a driver from being assigned the sequences *EM* and *NM*, *NE*. Constraint (7), combined with constraint (8), prevents drivers from working more than eight consecutive days. Constraints (8–15) are improvements for drivers on their strictly compulsory conditions. Constraint (8) ensures that there are two or three rest days in each calendar week. Constraints (9–11) ensure that there are no single isolated rest days. Constraints (12–15) prevent single, isolated morning or evening shifts and isolated working days. Constraints (16–20) impose a good distribution of work over time. Constraint (16) ensures that there is only one night shift in every six consecutive patterns. Constraints (17–20) ensure that there are no more than two working Sundays in every four consecutive patterns, between five and twelve morning shift days, at least six evening shift days and at least eighteen shifts in all. Constraint (21) ensures that the list of patterns is cyclical: a driver can work pattern 1 after working pattern 42.

3.2 Summer

The superscript s indicates summer. We denote by $a^s_{j,k}$ the number of days of shift k that must be worked on day j in summer, as indicated in Table 2.

– Binary variables: $x^s_{i,j,k}$, $i \in I$, $j \in J$, and $k \in K$.

– Constraints. The constraints are similar to the above: all that is needed is to adapt constraints (2), (7), (8), (16–20) to the fact that there are now more hours of work per week. Thus, in summer each weekly pattern has between four and six shifts, no more than three Sundays are worked in every five patterns, there are at least two nights worked in every four consecutive patterns, between five and fourteen morning shift days, at least seven evening shift days and at least nineteen shifts in all.

3.3 San Sebastián festival week

The superscript e indicates San Sebastián Festival Week. We denote by $a^e_{j,k}$ the number of days of shift k to be worked on day j of San Sebastián week, as indicated in Table 3.

– Binary variables: $x^e_{i,j,k}$, $i \in I$, $j \in J$, and $k \in K$.

Table 2 Number of shift types in summer

	Summer: Weeks 24–30 and 33–35		
	Morning shifts	Evening shifts	Night shifts
Monday	15	18	
Tuesday	15	18	
Wednesday	15	18	
Thursday	15	18	
Friday	15	18	
Saturday	9	12	5
Sunday	9	12	

Total number of shifts: 212

Table 3 Number of shift types in the week of San Sebastián

	Week of San Sebastián: Week 2		
	Morning shifts	Evening shifts	Night shifts
Monday	15	17	
Tuesday	15	17	
Wednesday	15	17	
Thursday	15	17	10
Friday	15	17	
Saturday	9	8	3
Sunday	9	8	
Total number of shifts: 207			

Table 4 Number of shift types in the Semana Grande 1

	Semana Grande 1:Week 31		
	Morning shifts	Evening shifts	Night shifts
Monday	15	18	
Tuesday	15	18	
Wednesday	15	18	
Thursday	15	18	
Friday	15	18	
Saturday	9	12	3
Sunday	9	15	13
Total number of shifts: 226			

– Constraints. In this case we are dealing with a period of only one week. Since patterns are assigned only once, the list of patterns does not need to be rotating. The constraints applicable here are (1), (2), (3), (5), (8), (9), (12) and (15).

3.4 Semana Grande festival weeks 1 and 2

The patterns for Semana Grande weeks 1 and 2 are obtained jointly. The superscript $G1$ indicates Semana Grande week 1 and $G2$ Semana Grande week 2. We denote by $a^{G1}_{j,k}$ the number of days of shift k to be worked on day j of Semana Grande week 1, and by $a^{G2}_{j,k}$ those of Semana Grande week 2, as indicated in Tables 4 and 5, respectively. Fourteen extra drivers have to be hired to handle the service provided in Semana Grande week 2.

