

### 3 Rostering: Estado del arte

En este capítulo se hace una clasificación de los artículos y publicaciones más relevantes sobre el tema de la organización de horarios atendiendo al tipo de problema que resuelve.

Hay que resaltar que se ha tomado para ello una bibliografía muy amplia teniendo en cuenta que el problema de la organización de horarios tanto por sus características como por las técnicas que se utilizan para abordarlo es de una gran extensión.

Se han excluido deliberadamente de esta clasificación todos los trabajos que tratan de los problemas previos a la construcción de horarios, como pueden ser la elaboración de las tareas, la agrupación de las mismas, etc. También se ha evitado en la medida de lo posible los artículos y trabajos que tratan de la elaboración de horarios para tareas de estructura no cíclica como DAY, P.R. y otros (1997).

Se ha querido recoger en este estado del arte un espectro de problemas mucho más amplio del que realmente sería el objetivo del presente trabajo ya que técnicas y resultados obtenidos para estos pudieran ser aplicables o dar luz sobre el modelo propuesto.

Cada uno de los grupos y subgrupos que se han utilizado para esta clasificación vienen descritos mediante un modelo básico en el que se destacan los datos de partida o parámetros del problema y las variables o resultados.

A continuación y para cada tipo de problema se dan una serie de referencias por orden cronológico y una breve descripción de las mismas.

#### 3.1 Clasificación según tipo de problema

##### **A Número de personas por patrón**

A1 Con el mínimo número de personas

A2 Valorando cada patrón

A3 Valorando el excedente de personal

##### **B Asignación de personas**

B1 Una tarea a una persona

B2 Una tarea a varias personas

B3 Varias tareas a una persona

##### **C Agrupación de jornadas de trabajo**

##### **D Creación de patrones**

D1 Day off

D2 Multiple Shift - Day off

D3 Hierarchical - Day off

## A Número de personas por patrón

El modelo organiza los *periodos* libres (Day Off) de toda la plantilla para poder garantizar que cada día se presenten a trabajar el número suficiente de empleados para cubrir toda la demanda.

Se observa que si en un mismo periodo se realizan distintos tipos de actividad, o se requiere personal de distintas categorías, el modelo no determina la actividad que llevará a cabo cada trabajador sino únicamente los periodos libres que tendrá.

También cabe destacar que los patrones vienen ya determinados en los datos de partida, puede que se haga mención explícita de ellos como un vector binario o puede que se definan implícitamente imponiendo condiciones del tipo "cinco días de trabajo consecutivos". El objeto del problema es determinar de entre todos los patrones posibles cuales serán realmente utilizados y cuantas veces o cuantas personas realizarán cada patrón.

### A1 Con el mínimo numero de personas

#### Datos de partida

- **Patrones prefijados**

$A$  matriz de  $m \times n$

Donde:  $m$  es el número de periodos

$n$  es el número de patrones

$$a_{ij} = \begin{cases} 0 & \text{si el periodo } t \text{ no es de trabajo para el patrón } j \\ 1 & \text{si el periodo } t \text{ es de trabajo para el patrón } j \end{cases}$$

- **Demanda**

$d$  vector columna de  $m$  componentes

Donde:  $d_t$  es la demanda de personal para el periodo  $t$

#### Resultado

La solución determinará cuantas personas hay que asignar a cada patrón para cubrir la demanda de todos los periodos con el mínimo número de trabajadores.

$x$  vector columna de  $n$  componentes

Donde:  $x_j$  representa el número de personas que serán asignadas al patrón  $j$

#### Modelo Básico

$$\text{Min} \quad \sum_{j=1}^n x_j$$

$$\sum_{j=1}^n a_{tj} x_j \geq d_t \quad t = 1 \dots m$$

$$0 \leq x_j \quad \text{entero} \quad j = 1 \dots n$$

## **Ejemplos**

### **BAKER, K.R. (1974-a) y BAKER, K.R. (1974-b)**

Siguiendo la estructura descrita en el modelo básico, hace lo que podríamos llamar el negativo del problema, es decir considera en los patrones no los periodos de trabajo sino los de descanso y en lugar de hablar de la demanda de personal requerida, habla del número máximo de personas que pueden tener fiesta simultáneamente en cada periodo. Para ello parte de lo que denomina  $W = workforce$  o mano de obra, es decir del número de trabajadores necesario para cubrir la demanda.

Parte de unos patrones muy sencillos inspirados en el conocido Kleen City Problem. En concreto se trata de cubrir con patrones donde se trabaja cinco días consecutivos a la semana, la demanda producida por una empresa que ofrece sus servicios siete días a la semana.

En BAKER, K.R. (1974-b) estudia el caso en que el problema no tiene solución contando únicamente con trabajadores a tiempo completo. Y da un método para resolverlo de forma óptima contando también con trabajadores a tiempo parcial. Hay que observar que el autor lo primero que hace es calcular  $W$  la plantilla mínima necesaria para cubrir la demanda. No obstante no puede garantizar que con este número se logre formar una solución contando solo con patrones a tiempo completo. Es este sentido al que se refiere la expresión: *el problema no tiene solución* ya que evidentemente con más de  $W$  trabajadores siempre tendría solución.

### **BARTHOLDI, J.J. y otros (1978)**

A partir de los patrones de trabajo construye lo que podríamos llamar patrones de descanso, sustituyendo los ceros por unos y los unos por ceros. A partir de la plantilla mínima es posible saber el número de personas que pueden estar de descanso en cada periodo.

Es posible explotar la estructura particular de los patrones de descanso para resolver el problema o la serie de problemas como *flujo en redes* o un problema de *matching*. En concreto un patrón que tuviera dos días libres por semana se podría transformar en un patrón con dos unos en cada columna que luego se podría interpretar como un arco en una red.

### **MORRIS, J.G. y otros (1983)**

En MORRIS, J.G. y otros (1983) se propone mezclar los dos planteamientos clásicos de *set covering*, por un lado el que considera las variables  $x_j$  como patrones de horas a lo largo de un día y por otro el que considera  $x_j$  como patrones de días dentro de una semana.

En este artículo, las variables  $x_j$  son patrones de horas dentro de una semana, incluyendo las restricciones de que un mismo trabajador siempre tiene que hacer el mismo horario y que debe trabajar cinco días consecutivos a la semana.

