EL PROBLEMA DE LA GENERACION AUTOMATICA DE HORARIOS DE SERVICIO DE TRIPULACIONES EN EL TRANSPORTE PUBLICO: UNA APLICACION PRACTICA Y UN ESTUDIO TEORICO.

RT 84/03

JAUME BARCELÓ

RESUMEN

El presente trabajo consta de dos partes, en la primera se presenta una panorámica de los métodos de solución del problema de la planificación automática de horarios de servicio en empresas de transporte público, en la segunda se describe la aplicación piloto desarrollada por nuestro departamento para las Empresas Transportes de Barcelona S.A. y Ferrocarril Metropolitano de Barcelona S.A., en ella se da cuenta del enfoque de modelización elegido, las alternativas algorítmicas implementadas, las mejoras aportadas a los algoritmos elegidos y los resultados computacionales obtenidos.

ABSTRACT

COMPUTER CREW SCHEDULING PROBLEMS IN MASS TRANSIT SYSTEMS

The present report consists of two parts. The first one is a short wurvey about the different approaches to the computer crew scheduling problems in mass transit systems, and the second part reports the pilot project developed by the Department of Operational Research for the public transportation Companies in Barcelona. Mathematical programming approach, algorithmic alternatives implemented, improvements to selected algorithms and computational results are reported.

Key words: Crew scheduling, set partitioning

EL PROBLEMA DE LA GENERACION AUTOMATICA DE HORARIOS DE SERVICIO DE TRIPULACIONES EN EL TRANSPORTE PUBLICO: UNA APLICACION PRACTICA Y UN ESTUDIO TEORICO.

XIV REUNION NACIONAL DE LA SEIOEI

J. Barceló
Departamento de Investigación Operativa y Estadística
Facultad de Informática
Universidad Politécnica
de Barcelona
(C/ Jordi Girona Salgado, 31,
Barcelona-34)

RESUMEN

El presente trabajo consta de dos partes, en la primera se presenta una panorámica de los métodos de solución del problema de la planificación automática de horarios de servicio en empresas de transporte público, en la segunda se describe la aplicación piloto desarrollada por nuestro departamento para las Empresas Transportes de Barcelona S.A. y Ferrocarril Metropolitano de Barcelona S.A., en ella se da cuenta del enfoque de modelización elegido, las alternativas algorítmicas implementadas, las mejoras aportadas a los algoritmos elegidos y los resultados computacionales obtenidos.

Clasificación AMS
Palabras Clave: Crew scheduling, set partitioning.

1. INTRODUCCION: PANORAMICA DE LOS METODOS DE PLANIFICACION DE HORARIOS DE SERVICIO DE LAS TRIPULACIONES EN EL TRANSPORTE PUBLICO.-

Los problemas de planificación de servicios de tripulaciones en empresas de transporte público, pueden considerarse como una variante de los problemas de itinerarios, con restricciones adicionales referidas a los instantes en que tienen que realizarse las diferentes actividades.

En la planificación de los servicios de las tripulaciones de los vehículos de una red de transporte público, el requerimiento primordial es el de secuenciar los movimientos de una tripulación en el espacio y en el tiempo de manera que se cumplan los movimientos deseados en el vehículo.

Aunque los problemas de planificación de servicios de tripulaciones son similares a los de planificación de los movimientos de los vehículos, en general han de incluir restricciones más complejas, como los requerimientos de descanso de las tripulaciones, condiciones de trabajo de los convenios aprobados por los sindicatos, etc...

Por ejemplo, consideremos el siguiente servicio (horario de trabajo) de un conductor de autobús:

7:00 AM: Fichar en el garage (principio de la jornada laboral)

7:03 AM: Desplazamiento desde el garage hasta el punto A de relevo de otro conductor.

7.11 AM: Sube al autobús número n en el punto A de relevo, releva al conductor y empieza la actividad de conducción del autobús número n.

11.20 AM: Abandona el autobús número n en el punto B de relevo y empieza una pausa.

11.52 AM: Se desplaza en otro autobús desde el punto B de relevo hasta el punto C de relevo.

12.20 PM: Sube al autobús número m en el punto C de relevo, releva al conductor y empieza la actividad de conducción del autobús número m.