– Binary variables: $x^G_{i,j,k}$, $x^G_{i_2,j,k}$, $i \in I$, $j \in J_2 = \{1, \dots, 14\}$, $k \in K$, and $i_2 \in I_2 = \{43, 44, \dots, 56\}$ (extra drivers). We also use the notation $x^{G1}_{i,j,k}$ and $x^{G2}_{i,j,k}$ instead of $x^G_{i,j,k}$ and $x^G_{i,j+7,k}$, $i \in I \cup I_2$, $j \in J$, $k \in K$, respectively.

Since the reserve drivers do not work in Semana Grande week 2 we have $x^G_{i_2,j,k} = 0$, $j \in J$.

– Constraints. Here also the list of patterns does not need to be rotating. The constraints used are the same as those

Table 5 Number of shift type in the Semana Grande 2

	Semana Grande 2: Week 32		
	Morning shifts	Evening shifts	Night shifts
Monday	15	20	13
Tuesday	15	20	13
Wednesday	15	20	13
Thursday	15	20	13
Friday	15	20	13
Saturday	9	15	13
Sunday	9	15	13
Total number of shifts: 314			

for San Sebastián festival week, but adapted to two weeks, with the following changes due to the increased number of shifts. In the two weeks the first 42 patterns must have at least three rest days, one or two of which must be in each week, there cannot be more than eight consecutive working days or six night-shift days. For the problem to be feasible the patterns may include isolated rest days, but may not include the sequences *ERM*, *NRM* or *NRT*. Constraints (9–11) are replaced by

$$x^G_{i,j-1,2} + x^G_{i,j+1,1} - \sum_{k=1}^{3} x^G_{i,j,k} \le 1,$$
$$i \in I \cup I_2,\ j \in \{2, \dots, 13\}; \qquad (22)$$

$$x^G_{i,j-1,3} + x^G_{i,j+1,1} + x^G_{i,j+1,2} - \sum_{k=1}^{3} x^G_{i,j,k} \le 1,$$
$$i \in I \cup I_2,\ j \in \{2, \dots, 13\}. \qquad (23)$$

In this case constraint (15) also needs to be eliminated, and isolated shifts of type *RNR* need to be allowed. Extra drivers only have one rest day, which must be neither a Saturday nor a Sunday.

4 Annual assignment of weekly patterns

The weeks of the year are identified by the subscript $l \in L = \{1, 2, \dots, 52\}$. Taking into account the service to be provided, the year is divided into five different periods: winter (weeks $L_W = \{1, 3, 4, \dots, 23, 36, \dots, 52\}$), summer (weeks $L_S = \{24, \dots, 30, 33, 34, 35\}$), San Sebastián festival week (week 2), Semana Grande festival week 1 (week 31) and Semana Grande festival week 2 (week 32). There are 51 drivers scheduled to work. Drivers are identified by the subscript $n \in N = \{1, 2, \dots, 51\}$. In each week there are 9 drivers on vacation or reserve duty. The nine drivers who have reserve duties in the first week occupy the last places on the list of drivers. Work is assigned weekly, using the various lists for the 42 weekly patterns as built up above. In winter and summer each driver is assigned patterns week by week in correlative order on the list.

We denote the number of working days by

$$\alpha_i^w = \sum_{j=1}^{7}\sum_{k=1}^{3} x_{i,j,k}^w \quad (\text{or } \alpha_i^s, \alpha_i^e, \alpha_i^{G1}, \alpha_i^{G2}),$$

the number of morning shift days by

$$\beta_i^w = \sum_{j=1}^{7} x_{i,j,1}^w \quad (\text{cf. } \beta_i^s, \beta_i^e, \beta_i^{G1}, \beta_i^{G2}),$$

the number of night shift days by

$$\gamma_i^w = \sum_{j=1}^{7} x_{i,j,3}^w \quad (\text{cf. } \gamma_i^s, \gamma_i^e, \gamma_i^{G1}, \gamma_i^{G2})$$

and the number of Sundays worked by

$$\lambda_i^w = \sum_{k=1}^{3} x_{i,7,k}^w \quad (\text{cf. } \lambda_i^s, \lambda_i^e, \lambda_i^{G1}, \lambda_i^{G2}),$$

all in winter pattern i (cf. summer, San Sebastián, Semana Grande 1 and Semana Grande 2 festival weeks). The reserve week is valued at five working days. Since only the last nine drivers have reserve duty, we define $\alpha_n^{RV} = 5$ if $n \geq 43$, and $\alpha_n^{RV} = 0$ if $n \leq 42$.