### **BECHTOLD, S.E. y otros (1987)**

Resuelve el mismo problema que BAKER, K.R. (1974-a) pero mediante un procedimiento heurístico.

### **BECHTOLD, S.E. y otros (1991)**

Compara entre sí tres métodos de resolución basados en programación lineal y tres métodos constructivos para resolver el modelo básico. Evalúa sobre las distintas soluciones halladas algunos de los objetivos secundarios como un mayor porcentaje de días de fiesta consecutivos, el número de patrones utilizados, etc.

**BRUSCO, M.J. y JACOBS, L.W. (1993)**

Resuelve el modelo básico mediante técnicas de *recocido simulado*. A pesar de que en principio los patrones parecen estar definidos implícitamente, a la hora de buscar una solución vecina para ser evaluada se está procediendo como si fueran explícitos.

**BECHTOLD, S.E. y BRUSCO, M.J. (1994)**

Cuando el modelo básico tiene muchas columnas, es decir muchos patrones diferentes, no se suelen considerar todos, sino que mediante un proceso previo se elige un subconjunto de patrones de trabajo.

El criterio o los criterios utilizados para esta selección son muy variados, desde el que intenta que en el subconjunto de trabajo haya patrones lo más diferentes posible, hasta el procedimiento que escoge los de mayor productividad.

Este artículo analiza una serie de procedimientos para generar el subconjunto de patrones de trabajo, comparando los resultados entre sí.

## A2 Valorando cada patrón

El modelo es prácticamente idéntico al descrito en A1 pero se añade una valoración  $c_j$  distinta para cada patrón.

### Modelo Básico

$$\begin{aligned} \text{Min} \quad & \sum_{j=1}^n c_j x_j \\ \sum_{j=1}^n a_{tj} x_j & \geq d_t \quad t = 1 \dots m \\ 0 \leq x_j \quad & \text{entero} \quad j = 1 \dots n \end{aligned}$$

### Ejemplos

#### **SEGAL, M. (1974)**

Los patrones que presenta este artículo son horarios, los periodos son de un cuarto de hora y la demanda varía en cada periodo.

Primero hace una simplificación considerando solamente algunos patrones, aumenta el periodo a media hora y no considera los tiempos de descanso entre medio de un patrón. Resuelve el problema de *set covering* expuesto en el modelo básico a base de transformarlo en un problema de *flujo en redes con restricciones de capacidad*.

#### **MILLER, H.E. y otros (1976)**

Para definir un patrón válido hay una serie de reglas que debe cumplir obligatoriamente, pero también hay otra serie de cualidades que lo hacen más o menos apetecible.

La función objetivo se plantea en este caso como una mezcla entre la penalización que se impone a un patrón por incumplir alguno de los requisitos no obligatorios y la diferencia entre el número "ideal" de personas que deben trabajar en cada periodo y el número de las que realmente trabajan. Esta diferencia también se valora de forma distinta si se queda por debajo o por encima del número mínimo de personas requeridas en cada periodo.

#### **SHEPARDSON, F. and MARSEN, R.E. (1980)**

Estudia el caso particular en el que todos los patrones constan de dos periodos consecutivos de trabajo (Two Duty Period) demuestra que se puede reducir al caso de un solo periodo y a su vez este se puede reducir a la búsqueda del *camino mínimo* en un grafo.

#### **BARTHOLDI, J.J. y otros (1980)**

Estudia el caso en que todos los patrones constan de un determinado numero de periodos de trabajo consecutivos  $k$  de entre  $m$  de ellos. Por ejemplo  $(k, m) = (5, 7)$  significaría que de cada 7 días, 5 deben ser de trabajo y 2 deben ser días de fiesta consecutivos.

#### **BARTHOLDI, J.J. (1981)**

Profundiza en un tipo de patrones a los que denomina circulares y propone para resolverlos, varios procedimientos de redondeo a partir de la solución de la relajación continua del problema.

**BECHTOLD, S.E. (1988)**

Presenta un modelo implícito para determinar el número de trabajadores a tiempo parcial y completo que hay que asignar a cada patrón. El modelo contempla también distintos puestos de trabajo y distintos horarios.

**PAIXÃO, J. and PATO, M. (1989)**

Los periodos son en este caso franjas horarias y el horizonte temporal se limita a un día. Por este motivo todos los patrones tienen una única franja de descanso y están de este modo formados por dos series consecutivas de *unos*.

Considera también que el número de personas que se puede asignar a cada patrón está acotado superiormente.

**EASTON, F.F. and ROSSIN, D.F. (1991)**

Determina el número de empleados que deben ser asignados a cada plantilla mezclando distintas categorías profesionales, con lo que añade una restricción indicando la proporción máxima de empleados de cada categoría que puede haber.

**BRUSCO, M.J. and JACOBS, L.W. (1998)**

Aborda el problema de los turnos continuos, donde las jornadas de trabajo pueden extenderse de un día a otro. En concreto plantea los distintos patrones que pueden realizarse trabajando un número determinado de periodos consecutivos durante un número consecutivo de días.

En este trabajo Brusco & Jacobs proporcionan un algoritmo para reducir considerablemente el número de columnas del problema de *set covering* cuando existen periodos de demanda cero.

### A3 Valorando el excedente de personal

La demanda  $d_t$  no tiene por qué saturarse estrictamente para todos los periodos, puede darse el caso de que para cubrir la demanda en unos periodos, deba sobrar personal en otros.

A la diferencia entre la demanda y el personal que trabaja efectivamente en un periodo determinado se le denomina *excedente*.

De forma análoga, se puede permitir en el modelo que la demanda no se vea satisfecha totalmente, en este caso se penalizaría la *escasez* o *déficit* de personal.

#### Datos de partida

A los datos de partida ya descritos en el modelo A1 habría que añadir:

- **Valoración del excedente y el déficit de personal**

$e$  vector fila de  $m$  componentes

Donde:  $e_t$  es el coste de cada persona excedente para el periodo  $t$

$f$  vector fila de  $m$  componentes

Donde:  $f_t$  es el coste del déficit para el periodo  $t$

- **Limites superior e inferior para el excedente**

$s$  vector columna de  $m$  componentes

Donde:  $s_t$  es el excedente máximo permitido para el periodo  $t$

$r$  vector columna de  $m$  componentes

Donde:  $r_t$  es el déficit máximo permitido para el periodo  $t$

#### Modelo Básico

$$\text{Min} \quad \sum_{j=1}^n c_j x_j + \sum_{t=1}^m e_t y_t + \sum_{t=1}^m f_t z_t$$

$$\sum_{j=1}^n a_{tj} x_j + y_t - z_t = d_t \quad t = 1 \dots m$$

$$y_t \leq s_t \quad t = 1 \dots m$$

$$z_t \leq r_t \quad t = 1 \dots m$$

$$0 \leq x_j, y_t, z_t \quad \text{enteros}$$

#### Ejemplos

##### **BAKER, K.R. (1976)**

Recoge exactamente este modelo básico en su recopilación.