3.30 PM: Conduce el autobús m hasta el garage, lo abandona allí y termina su servicio. (Fin de la jornada laboral).

Este ejemplo ilustra algunas de las complejidades contenidas en la planificación de servicios de tripulaciones. En particular cabe señalar la presencia de periodos de descanso, así como, a lo largo de la jornada laboral, de períodos durante los cuales el conductor conduce el vehículo, y períodos durante los cuales el conductor camina, descansa o se desplaza como un usuario más del sistema de transporte. Por otra parte, es importante tener en cuenta que el servicio de un conductor puede estar asociado diariamente con más de un vehículo.

En general el problema de la planificación de los servicios de los vehículos y el de la planificación de los servicios de las tripulaciones interaccionan: la especificación de los servicios de los vehículos establece ciertas restricciones sobre la planificación de los servicios de las tripulaciones (y viceversa). Por ello, aunque los modelos que incorporan ambos problemas en un único problema de optimización son en general bastante complejos, se tiende actualmente a definir sistemas en los que se intenta resolver los dos problemas simultáneamente.

Una de las opciones más corrientes en la práctica es la del procedimiento secuencial que resuelve un problema primero, (el de la planificación de los servicios de los vehículos), y después el otro, (planificación de los servicios de las tripulaciones), con algunos mecanismos para tener en cuenta la interacción entre ambos, (Hoffstadt, /5/, Bodin et al., /2/).

Una descripción inicial de la relación entre la planificación de los servicios de los vehículos y la de los servicios de las tripulaciones nos la proporciona el siguiente ejemplo: la figura 1 ilustra la planificación de los servicios de un conjunto de vehículos. Planificación que describe los movimientos requeridos por cierto sistema de transporte. Cada servicio individual contiene un conjunto de puntos en los que una tripulación puede relevar a otra. En un sistema de transporte público un punto de relevo es una parada, previamente designada, a lo largo de la línea recorrida por el vehículo. En la figura 1 se supone que los servicios de los vehículos tienen marcados los puntos de relevo.

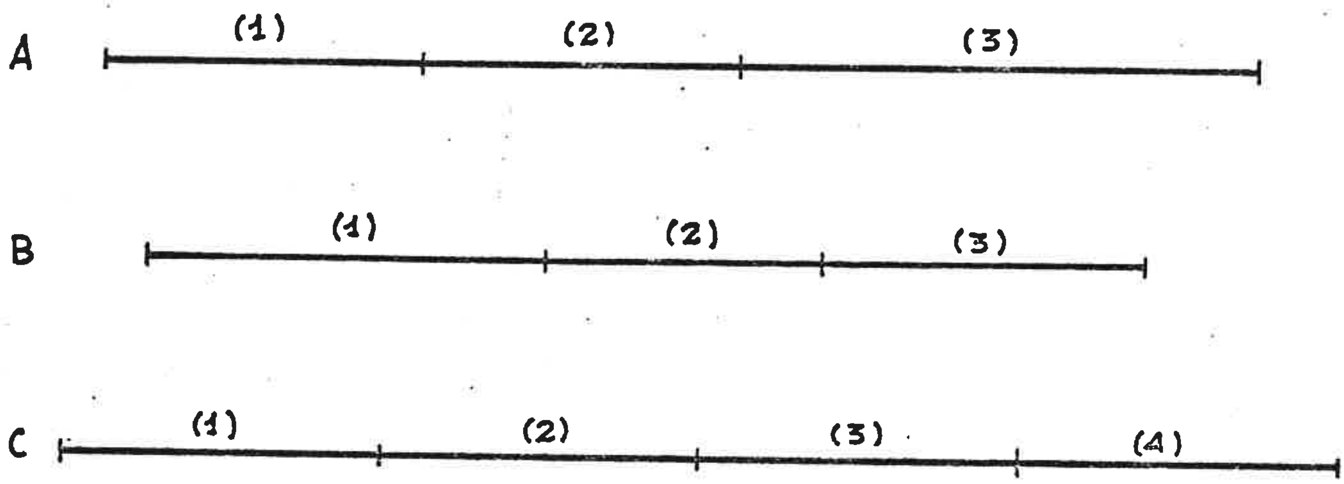

figura 1.

Para obtener una planificación de los servicios de las tripulaciones se descompone el servicio (trayecto) de cada vehículo en piezas (tramos) entre puntos de relevo. La planificación del servicio de una tripulación se obtiene agrupando varias piezas, teniendo en cuenta que la posibilidad de unir un tramo con otro depende entre otras cosas, del instante en que termina el primer tramo, (instante en que el vehículo pasa por el punto de relevo que define el final del primer tramo) con respecto al instante en que empieza el segundo tramo (instante en que el vehículo pasa por el punto de relevo que define el principio del segundo tramo); y de la localización del final del primer tramo con respecto a la localización del principio del segundo tramo. Aunque por simplicidad, en la mayor parte de las aplicaciones la función de este coste se considera lineal, no tiene porque serlo en general, ya que el coste total de un servicio no tiene por que ser la suma de los costes de sus componentes.