– Binary variables: $y_{n,l,i}$, $n \in N$, $l \in L$, and $i \in I$, is 1 if driver n has week l in pattern i, and 0, otherwise.

Vacations define the binary constants $b_{n,l}$, $n \in N$, $l \in L$, such that $b_{n,l} = 1$ if driver n is on vacation or reserve duty in week l, and $b_{n,l} = 0$, otherwise. Of course, we make the assignment

$$y_{n,l,i} = 0 \quad \text{if } b_{n,l} = 1,\ n \in N,\ i \in I.$$

In the first week the first 42 drivers are on active duties. We assign them the pattern correlative to their position:

$$y_{n,1,i} = \delta_{n,i}, \quad i \in I,$$
$$\text{where } \delta_{n,i} = 1 \text{ if } n = i, \text{ and } \delta_{n,i} = 0 \text{ if } n \neq i.$$

– Constraints:

We apply the usual constraints that ensure that each active driver is assigned a pattern for each week, and that all patterns are assigned. In the annual assignment there are six changes of period, six changes of list with 42 patterns per week. Each change generates a number of changeover constraints that prevent the sequences *EM*, *NM* and *NT*, and ensure that drivers do not work more than eight consecutive days and do not have isolated working days or rest days. These constraints are similar to (1), (2), (3), (5), (7), (9) and (15). For the problem to be feasible, isolated morning or afternoon shift days or isolated rest days must be allowed in some joins. In this last case the sequences *ERM*, *NRM* and *NRT* must be prevented. For all $n \in N$, the specific constraints for this problem are:

$$y_{n,31,i} = y_{n,32,i}, \quad \text{whenever } b_{n,31} = 0 \text{ and } b_{n,32} = 0,\ i \in I; \tag{24}$$

$$y_{n,l,i+1} = y_{n,l-1,i}, \quad \text{whenever } b_{n,l} = 0 \text{ and } b_{n,l-1} = 0,\ i = 1, \dots, 41, \tag{25}$$

$$y_{n,l,1} = y_{n,l-1,42}, \quad \text{whenever } b_{n,l} = 0 \text{ and } b_{n,l-1} = 0,\ l \in \{4, \dots, 23, 37, \dots, 52\} \cup \{25, \dots, 30, 34, 35\}; \tag{26}$$

$$205 \leq \sum_{i=1}^{42}\Bigg(\sum_{l\in L_w} y_{n,l,i}\alpha_i^w + \sum_{l\in L_s} y_{n,l,i}\alpha_i^s + y_{n,2,i}\alpha_i^e + y_{n,31,i}\alpha_i^{G1} + y_{n,32,i}\alpha_i^{G2} + \alpha_n^{RV}\Bigg) \leq 207; \tag{27}$$

$$89 \leq \sum_{i=1}^{42}\Bigg(\sum_{l\in L_w} y_{n,l,i}\beta_i^w + \sum_{l\in L_s} y_{n,l,i}\beta_i^s + y_{n,2,i}\beta_i^e + y_{n,31,i}\beta_i^{G1} + y_{n,32,i}\beta_i^{G2}\Bigg) \leq 100; \tag{28}$$

$$3 \leq \sum_{i=1}^{42}\Bigg(\sum_{l\in L_w} y_{n,l,i}\gamma_i^w + \sum_{l\in L_s} y_{n,l,i}\gamma_i^s + y_{n,2,i}\gamma_i^e + y_{n,31,i}\gamma_i^{G1} + y_{n,32,i}\gamma_i^{G2}\Bigg) \leq 8; \tag{29}$$

$$17 \leq \sum_{i=1}^{42}\Bigg(\sum_{l\in L_w} y_{n,l,i}\lambda_i^w + \sum_{l\in L_s} y_{n,l,i}\lambda_i^s y_{n,2,i}\lambda_i^e + y_{n,31,i}\lambda_i^{G1} + y_{n,32,i}\lambda_i^{G2}\Bigg) \leq 20. \tag{30}$$