##### **MABERT, V. and WATTS (1982)**

La escasez de personal en un periodo determinado dejará una serie de tareas por hacer, dichas tareas se cuantifican en *número de items por procesar*. El modelo que aquí se presenta permite que los items no procesados de un periodo se puedan procesar en el siguiente, dentro del mismo día.

Esta espera tiene una penalización pero de alguna manera el excedente de unos periodos está contrarrestando la escasez de otros.

**THOMPSON, G.M. (1995)**

La demanda de personal en cada periodo es en muchas ocasiones un parámetro que no es posible conocer de antemano con certeza, ya que depende de factores que no son controlables por la empresa. Ello obliga a trabajar con unos valores de demanda aproximados que se estiman estadísticamente a partir de unas muestras representativas de observaciones.

Thomson presenta un modelo donde un pequeño excedente de personal es valorado positivamente porque puede mejorar el servicio, pero la mejora no es un valor constante sino que disminuye a medida que se aleja del valor previsto y acaba siendo valorada negativamente cuando se aparta demasiado.

## B Asignación de personas

Se agrupan bajo este nombre los problemas que determinan no el tamaño de la plantilla necesaria sino las personas concretas que realizarán cada una de las tareas.

Bajo el nombre de tareas se puede estar haciendo referencia a patrones de días libres, tipos de tareas, turnos horarios, etc...

### B1 Una tarea a una persona

Cada tarea debe ser asignada a un solo trabajador y cada trabajador podrá realizar una sola tarea.

#### Datos de partida

- **Coste (o beneficio) de la asignación**

$C$  matriz de  $p \times p$

Donde:  $c_{ij}$  es el coste (o beneficio) de asignar al trabajador  $i$  la tarea  $j$

#### Resultado

La solución determinará la tarea concreta que debe realizar cada persona o bien el tipo de tarea o el patrón de días libres que debe seguir cada persona.

$Y$  matriz de  $p \times p$

Donde:  $y_{ji} = \begin{cases} 0 & \text{si la tarea } j \text{ no es asignada al trabajador } i \\ 1 & \text{si la tarea } j \text{ es asignada al trabajador } i \end{cases}$

#### Modelo Básico

$$\text{Min} \quad \sum_{i=1}^p \sum_{j=1}^p c_{ij} y_{ji}$$

$$\sum_{i=1}^p y_{ji} = 1 \quad j = 1 \dots p$$

$$\sum_{j=1}^p y_{ji} = 1 \quad i = 1 \dots p$$

$$y_{ji} \in \{0,1\} \quad \begin{matrix} j = 1 \dots p \\ i = 1 \dots p \end{matrix}$$

## B2 Una tarea a varias personas

La realización de una misma tarea puede requerir a varios trabajadores pero un trabajador solo puede ser asignado a una tarea.

Puede plantearse como la continuación de los problemas de tipo A. Una vez determinado que el número de personas que se debe asignar al patrón  $j$  es  $x_j$  hay que decidir de entre los  $p$  trabajadores de que se dispone cuales serán asignados a dicho patrón.

En esta clasificación pudiera parecer que falta el caso *Varias a Varias*, cuando una persona puede hacer más de una tarea y a su vez esa tarea requiere la presencia de más de un trabajador. Pero examinando más detenidamente este caso se observa que muy difícilmente sería posible asignar cualquier subconjunto de tareas a un trabajador cualquiera, es decir primero tendríamos que definir unos patrones válidos de *tareas compatibles* entre sí que pudieran ser asignadas a un mismo trabajador, con lo cual se vuelve al caso *Una a Varias* pues un trabajador podría ser asignado a un solo patrón de *tareas compatibles*.

En este caso se suelen considerar otras restricciones adicionales como por ejemplo la que reflejara que con todos los subconjuntos de tareas compatibles finalmente seleccionados, quedara cada una de las tareas cubierta por el número de personas necesario.

Se usa el termino *tarea* pero realmente puede ser un *tipo de tarea, patrón...*

### Datos de partida

- **Número de personas que hay que asignar a cada tarea**

$b$  vector columna de  $n$  componentes

Donde:  $b_j$  es el numero de personas necesario para realizar la tarea  $j$

- **Coste (o beneficio) de la asignación**

$C$  matriz de  $p \times n$

Donde:  $c_{ij}$  es el coste (o beneficio) de asignar al trabajador  $i$  la tarea  $j$

### Resultado

La solución determinará la tarea concreta que debe realizar cada persona o bien el tipo de tarea o el patrón de días libres que debe seguir cada persona.

$Y$  matriz de  $n \times p$

Donde:  $y_{ji} = \begin{cases} 0 & \text{si la tarea } j \text{ no es asignada al trabajador } i \\ 1 & \text{si la tarea } j \text{ es asignada al trabajador } i \end{cases}$

### Modelo Básico

$$\text{Min} \quad \sum_{i=1}^p \sum_{j=1}^n c_{ij} y_{ji}$$

$$\sum_{i=1}^p y_{ji} = b_j \quad j = 1 \dots n$$

$$\sum_{j=1}^n y_{ji} = 1 \quad i = 1 \dots p$$

$$y_{ji} \in \{0,1\} \quad \begin{matrix} j = 1 \dots n \\ i = 1 \dots p \end{matrix}$$

## Ejemplos

### **RYAN, D.M. 1992**

Propone un ejemplo para la compañía aérea *Air New Zealand Ltd*

Mediante un sencillo algoritmo enumerativo se generan los diferentes *subconjuntos de vuelos* que puede realizar cada miembro de la tripulación.

Hay más de 30 reglas que debe cumplir cualquier *subconjunto* válido, una de las más importantes es que cada *patrón* de 28 días debe contener como mínimo 10 días libres.