La figura 2 representa la planificación de servicios de las tripulaciones a partir de la planificación de servicios de los vehículos dada en la figura 1.

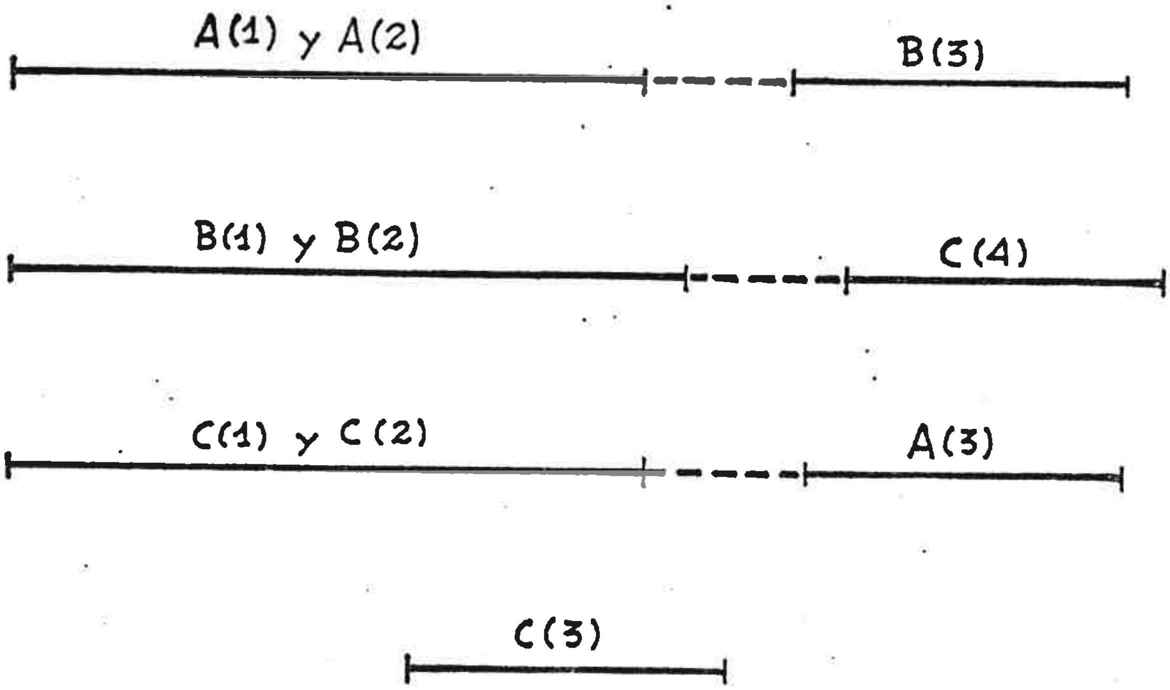

figura 2

(⊢ – – – ⊣ representa las interrupciones y descanso)

El enfoque representado en las dos figuras anteriores supone que los vehículos han sido planificados a priori e independientemente de las tripulaciones, que es el caso habitual.

El servicio de pasajeros en un sistema de transporte público se define mediante un conjunto de LINEAS, horarios y trayectos que especifican el servicio de la línea, de la manera siguiente:

HORARIO DE LA LINEA L

PRINCIPIO DE TRAYECTO (I)	PARADA A (II)	PARADA B	FIN DE TRAYECTO (III)
sentido →			
9:00	9:20	9:55	10:05
9:15	9:35	10:10	10:30
⋮	⋮	⋮	⋮
			← sentido
10:20	10:00	9:25	9:05
10:35	10:15	9:40	9:20
⋮	⋮	⋮	⋮

Cada fila de la tabla representa el viaje de un autobús y cada viaje queda caracterizado por:

- instante en que se inicia
- punto desde el que se inicia
- instante en que termina
- punto en que termina