Constraint (24) means that a driver who works both the Semana Grande festival weeks is assigned the same pattern number in each of them. Constraints (25) and (26) mean that the assignment of patterns in winter and in summer moves upward in order through the relevant list of patterns. Constraints (27–30) mean that drivers are assigned between 205 and 207 working days per year, between 89 and 100 morning shift days, between 3 and 8 night shift days, and between 17 and 20 working Sundays.

5 Public holidays

Once the solution to the problem of assigning weekly patterns to drivers is obtained we can deploy that solution in days to obtain an annual assignment of shifts. We, therefore, have the binary constants $d_{n,r,k}$, $n \in N$, $r \in D = \{1, 2, \dots, 364\}$, $k \in K$, defined in such a way that $d_{n,r,k} = 1$ if driver n is assigned shift k on day r, and $d_{n,r,k} = 0$, otherwise.

On public holidays services are reduced to the level of a Sunday in the corresponding period of the year. The total number of public holidays in winter is $D_{fw} = \{74, 75, 78, 113, 276, 296, 331, 333, 362\}$, and in summer $D_{fs} = \{197\}$. To reduce services on public holidays we introduce

– Binary variables: $z_{n,r,k}$, $n \in N$, $r \in D$, and $k \in K$, is 1 if driver n is assigned shift k on day rk, and 0, otherwise.

Therefore, only the following are modified on holidays

$$z_{n,r,k} = d_{n,r,k}, \quad n \in N,\ r \notin D_{fw} \text{ or } r \notin D_{fs},\ k \in K.$$

– Constraints:

$$\sum_{k=1}^{3} z_{n,r,k} \le 1, \quad n \in N,\ r \in D_{fw} \text{ or } r \in D_{fs}; \tag{31}$$

$$\sum_{n=1}^{51} z_{n,r,k} = a^{w}_{7,k}, \quad r \in D_{fw},\ k \in K; \tag{32}$$

$$\sum_{n=1}^{51} z_{n,r,k} = a^{s}_{7,k}, \quad r \in D_{fs},\ k \in K. \tag{33}$$

$$\text{If } d_{n,r,1} = 1 \text{ we have } z_{n,r,2} = 0,\ n \in N,\quad r \in D_{fw} \text{ or } r \in D_{fs}. \tag{34}$$

$$\text{If } d_{n,r,2} = 1 \text{ we have } z_{n,r,1} = 0,\ n \in N,\quad r \in D_{fw} \text{ or } r \in D_{fs}. \tag{35}$$

$$203 \le \sum_{r=1}^{264} \sum_{k=1}^{3} z_{n,r,k} + \alpha^{RV}_{n} \le 204, \quad n \in N; \tag{36}$$

$$87 \le \sum_{r=1}^{264} z_{n,r,1} \le 98, \quad n \in N; \tag{37}$$

$$96 \le \sum_{r=1}^{264} z_{n,r,2} \le 113, \quad n \in N. \tag{38}$$

Constraints (31–33) ensure that a driver cannot have two shifts on the same calendar day, and that all shifts are assigned. Constraints (34–35) ensure that on public holidays morning shifts are not transferred to evenings or evening shifts to mornings, and that there are merely reductions in service. Constraints (36–38) ensure a more balanced distribution of work. These constraints mean that drivers have between 203 and 204 working days per year, between 87 and 98 morning shift days, and between 96 and 113 evening shift days. Reductions in service on public holidays do not affect night shift days nor Sundays off. Thus, each driver works between 3 and 8 nights and between 17 and 20 Sundays per year. Note that introducing rest days on public holidays may result in isolated morning or evening shifts and isolated rest days, and working days.