Dado el gran número de combinaciones que existe, no se generan todos los *patrones* sino sólo aquéllos que pasan una serie de filtros adicionales. Por ejemplo:

- Tener un número de días libres inferior o igual a un número máximo que es posible calcular con anterioridad.
- Se pueden obviar todos los *subconjuntos* que tengan una secuencia no deseada de viajes.
- En general se escogerán *subconjuntos* que enlacen viajes lo más próximos posibles (respetando el descanso obligatorio).

A pesar de todo, considerando un horizonte temporal de 28 días, el número de *subconjuntos* posibles por trabajador puede oscilar entre 400 y 500.

Introduce en el modelo básico la matriz  $A$  matriz de  $m \times n$  donde

$$a_{kj} = \begin{cases} 0 & \text{si el subconjunto } k \text{ no incluye el viaje } j \\ 1 & \text{si el subconjunto } k \text{ incluye el viaje } j \end{cases}$$

y mediante las restricciones

$$\sum_{k=1}^m \sum_{i=1}^p a_{kj} y_{jk} = b_j \quad j = 1 \dots n$$

garantiza que en cada viaje  $j$  vayan  $b_j$  trabajadores.

### **BECHTOLD, S.E. and BRUSCO, M.J. (1994-b)**

Partiendo de patrones horarios y de una demanda horaria para cada día de la semana, resuelve 7 problemas distintos, determinando cuanta gente debe ser asignada a cada patrón horario. Hay patrones horarios para trabajadores a tiempo completo y a tiempo parcial con una dedicación variable ( 1 hora /día, 2 horas/día,...)

A demás del coste de cada uno de estos patrones tiene en cuenta que hay unos patrones más deseables que otros, por ejemplo los que empiezan a primera o a ultima hora. Esta consideración la tiene en cuenta optimizando por segunda vez el problema y buscando de entre todas las soluciones al mismo coste la que mejor se ajusta a este objetivo secundario.

### **B3 Varias tareas a una persona**

Es el problema que se presenta por ejemplo en una empresa donde hay puntas de trabajo que deben ser cubiertas por personal a tiempo parcial. Con estas puntas de trabajo se crean unas agrupaciones a las que denominamos tareas y estas tareas son realizadas por distintos operarios a tiempo parcial, pudiendo realizar evidentemente cada empleado más de una tarea.

#### **Ejemplo**

##### **GOPALKRISHNAN, M. y otros (1993)**

Da un sistema de soporte a la decisión para calcular primero las necesidades de personal a tiempo parcial y luego construir el programa de trabajo de estos empleados a tiempo parcial.

## C Agrupación de jornadas de trabajo

En este planteamiento no se habla de personas. Lo único que pretende, es agrupar las jornadas de trabajo a realizar en subconjuntos disjuntos, de modo que cualquier jornada de trabajo pertenezca a uno y solo uno de estos subconjuntos. Más tarde se asignará cada uno de estos subconjuntos a un trabajador.

Este es el planteamiento más habitual en las compañías aérea, en ese caso no se habla de jornadas de trabajo sino directamente de vuelos (*flight leg*).

Este tipo de problemas contempla también la secuenciación de las jornadas de trabajo dentro de un mismo subconjunto. No se puede olvidar que el problema de *Rostering* en una empresa de transporte colectivo trata de jornadas de trabajo que están ubicadas en el tiempo, muchas de ellas se repetirán diariamente, pero otras no.

### Datos de partida

- **Coste de realizar una jornada de trabajo después de otra.**

$C$  matriz de  $n \times n$

Donde:  $c_{ij}$  es el coste (o beneficio) de realizar la jornada  $i$  y a continuación la  $j$

En muchas ocasiones este coste se evalúa como el tiempo que transcurre desde que comienza la jornada  $i$  hasta comienza jornada  $j$

### Resultado

Se trata de determinar cómo se agruparán las jornadas de trabajo y en qué secuencia se realizarán.

$X$  matriz de  $n \times n$

Donde: 
$$x_{ij} = \begin{cases} 0 & \text{si la jornada } j \text{ no es realizada después de la } i \\ 1 & \text{si la jornada } j \text{ es realizada después de la } i \end{cases}$$

### Modelo Básico

El modelo se basa en un grafo dirigido  $G(V,A)$  cuyos vértices  $V$  representan las distintas jornadas de trabajo y los arcos  $A$  unen jornadas de trabajo que se pueden realizar de forma consecutiva.

Se trata de determinar caminos dentro de  $G$  tales que cualquier vértice pertenezca a un camino y sólo a uno. De manera que la suma de los costes asociados a los arcos utilizados sea mínima o bien que el coste del camino más caro sea mínimo.

Las jornadas de trabajo que pertenezcan al mismo camino serán jornadas de trabajo realizadas por el mismo trabajador. Los días libres se plantean de distintas formas según el autor, en algunos casos son sencillamente jornadas ficticias, en otros casos los caminos unen un número de nodos inferior al número de periodos en el horizonte de planificación, lo cual implica que haya periodos donde no se trabaje, etc.

## **Ejemplos**

### **CARRARESI, P. and GALLO, G. (1984)**

Agrupa y secuencia jornadas de trabajo con el objetivo de distribuir de la forma más equitativa posible las cargas de trabajo (duración de las jornadas) entre todos los trabajadores. Para ello utiliza un modelo *multilevel bottleneck assignment problem* que alcanza la solución cuando el camino más caro tiene un coste mínimo, es decir cuando el trabajador que trabaja más horas, logra trabajar las menos posibles.

### **BIANCO, L. and BIELLI, M (1992)**

Difiere del modelo básico en que el coste no va asociado al arco sino al nodo. El coste de un camino será la suma de los costes de todos los nodos por los que pasa.

Los días libres vienen representados por nodos ficticios a los que se le asigna un coste igual al coste medio de los demás nodos.

El procedimiento no trata de encontrar los caminos cuya suma sea mínima, sino los caminos tales que el camino más caro tenga un coste mínimo (*multilevel bottleneck assignment problem*).

### **VASCONCELOS, J. and CAMPOS, R. (1995)**

Parte de unas jornadas de trabajo que tienen cada una de ellas una duración distinta y que pueden empezar y terminar en horas distintas. Se trata de secuenciar dichas jornadas de forma que los tiempos de descanso entre jornadas sean lo más homogéneos posible.

Una vez establecida la secuencia óptima se asignará a cada trabajador con un día de decalaje. Los días libres se resuelven con corre- turnos.