Para definir la planificación de los servicios de las tripulaciones debemos considerar la naturaleza de sus movimientos, para lo cual hemos de definir los puntos de relevo, es decir las paradas a lo largo de la línea en los que una tripulación puede relevar a otra (que no necesariamente tienen porque coincidir con el principio y el final de la línea). Cada línea tendrá uno o más puntos de relevo. El período de trabajo (tramo elemental) de una tripulación en un vehículo empieza y termina en un punto de relevo o en un garage. A partir del horario de la línea y de la definición de los puntos de relevo podemos crear un conjunto de períodos de trabajo, o tramos elementales, particionando cada viaje según sus puntos de relevo. Así, para la línea del ejemplo anterior podríamos definir sus viajes de la manera siguiente:

VIAJES DE LA LINEA L

[(9:00,I), (10:15,III)], [(9:15,I), (10:30,III)], ...
[(9:05,III), (10:20,I)], [(9:20,III), (10:35,I)], ...

y sus períodos de trabajo o tramos elementales como:

[(9:00,I), (9:20,II)], [(9:20,II), (10:15,III)]
[(9:15,I), (9:35,II)], [(9:35,II), (10:15,III)]
[(9:05,III), (10:00,II)], [(10:00,II), (10:20,I)]
[(9:20,III, (10:15,II)] , [(10:15,II), (10:35,I)]

Por consiguiente, si un viaje representa la mínima cantidad de trabajo realizado por un vehículo, un tramo elemental representa la mínima cantidad de trabajo a realizar por un conductor. A partir de estos conceptos básicos podemos concebir la planificación del servicio de una tripulación de la manera siguiente: una tripulación (conductor), conduce un vehículo dado durante una porción de tiempo denominado tramo, (conjunto de tramos elementales), por lo que un servicio no es más que una secuenciación de tramos. En general habrá dos tipos de servicio para las tripulaciones, en función de las condiciones laborales. Servicios completos, son aquellos que constan de un único tramo o a lo sumo de dos tramos separados por un corto período de descanso, (por ejemplo de 30 ó 45 minutos como máximo). Servicios partidos son los formados por dos o más tramos separados por períodos de interrupción de mayor duración (por ejemplo más de 45 minutos pero menos de 4 horas, etc.). Ordinariamente las tripulaciones prefieren los servicios completos y en general resultarán los más baratos, por lo que se tenderá a cubrir la planificación con el máximo número de servicios completos posible, así, por ejemplo una típica restricción adicional en la modelización del problema puede ser la de que el número total de servicios completos sea superior a una cantidad dada. Los servicios partidos, necesarios para poder cubrir los requerimientos de servicio, están gravados, en general, por un coste adicional definido por las condiciones laborales en función del número de interrupciones y la duración de las mismas.

En estos términos el proceso de generación automática de servicios puede concebirse como un proceso constituido por las cuatro etapas siguientes:

1) Descomposición de los viajes en tramos elementales
2) Agrupación de los tramos elementales en tramos separados por puntos de relevo
3) Generación de servicios posibles por secuenciación de uno o más tramos
4) Selección de un subconjunto de servicios posibles que satisfaga las necesidades totales de servicio.

Las tres primeras etapas suelen agruparse en un proceso heurístico de generación de un gran número de servicios posibles, de manera que las secuenciaciones sean coherentes con las reglamentaciones laborales, que establecen condiciones de operación tales como:

. Máxima longitud de un tramo sin interrupción de descanso
. Máxima y mínima duración de un descanso
. Porcentaje de servicios completos
. Etc.

Tradicionalmente estos enfoques heurísticos se han agrupado en dos tendencias, la de intentar minimizar directamente el número de servicios a generar, reproduciendo las heurísticas de implementación manual, de manera que la mayor parte de los servicios generales ya sean aceptables, y la de generar simplemente servicios compatibles con la reglamentación laboral establecida.

Un ejemplo de heurística del primer tipo es el siguiente (Parker y Smith, /9/):