6 Assignment of hours to shifts

Each day of the year is linked to a day of the week and to the period of the year, to which it belongs. The set of days in winter is denoted by D_w, the set for the summer by D_s, the set for San Sebastián festival week by D_e, the set for Semana Grande festival week 1 by D_{G1}, and the set for Semana Grande week 2 by D_{G2}. To specify, which day of the week j corresponds to a day r, and whether it belongs to winter (or summer, San Sebastián week, Semana Grande week 1 or Semana Grande week 2), we write $r(j, w)$ ($r(j, s)$, $r(j, e)$, $r(j, G1)$, or $r(j, G2)$). This assignment is clearly day-specific.

Not all morning, evening and night shifts have the same duration, and nor do shifts in different periods of the year: their duration varies between 6 and 9 hours. The different shift types with hours are identified by $q \in Q = \{1, 2, \dots, 60\}$. If $q \in Q_1 = \{1, 2, \dots, 20\}$, the shift is a morning shift; if $q \in Q_2 = \{21, 22, \dots, 40\}$, it is an evening shift; and if $q \in Q_3 = \{41, 42, \dots, 60\}$, it is a night shift. The set of different shifts of day j in winter (summer, San Sebastián week, Semana Grande week 1 and Semana Grande week 2) is denoted by Q^w_j (Q^s_j, Q^e_j, Q^{G1}_j, or Q^{G2}_j). Note that on each day in each period of the year there are fewer than the 60 shifts that we have indicated in Q. For instance, it can be seen from Table 1 that in winter $Q^w_1 = \{1, 2, \dots, 15, 21, 22, \dots, 37\}$, $Q^w_7 = \{1, 2, \dots, 9, 21, 22, \dots, 28\}$, and from Table 2 that in summer $Q^s_2 = \{1, 2, \dots, 15, 21, 22, \dots, 38\}$ and $Q^s_6 = \{1, 2, \dots, 9, 21, 22, \dots, 32, 41, 42, \dots, 45\}$. On public holidays the shifts are those of a Sunday in the relevant period. In Semana Grande week 2 only the first 42 patterns need to be considered. The shifts for those patterns are the corresponding first elements of Q_1, Q_2 and Q_3. The duration of shift $q \in Q^w_j$ is denoted by $\sigma^w_{q,j}$. Similarly, $\sigma^s_{q,j}$, $\sigma^e_{q,j}$, $\sigma^{G1}_{q,j}$, and $\sigma^{G2}_{q,j}$ denote the duration of shift q of day j in summer, San Sebastián week, Semana Grande week 1 and Semana Grande week 2. Reserve weeks are assigned the average number of hours per week, i.e., 38 hours 34 minutes (38.57 hours). In winter (summer) the hours of shift q from Monday to Friday are the same: $\sigma^w_{q,j} = \sigma^w_{q,h}$ ($\sigma^s_{q,j} = \sigma^s_{q,h}$), $j, h = 1, 2, \dots, 5$.

To assign working hours to shifts, we introduce

– Binary variables: $u_{n,r,q}$, $n \in N$, $r \in D$, $q \in Q$, is 1 if driver n is assigned working hours q on day r, and 0, otherwise.

The following direct assignments apply.

If there is no working hours variable q then variable u is 0:

$$\text{If } q \notin Q_j^w, \text{ then } u_{n,r(j,w),q} = 0, \; n \in N, \; r(j,w) \in D_w.$$

This applies for winter. The same can be said for the remaining periods of the year.