Como todos los trabajadores acabarán haciendo todas las jornadas de trabajo, el problema que se intenta solucionar con esta secuenciación no es el reparto equitativo del trabajo entre los trabajadores sino el reparto de las horas de descanso entre jornadas. Como es posible calcular el tiempo de trabajo total sumando el tiempo de trabajo de todas las jornadas, se establece el tiempo de descanso total que dividido por el tiempo en el que deben realizarse todas las jornadas daría el tiempo medio de descanso entre jornadas.

Cada secuencia de jornadas de trabajo daría como resultado una observación distinta de horas de descanso. El modelo trata de hallar la secuenciación con varianza mínima respecto al tiempo de descanso medio calculado. Para ello establece los costes apropiados para pasar de una jornada a la siguiente y plantea el problema con la estructura del denominado problema del *viajante de comercio* (*The Traveling Salesman Problem*).

### **CAPRARA, A. y otros (1997-a)**

Utiliza una agrupación de jornadas tipo para determinar una cota inferior del tiempo total de jornales a pagar, incluyendo los días festivos a los que tienen derecho.

Cada par de jornadas tipo las secuencia de tres formas distintas. Directamente, con un día de descanso entremedio y con dos días de descanso entre ambas.

Se establecen las restricciones que garantizan el número mínimo de días de descanso, así como la proporción entre descansos simples (de un día) y dobles (de dos días). Dichas restricciones acaban incorporándose a la función objetivo para obtener la mencionada cota inferior.

Una vez calculada dicha cota inferior se procede mediante una heurística a agrupar y secuenciar efectivamente las jornadas de trabajo.

### **CAPRARA, A. y otros (1997-b)**

Da una visión de conjunto del problema de la organización de las plantillas en las empresas de transporte ferroviario. Dividiendo el problema en tres etapas consecutivas: la generación de jornadas de trabajo (*pairing generation*), la optimización de jornadas de trabajo (*pairing optimization*) y la optimización de las listas de tareas (*rostering optimization*).

### **CAPRARA, A. y otros (1998-b)**

Caprara et al. en 1998-b define y resuelve lo que él denomina Crew Rostering Problem mediante la combinación de técnicas de investigación operativa y programación de restricciones lógicas (*constraint logic programming*). Programa meses completos donde los empleados hacen cada día una tarea distinta. Las tareas son diarias por eso mientras un empleado hace la tarea asignada al día 1 hay otro que hace la tarea asignada al día 2 y así sucesivamente. Hay tantos empleados cubriendo cada *roster* como días tiene el *roster*.

### **VANCE, P.H. y otros (1997)**

Agrupar los vuelos de una compañía aérea con el fin de asignar después esas agrupaciones a distintas tripulaciones. Cada vuelo tiene entidad propia, es decir dos vuelos con el mismo origen y destino y con las mismas horas de salida y llegada, no se considerarán el mismo vuelo si se efectúan en día distinto.

### **DAY, P.R. and RYAN, D.M. (1997)**

Organiza el horario de la tripulación para los vuelos domésticos de una compañía aérea. En un primer paso los días de descanso se asignan al esqueleto del patrón y a continuación se asignan los servicios alrededor de los días de descanso.

### **BRAILSFORD y otros (1999)**

Como una aplicación más de las técnicas de propagación de restricciones plantea el problema de *rostering* como un problema de *satisfacción de restricciones*. El primer objetivo será seleccionar un conjunto de rosters de manera que el número de tripulaciones requeridas sea mínimo. El objetivo secundario será minimizar el número de días libres entre dos jornadas de la misma semana. Las variables son las jornadas y el dominio de cada variable es el conjunto de rosters donde se podría incluir dicha variable y la posición dentro de ese roster. Utiliza también otras variables adicionales asociadas con las jornadas precedentes y siguientes.

## D Creación de patrones

Dentro de este tipo de problemas se engloban aquellos que buscan los patrones más adecuados para satisfacer la demanda con el número mínimo de personal, repartiendo de forma equitativa el trabajo, maximizando la satisfacción de los trabajadores, etc.

### D1 Day off

Por un lado se consideran los problemas que solo resuelven los patrones de días libres, es decir una vez determinado el patrón se conocerán los periodos que serán de trabajo para las personas que se asigne a ese patrón, pero no qué tipo de trabajo hará.

El modelo básico pretende cubrir la demanda de personal con una plantilla lo mas pequeña posible, pero en muchas ocasiones el tamaño mínimo de la plantilla se puede calcular teóricamente al principio y el modelo lo único que hace es hallar una solución factible para ese tamaño.

#### Datos de partida

Reglas que debe cumplir cualquier patrón válido.

A modo de ejemplo:

- $A$  fines de semana libres de cada  $B$
- Una semana se trabajan 3 días y la siguiente 4
- No más de 10 días de trabajo seguidos

#### Resultado

El resultado consistirá en dar los patrones que mejor se ajusten a la demanda. Cada patrón será:

- $X$  vector binario de  $n$  componentes

Donde:  $n$  es el número de periodos a considerar

#### Ejemplos

##### **BAKER, K.R. and MAGAZINE, M.J. (1977)**

Da una fórmula para calcular el tamaño mínimo de plantilla y un algoritmo para encontrar una solución con ese tamaño mínimo.

Resuelve cuatro planteamientos diferentes:

- a) Dos días libres cada semana.
- b) Dos días libres consecutivos cada semana.
- c) Un fin de semana libre y otro no. Cuatro días libres cada dos semanas.
- d) Un fin de semana libre y otro no. Dos días libres consecutivos cada semana.

### **BAKER, K.R. y otros (1979)**

Determina el tamaño de la plantilla y crea patrones válidos para cubrir una demanda constante de  $n$  trabajadores al día, con las siguientes restricciones adicionales:

- Cada semana debe haber 5 días de trabajo y 2 de fiesta.
- $A$  fines de semana libres de cada  $B$
- No más de 6 días consecutivos de trabajo.
- Los periodos de trabajo deben ser lo más homogéneos posible.

### **BURNS, R.N. and CARTER, M.W. (1985)**

Determina el tamaño de la plantilla y crea patrones válidos para cubrir una demanda variable con las siguientes restricciones :

- Cada semana debe tener 5 días de trabajo (la semana empieza en domingo y acaba en sábado)
- $A$  fines de semana libres de cada  $B$
- No más de 6 días consecutivos de trabajo.