On rest days drivers cannot be assigned any kind of shift:

$$\text{If } \sum_{k=1}^{3} z_{n,r,k} = 0, \text{ then } u_{n,r,q} = 0, \; n \in N, \; r \in D \text{ and } q \in Q.$$

On morning shift days they can only be assigned a q of 20 or less:

$$\text{If } z_{n,r,1} = 1, \text{ then } u_{n,r,q} = 0, \; n \in N, \; r \in D, \text{ and } q > 20.$$

On afternoon shift days they can only be assigned a q between 21 and 40:

$$\text{If } z_{n,r,2} = 1, \text{ then } u_{n,r,q} = 0,$$
$$n \in N, \; r \in D \text{ and } q \le 20, \; q \ge 41.$$

On night shift days they can only be assigned a q greater than 40:

$$\text{If } z_{n,r,3} = 1, \text{ then } u_{n,r,q} = 0, \; n \in N, \; r \in D, \text{ and } q \le 40.$$

– Constraints:

$$\sum_{q=1}^{60} u_{n,r,q} \le 1, \quad n \in N, \; r \in D; \tag{39}$$

$$\sum_{n=1}^{51} u_{n,r(j,w),q} = 1, \quad r(j,w) \in D_w, \; q \in Q_j^w. \tag{40}$$

This last constraint is written for days in the winter period. For days in other periods of the year similar constraints are obtained, which are omitted here. Public holidays are treated the same as Sunday in the relevant week.

$$\begin{aligned}
&\text{If } \sum_{n=1}^{51} z_{n,r,1} = \sum_{n=1}^{51} z_{n,r+1,1}, \; \sum_{n=1}^{51} z_{n,r,2} = \sum_{n=1}^{51} z_{n,r+1,2}, \text{ and} \\
&\quad z_{n,r,1} = z_{n,r+1,1} = 1 \text{ or } z_{n,r,2} = z_{n,r+1,2} = 1, \\
&\quad \text{with } n \in N, r \in \{1, 2, \ldots, 363\}, \\
&\quad \text{we have } u_{n,r,q} = u_{n,r+1,q}.
\end{aligned} \tag{41}$$

$$\begin{aligned}
1563 \le &\sum_{r(j,w)\in D_w} \sum_{q \in Q_j^w} u_{n,r(j,w),q}\sigma_{q,j,w} \\
&+ \sum_{r(j,s)\in D_s} \sum_{q \in Q_j^s} u_{n,r(j,s),q}\sigma_{q,j,s} \\
&+ \sum_{r(j,e)\in D_e} \sum_{q \in Q_j^e} u_{n,r(j,e),q}\sigma_{q,j,e} \\
&+ \sum_{r(j,G1)\in D_{G1}} \sum_{q \in Q_j^{G1}} u_{n,r(j,G1),q}\sigma_{q,j,G1} \\
&+ \sum_{r(j,G2)\in D_{G2}} \sum_{q \in Q_j^{G2}} u_{n,r(j,G2),q}\sigma_{q,j,G2} \\
&+ 38.57\delta_n \le 1573, \quad n \in N,
\end{aligned} \tag{42}$$

where $\delta_n = 1$ if $43 \le n \le 51$ (drivers on reserve duty), and $\delta_n = 0$ if $1 \le n \le 42$.

Constraints (39) and (40) ensure that no driver is assigned two sets of working hours on the same day, and that all working hours are assigned each day. If the number of morning and evening shift days on one day of the week is the same as on the following day, the number of hours is the same. So that drivers' working timetables are homogeneous, constraint (41) ensures that drivers who work those two morning shift or evening shift days are assigned the same working hours. Constraint (42) ensures that the each driver works between 1563 and 1573 hours per year.

7 Computational experiments

The top priorities of the firm and the workers regarding the annual distribution of workload are equality in the number of days and hours worked. These are followed by equal numbers of Sunday rest days, morning and evening shifts and night shifts, in that order. In the final solution obtained, the workload for each driver is as shown in Table 6. To reach these final figures we had to carry out a number of trials. In the first trial, when we built up the ordered sets of the weekly patterns for winter (summer), we did not impose any conditions for the solution to provide a good distribution of work between consecutive patterns. Specifically, we did not apply constraints (16–20). Solving the whole problem with the resulting sets of patterns for winter and summer gives an annual final solution with an equal number of days worked, but for the problem to be feasible, the bounds in constraint (30), which gives the number of Sundays worked, must be 14 and 23, and in constraint (28) they must be 80 and 110. To improve this result we included constraints (16–20), which force a good distribution over time of the type of workload in winter (summer). This not only improves the quality of the final solution but is also welcomed by drivers. Thus, we began by considering bounds in constraints (16–20) that were not too demanding: an upper bound of 3 in (17), a lower bound of 4 and an upper bound of 13 in (18), a lower bound of 5 in (19) and a lower bound of 15 in (20). We resolved this problem, improved the bounds again and resolved it again until we arrived at an unfeasible problem. In the end we kept

Table 6 Annual workload per driver

Working days	Working hours	Working Sundays	Morning shifts	Evening shifts	Night shifts
Between 203 and 204	Between 1563 and 1573	Between 17 and 20	Between 87 and 98	Between 96 and 113	Between 3 and 8

Table 7 Computational aspects

	Problem 1 (Summer)	Problem 2	Problem 3	Problem 4
Binary variables	918	93 870	21 646	530 465
Constraints	2553	368 986	19 441	182 919
Non-zero elements	17 550	1 396 454	90 308	2 071 173
CPU-time	4 min 10 s	6 h 20 min	11 s	3 h 31 min

the ordered sets of weekly patterns for winter (summer) obtained with the bounds written into constraints (16–20) in Sect. 3.

With these ordered sets we drew up the annual assignment of patterns to drivers. We began by imposing relaxed bounds in constraints (27–30): 195 and 215 in (27), 80 and 110 in (28), 2 and 10 in (29) and 15 and 22 in (30). We resolved this problem of annual assignment, improved the bounds of (27) and (30), resolved again, improved the bounds of (28) and (29), resolved again and repeated the process until we arrived at an unfeasible problem. In this process we reached the feasible bounds written into constraints (27–30) in Sect. 4.

Adjusting the bounds of constraints (36–38) in the problem of elimination of service on nonworking days (Sect. 5) is straightforward. Finally, for the assignment of hours to the patterns we began by setting bounds of 1555 and 1580 in constraint (42) and obtained a solution in 30 min of CPU time. We adjusted the bounds, resolved the problem, readjusted, and so on, until the problem was unfeasible. We thus arrived at the bounds of 1563 and 1573 that appear in constraint (42) in Sect. 5. This final solution fits the priorities indicated above and is satisfactory to all the parties involved.

Observe that to obtain an equitable annual solution we need not just to choose the right bounds in constraints (27–30), (36–38) and (42), but also to begin with balanced ordered sets of weekly patterns for winter (summer), such as those provided by constraints (16–20).

Throughout this description, a problem is considered unfeasible if it effectively has no solution or if the commercial software used is incapable of finding a solution within a reasonable time (set at 36 hours). In all cases, the more relaxed the bounds are the less CPU time is, of course, needed to resolve the problem.

The four types of binary programming problem that lead to the final solution are solved using Lingo commercial software on a Pentium IV PC with a clock-speed of 2.26 GHz. Table 7 shows the computational aspects of the problems solved to obtain the final solution in Table 6. For problem 1 we show the list for the summer, which was the hardest to draw up. Evidently, if we relax the bounds of constraints (27–30), (36–38) and (42), the CPU time will decrease.

8 Final comments

With the solution proposed, at the end of the year EuskoTren assigns workloads to drivers for the following year. Apart from the advantages of the fair distribution achieved, this means that drivers know in advance their working hours for the whole year. The assignment drawn up corresponds to ordinary, scheduled work, but passenger transport companies can be faced with unforeseen requirements resulting from local holidays, sports events, large-scale concerts, etc. To meet such requirements the company assigns a higher workload to those drivers willing to take it, counting the hours as overtime if the driver has already been assigned his/her full complement of hours, or else hires additional drivers.