### **HUNG, R. (1991)**

Crea patrones para *semanas laborales comprimidas* es decir cada semana se trabaja menos de cinco días (por ejemplo 4 días) pero cada jornada es más larga (por ejemplo 10 horas/día).

Parte de una demanda  $D$  para los días laborables y  $E$  para los fines de semana.

Debe respetarse también la condición de  $A$  fines de semana libres de cada  $B$ .

### **BEAUMONT, N. (1997)**

Parte de una demanda variable para cada día de la semana. Busca patrones de muchas semanas de duración, por ejemplo: 47 semanas. El reparto equitativo del trabajo lo resuelve asignando el mismo patrón con un periodo de decalaje a cada empleado. Los patrones deben cumplir una serie de requisitos, a modo de ejemplo:

- No más de 7 ni menos de 3 días consecutivos de trabajo.
- En cada semana natural (de domingo a sábado) no se pueden trabajar más de 5 días.
- Los periodos de descanso deben ser de mínimo 2 días y máximo 4.
- La media de horas trabajadas a la semana será de 38.

La valoración de cada uno de los patrones que cumplen estas restricciones la hace atendiendo a los siguientes conceptos:

- Los trabajadores prefieren largos periodos de trabajo y largos periodos de descanso.
- El número de días trabajados cada mes (de 30 días) debe ser aproximadamente igual.
- El número de trabajadores que se presentan a trabajar cada día de la semana será proporcional a la demanda.

## D2 Multiple Shift – Day off

Este segundo tipo de planteamiento trata de buscar patrones que además de distinguir los periodos de trabajo y descanso, determine qué tipo de turno hará en los periodos de trabajo, por ejemplo (turno de mañana o turno de tarde)

### Datos de partida

Reglas que debe cumplir cualquier patrón válido.

A modo de ejemplo:

- Una semana se trabajan 3 días y la siguiente 4
- Un fin de semana libre de cada 4
- No más de 10 días de trabajo seguidos
- Entre cambios de turno debe haber como mínimo 2 días de descanso
- Preferencia de una secuencia determinada de turnos horarios, por ejemplo: Mañana – Tarde - Noche. Frente a otras posibles como puede ser: Mañana – Noche - Tarde).

### Resultado

Una vez codificados los distintos turnos. Por ejemplo:

0 – Día libre

1 – Turno de mañana

2 – Turno de tarde

3 – Turno de noche

El resultado consistirá en dar los patrones que mejor se ajusten a la demanda. Donde cada patrón será:

- $X$  vector  $n$  componentes

Donde:

$n$  es el número de periodos a considerar

$$x_i \in \{0,1,2,3\}$$

### Ejemplos

#### **TOWNSEND, W. (1988)**

Agrupa las jornadas de trabajo de los conductores de autobús en dos grupos dependiendo de se empiezan temprano o tarde. Parte de unos sub-patrones de duración variable (en concreto 2 patrones de 1 semana, 1 de 4 semanas y 1 de 5 semanas) y a partir de ellos va construyendo patrones de 50 semanas a base de primero conocer cuantos patrones de cada tipo le hacen falta y a continuación secuenciándolos de la mejor forma posible.

**RANDAHAWA, S. U. and SITOMPUL, D. (1993)**

Propone un sistema de soporte a la decisión que cubre todo el proceso, desde la generación de los patrones (multi-shift) más adecuados hasta la elección y asignación de estos patrones a las personas que deben realizarlos, en este caso enfermeras.

**HUNG, R. (1993)**

Da un método manual para resolver de forma óptima los patrones multi-turno con demanda variable. Donde se trabaja 3 días por semana y donde cada trabajador puede trabajar como máximo  $A$  fines de semana de cada  $B$ .

**HUNG, R. (1994-a) y HUNG, R. (1994-b)**

Da un método manual para resolver de forma óptima los patrones multi-turno con demanda variable. Donde se trabaja 3 ó 4 días por semana y 4 días por semana respectivamente.

**MASON, A.J. y otros (1998)**

Este trabajo empieza estimando por simulación el personal requerido en cada periodo (15 minutos) a continuación determina el número de trabajadores a tiempo completo y a tiempo parcial ( 3, 4 y 5 horas /día ) que hace falta y por último establece para los trabajadores a tiempo completo los patrones de días libres y el turno horario en el que trabajará cada uno de ellos.

**MASON, A.J. and SMITH, M.C. (1998)**

Utiliza técnicas de generación de columnas para ir creando los patrones más convenientes. Cada patrón indica los días libres y también el turno horario en el que se trabajará. Cada patrón se divide en segmentos (*workstretch*). A partir de los costes reducidos de las posibles columnas aún por generar, crea los segmentos más convenientes (segmentos de longitud igual a 1 día, 2 días, 3 días...) a continuación empalma estos segmentos de forma óptima atendiendo a las penalizaciones que puede tener la realización consecutiva de dos de ellos. En cada iteración del problema maestro restringido resuelve dos subproblemas: uno para calcular los segmentos más adecuados de cada longitud y otro para empalmarlos formando un patrón válido. Ambos subproblemas los plantea como un problema de camino mínimo en un grafo.

**MILLAR, H.H. and KIRAGU, M. (1998)**

Construye patrones cíclicos y no cíclicos para turnos de 12 horas de enfermería. Las 12 horas de trabajo pueden ser diurnas o nocturnas. El modelo en el que se basa es el del *camino más corto*.

**LAPORTE, G. (1999)**

Discute la polémica cuestión de que la elaboración de patrones está muchas veces sometida a toda una serie de rígidos y aritméticos principios que no permiten ver o examinar otras soluciones que podrían ser mas adecuadas.

En concreto pone ejemplos donde saltándose una sola vez en 5 semanas el requisito de tener un día libre siempre que se cambia de turno, las soluciones que se obtienen son globalmente mucho mejores que las que se obtendrían si se observara siempre dicho principio.

Propone otras acciones que usadas puntualmente pueden mejorar sensiblemente la calidad global de la solución, entre ellas

- Permitir alargar el tiempo o la duración de un turno horario.
- Sobreponer distintos juegos de patrones.
- Utilizar turnos de relevo.

### D3 Hierarchical - Day off

Este planteamiento trata de buscar patrones que además de distinguir los periodos de trabajo y descanso, determine la categoría del trabajo que hará la persona que se incorpora un día determinado al trabajo.