A look at the solution obtained raises the possibility of whether the result, i.e., the quality of the solution, could be improved. The main difficulty encountered in this regard lies in the limitations of the commercial software available.

Finally, it is worth noting that the method proposed is not too difficult for it to be implemented by a company's IT technician.

Acknowledgements The authors would like to thank Gorka Ugalde, Juan Carlos Paredes and Julián Medina of EuskoTren for their help in resolving this problem.

This study was funded by EuskoTren and performed under the management of Fundación Euskoiker.

References

Azmat, C. S., Hürlimann, T., & Widmer, M. (2004). Mixed integer programming to schedule a single-shift workforce under annualized hours. *Annals of Operations Research*, *128*, 199–215.

Beaumont, N. (1997). Using mixed integer programming to design employee rosters. *Journal of the Operations Research Society*, *48*, 585–590.

Burke, E. K., De Causmaecker, P., Vanden Berghe, G., & Van Landeghem, H. (2004). The state of the art of nurse rostering. *Journal of Scheduling*, *7*, 441–499.

Caprara, A., Fischetti, M., Guida, P. L., Toth, P., & Vigo, D. (1997). Algorithms for railway crew management. *Mathematical Programming*, *79*, 125–141.

Caprara, A., Toth, P., Vigo, D., & Fischetti, M. (1998). Modeling and solving the crew rostering problem. *Operations Research*, *46*, 820–830.

Caprara, A., Fischetti, M., Guida, P. L., Toth, P., & Vigo, D. (1999). *Lecture notes in economics and mathematical systems: Vol. 471. Solution of large-scale railway crew planning problems: the Italian experience*. Berlin: Springer.

Ernst, A., Jiang, H., Krishnamoorthy, M., Nott, H., & Dier, D. (2000). Rail crew scheduling and rostering: optimisation algorithms. In *Computer aided scheduling of public transport, 8th international conference*, Berlin.

Ernst, A., Jiang, H., & Krishnamoorthy, M. (2004a). Staff scheduling and rostering: a review of applications, methods and models. *European Journal of Operational Research*, *153*, 3–27.

Ernst, A., Jiang, H., Krishnamoorthy, M., Owens, B., & Sier, D. (2004b). An annotated bibliography of personnel scheduling and rostering. *Annals of Operations Research*, *127*, 21–144.

Esclapés, C. (2000). *Asignación de conductores a jornadas de trabajo en empresas de transporte colectivo*. PhD thesis, Universitat Politécnica de Catalunya, Spain.

Isken, M. W. (2004). An implicit tour scheduling model with applications in healthcare. *Annals of Operations Research*, *128*, 91–109.

Laporte, G. (1999). The art and science of designing rotating schedules. *Journal of the Operations Research Society*, *50*, 1011–1017.

Laporte, G., & Pesant, G. (2004). A general multi-shift scheduling system. *Journal of the Operations Research Society*, *55*, 1208–1217.

Lezaun, M., Pérez, G., & Sáinz de la Maza, E. (2006). Crew rostering problem in a public transport company. *Journal of the Operations Research Society*, *57*, 1173–1179.

Lindo Systems Inc. (2003). Lingo User's Guide, Chicago.

Muslija, N., Gärtmer, J., & Slany, W. (2002). Efficient generation of rotating workforce schedules. *Discrete Applied Mathematics*, *118*, 85–98.

Tharmmaphornphilas, W., & Norman, B. A. (2004). A quantitative method for determining proper job rotation intervals. *Annals of Operations Research*, *128*, 251–266.

Townsend, W. (1988). An approach to bus-crew roster design in London regional transport. *Journal of the Operations Research Society*, *6*, 543–550.