Se parte del supuesto de que existen distintas categorías de trabajos y trabajadores no igualmente cualificados todos ellos. Se parte también de la base de que a un trabajador se le permite hacer cualquier trabajo de su categoría o de categorías inferiores. En caso contrario bastaría dividir el problema en las diferentes categorías y no aportaría ningún beneficio resolverlo conjuntamente.

#### Datos de partida

- **Demanda en cada una de las categorías.**

$d$  vector fila de  $m$  componentes.

Donde:  $d_k$  es el numero de personas de categoría  $k$  que hace falta cada día.

- **Reglas que debe cumplir cualquier patrón valido.**

A modo de ejemplo:

- $A$  fines de semana libres de cada  $B$
- Como mínimo dos días de descanso a la semana
- Los días de trabajo consecutivos deben oscilar entre 2 y 4.

#### Ejemplos

##### **EMMONS, H. and BURNS, R.N. (1991)**

Calculan primero el número mínimo de trabajadores de cada categoría necesarios para cubrir toda la demanda y a continuación dan una heurística para elaborar efectivamente los patrones de días libres.

##### **HUNG, R. (1994-c)**

Este trabajo considera que la demanda de trabajadores en cada una de las categorías no es siempre la misma, estableciendo una demanda potencialmente distinta para cada día de la semana.

$d$  matriz de  $7 \times K$ .

Donde:  $d_{km}$  es el numero de personas de categoría  $k$  que hacen falta el día  $m$ .

Proporciona una heurística que nos da los patrones mas adecuados para cubrir toda la demanda al mínimo coste, entendiendo que cubrir una plaza de categoría  $k$  con un empleado de dicha categoría será más barato que hacerlo con uno de categoría superior.

## **3.2 Artículos clasificados**

### **BAKER, K.R. (1974-a)**

Scheduling a Full-Time Workforce to Meet Cyclic Staffing Requirements  
Management Science. Vol 20, N° 12, August 1974  
Pag: 1561 - 1568

### **BAKER, K.R. (1974-b)**

Scheduling Full-Time and Part-Time Staff to Meet Cyclic Requirements.  
Operational Research Quarterly Vol. 25 N°1, 1974  
Pag: 65 - 75

### **BAKER, K.R. (1976)**

Workforce Allocation in Cyclical Scheduling Problems: A Survey  
Operational Research Quarterly Vol 27 N°1 ii, 1976  
Pag: 155 - 167

### **BAKER, K.R. and MAGAZINE, M.J. (1977)**

Workforce Scheduling with Cyclic Demands and Day - Off Constraints  
Management Science vol.24 N° 2, 1977  
Pag: 161 - 167

### **BAKER, K.R.; BURNS, R.N. and CARTER, M.W. (1979)**

Staff Scheduling with Day - off and Workstretch Constraints  
AIIE Trans. 6, 11, 1979  
Pag: 286 - 292

### **BARTHOLDI, J.J. (1981)**

A Guaranteed Accuracy Round-off Algorithm for Cyclic Scheduling and Set Covering  
Operations Research 29, 3, 1981  
Pag: 501 - 510

### **BARTHOLDI, J.J.; ORLIN, J.B. and RATLIFF, H.D. (1980)**

Cyclic Scheduling via Integer Programs with Circular Ones  
Operations Research 28, 5, 1980  
Pag: 1074 - 1085

### **BARTHOLDI, J.J. and RATLIFF, H.D. (1978)**

Unnetworks with Applications to Idle Time Scheduling  
Management Science 24, 8, 1978  
Pag: 850 - 858

### **BEAUMONT, N. (1997)**

Using Mixed Integer Programming to Design Employee Rosters  
Journal of the Operational Research Society, 48, 1997  
Pag: 585 - 590

### **BECHTOLD, S.E. and SHOWALTER, M.J. (1987)**

A Methodology for Labor Scheduling in a Service Operating System  
Decision science 18, 1987  
Pag: 89 - 107

### **BECHTOLD, S.E. (1988)**

Implicit Optimal and Heuristic Labor Staffing in a Multi - Objective, Multilocation Environment  
Decision Science, 19, 2, 1988  
Pag: 353 - 373

- BECHTOLD, S.E.; BRUSCO, M. J. and SHOWALTER, M.J.(1991)**  
A Comparative Evaluation of Labor Tour Scheduling Methods  
Decision Sciences,22, 4, 1991  
Pag: 683 - 699
- BECHTOLD, S.E. and BRUSCO, M.J. (1994)**  
Working Set Generation Methods for Labor Tour Scheduling  
European Journal of Operational Research 74, 1994  
Pag: 540 - 551
- BECHTOLD, S.E. and BRUSCO, M.J. (1994-b)**  
A Microcomputer - Based Heuristic for Tour Scheduling of a Mixed Workforce  
Computers & Operations Research Vol.21 N° 9, 1994  
Pag: 1001 - 1009
- BIANCO, L. and BIELLI, M. (1992)**  
A Heuristic Procedure for the Crew Rostering Problem  
European Journal of Operational Research 58, 1992  
Pag: 272 – 283
- BRAILSFORD, S.C. ; POTTS, C.N. and SMITH, B.M. (1999)**  
Constraint Satisfaction Problems: Algorithms and Applications  
European Journal of Operational Research 119, 1999  
Pag: 557 - 581
- BRUSCO, M.J. y JACOBS, L.W. (1993)**  
A Simulated Annealing Approach to the Solution of Flexible Labour Scheduling  
Problems earch Society Vol. 44, N° 12, 1993  
Pag: 1191 - 1200
- BRUSCO, M.J. and JACOBS, L.W. (1998)**  
Eliminating redundant columns in Continuous Tour Scheduling Problems  
European Journal of Operational Research, 111, 1998  
Pag: 518-525
- BURNS, R.N. and CARTER, M.W. (1985)**  
Work Force Size and Schedules with Variable Demands  
Management Science 31, 1985  
Pag: 599 - 607
- CAPRARA, A.; FISCHETTI, M.; TOTH, P.; VIGO, D. and LUIGI GUIDA, P. (1997-a)**  
Algorithms for Railway Crew Management  
Mathematical Programming 79, 1997  
Pag: 125 - 141
- CAPRARA, A.; FISCHETTI, M.; GUIDA, P.L.; TOTH, P. and VIGO, D. (1997-b)**  
Solution of Large-Scale Railway Crew Planning Problems: the Italian Experience  
submitted for publication, 1997  
Pag:
- CAPRARA, A.; TOTH, P.; VIGO, D. and FISCHETTI, M. (1998-a)**  
Modeling and Solving the crew Rostering Problem  
Operations Research, 1998  
Pag: 820 - 830
- CAPRARA, A.; FOCACCI, F.; LEMMA, E.; MELLO, P.; MILANO, M.; TOTH, P. and VIGO, D. (1998-b)**  
Integrating Constraint Logic Programming and Operations Research Techniques for the Crew  
Rostering

Software-Practice and Experience, 28, 1998  
Pag: 49-76

**CARRARESI, P and GALLO, G. (1984)**

A Multi-Level Bottleneck Assignment Approach to the Bus Drivers' Rostering Problem  
European Journal of Operation Research 16, 1984  
Pag: 163 - 173

**DAY, P.R. and RYAN, D.M. (1997)**

Flight Attendant Rostering for Short-Haul Airline Operations  
Operations Research vol. 45, n.5, 1997  
Pag: 649 - 661

**EASTON, F.F. and ROSSIN, D.F. (1991)**

Equivalent Alternative Solutions for the Tour Scheduling Problem.  
Decision science 22, 1991  
Pag: 985 - 1007

**EMMONS, H. & BURNS R.N. (1991)**

Off-Day Scheduling with Hierarchical Worker Categories  
Operations Research v39, N° 3, 1991  
Pag: 484 - 495

**GOPALKRISHNAN, M. ; GOPALKRISHNAN, S. & MILLER, D.M. (1993)**

A Decision Support System for Scheduling Personnel in a Newspaper Publishing Environment  
Interfaces vol. 23, 1993  
Pag: 104 - 115

**HUNG, R. (1991)**

Single-Shift Workforce Scheduling under a Compressed Workweek  
Omega v19, 1991  
Pag: 494 - 497

**HUNG, R. (1993)**

A Three-Day Workweek Multiple-Shift Scheduling Model.  
Journal of the operation Research Society 44, 1993  
Pag: 141 - 146

**HUNG, R. (1994-a)**

A Multiple-Shift Workforce Scheduling Model Under the 4-Day Workweek With Weekday and Weekend  
Journal of the operation Research Society 45 N° 9, 1994  
Pag: 1088 - 1092

**HUNG, R. (1994-b)**

Multiple-Shift Workforce Scheduling Under the 3-4 Workweek With Different Weekday and Weekend Labor  
Management Science, Vol 40 N° 2, 1994  
Pag: 280 - 284

**HUNG, R. (1994-c)**

Single-Shift Off Day Scheduling of a Hierarchical Workforce with Variable Demands  
European Journal of Operational Research vol.78, 1994  
Pag: 49 - 57

**LAPORTE, G (1999)**

The art and science of designing Rotating Schedules  
Journal of the Operational Research Society, 50, 1999  
Pag: 1011-1017

- MABERT, V. and WATTS, C. (1982)**  
A Simulation Analysis of Tour-Shift Construction Procedures  
Management Science, 28, 5, 1982  
Pag: 520 - 532
- MASON, A.J.; RYAN, D.M. & PANTON, D.M. (1998)**  
Integrated Simulation, Heuristic and Optimisation Approaches to Staff Scheduling  
Operations Research Vol 46, Nº 2, 1998  
Pag: 161 - 174
- MASON, A.J. and SMITH, M.C. (1998)**  
Anested Column Generator for solving Rostering Problems with Integer Programming  
<http://www.esc.auckland.ac.nz/Mason>  
Pag: 1 - 8
- MILLAR, H.H. and KIRAGU, M. (1998)**  
Cyclic and Non-Cyclic Scheduling of 12h Shift Nurses by Network Programming  
European Journal of the Operational Research, 1998  
Pag: 582 - 592
- MILLER, H.E.; PIERSKALLA, W.P. and RATH, G.J. (1976)**  
Nurse Scheduling Using Mathematical Programming  
Operations Research, 24, 1976  
Pag: 857 - 869
- MORRIS, J.G.; SHOWALTER, M.J. (1983)**  
Simple Approaches to Shift, Days-Off and Tour Scheduling.  
Management Science, 29 (8), 1983  
Pag: 942 - 950
- PAIXÃO, J and PATO, M. (1989)**  
A Structural Lagrangian Relaxation for Two - Duty Period Bus Driver Scheduling Problems  
European Journal of Operation Research 39, 1989  
Pag: 213 - 222
- RANDHAWA, S.U. & SITOMPUL, D. (1993)**  
A Heuristic-Based Computerized Nurse Scheduling System  
Computers & Operations Research v20 Nº 8, 1993  
Pag: 837 - 844
- RYAN, D.M. (1992)**  
The Solution of Massive Generalized Set Partitioning Problems in Aircrew Rostering  
Operational Research Society, Vol. 43, Nº 5, 1992  
Pag: 459 - 467
- SEGAL, M. (1974)**  
The Operator Scheduling Problem: A Network Flow Approach  
Operations Research, 22 (4), 1974  
Pag: 808 - 823
- SHEPARDSON, F. and MARSEN, R.E. (1980)**  
A Lagrangean Relaxation Algorithm for the Two Duty Period Scheduling Problem  
Management Science 26, 1980  
Pag: 274 - 281
- THOMPSON, G. M. (1995)**  
Labor Scheduling Using NPV Estimates of the Marginal Benefit of Additional Labor Capacity  
Journal of Operations Management 13, 1995  
Pag: 67 - 86

**TOWNSEND, W. (1988)**

An Approach to Bus-Crew Roster Design in London Regional Transport  
Journal of the operation Research Society Vol. 39 N° 6, 1988  
Pag: 543 - 550

**VANCE, P.H.; BARNHART, C.; JOHNSON, E.L. and NEMHAUSER G.L. (1997)**

Airline Crew Scheduling: A new Formulation and Decomposition Algorithm  
Operations Research vol 45 n.2 (1997)  
Pag: 188 - 200

**VASCONCELOS, J. and CAMPOS, R. (1995)**

A Travelling Salesman Model for the Sequencing of Duties in Bus Crew Rotas  
Journal of the operation Research Society (1995) 46 ,1995  
Pag: 415 – 426