a) Determinar para cada vehículo que sale del depósito antes del máximo de las mañanas, el último momento en que el conductor que sacó el vehículo del depósito puede ser relevado. Para ello
   a1) Explorar los tiempos de paso por los puntos de relevo para cada vehículo, empezando por el primero, hasta que se encuentre uno que, de acuerdo con los parámetros de las condiciones de trabajo, excede la longitud máxima del tiempo sin descanso para un servicio completo.
   a2) Tomar el punto de relevo identificado y verificar si el tramo determinado puede ser parte de un servicio. En caso afirmativo marcarlo.
   En caso contrario retroceder hasta el punto de relevo inmediatamente anterior y repetir el proceso.
b) Al final de la repetición de (a1) y (a2) hemos obtenido un conjunto de puntos de relevo marcados, que indican la obligación de tomar o dejar vehículo por una tripulación.
c) Ordenar cronológicamente, en orden ascendente, el conjunto de puntos marcados.
d) Repetir los procesos (a) y (b) en sentido inverso para los vehículos que regresan al depósito al final de la jornada.
e) Ordenar los nuevos puntos marcados en orden descendente.
f) Explorar el primer conjunto de puntos marcados. Ir eligiendo secuencialmente los tramos formados y determinar los tramos en los que el elegido podría completar un servicio. Marcar los nuevos puntos y repetir el proceso con el segundo conjunto de puntos marcados.
g) Al final del proceso (f) han quedado definidos conjuntos de servicios completos de principio y final de jornada respectivamente. Los tramos que no han quedado marcados son los candidatos a formar servicios partidos. Apareados para formar servicios partidos.

La figura 3 ilustra un ejemplo de la heurística propuesta en el caso de un conjunto de autobuses para los que las condiciones de operación fuesen las siguientes: el máximo tiempo permitido de conducción sin descanso es de 4 1/2 horas, ha de haber dos servicios partidos y el mínimo tiempo de interrupción es de 45 minutos.

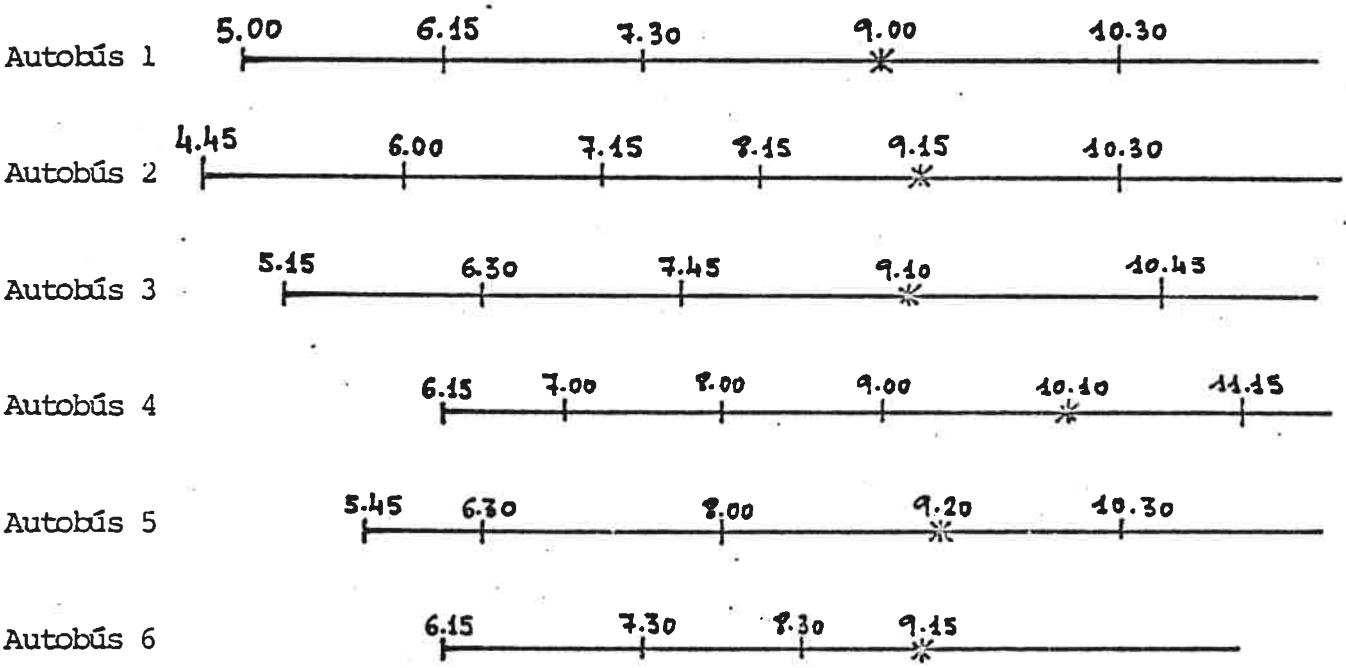

Figura 3

Los puntos señalados con una x identifican los puntos de relevo determinados en el paso (a1), y son para cada autobús el último instante en que puede abandonarlo el conductor que inició el servicio sin violar la restricción de no conducir sin descanso más de 4 1/2 horas.

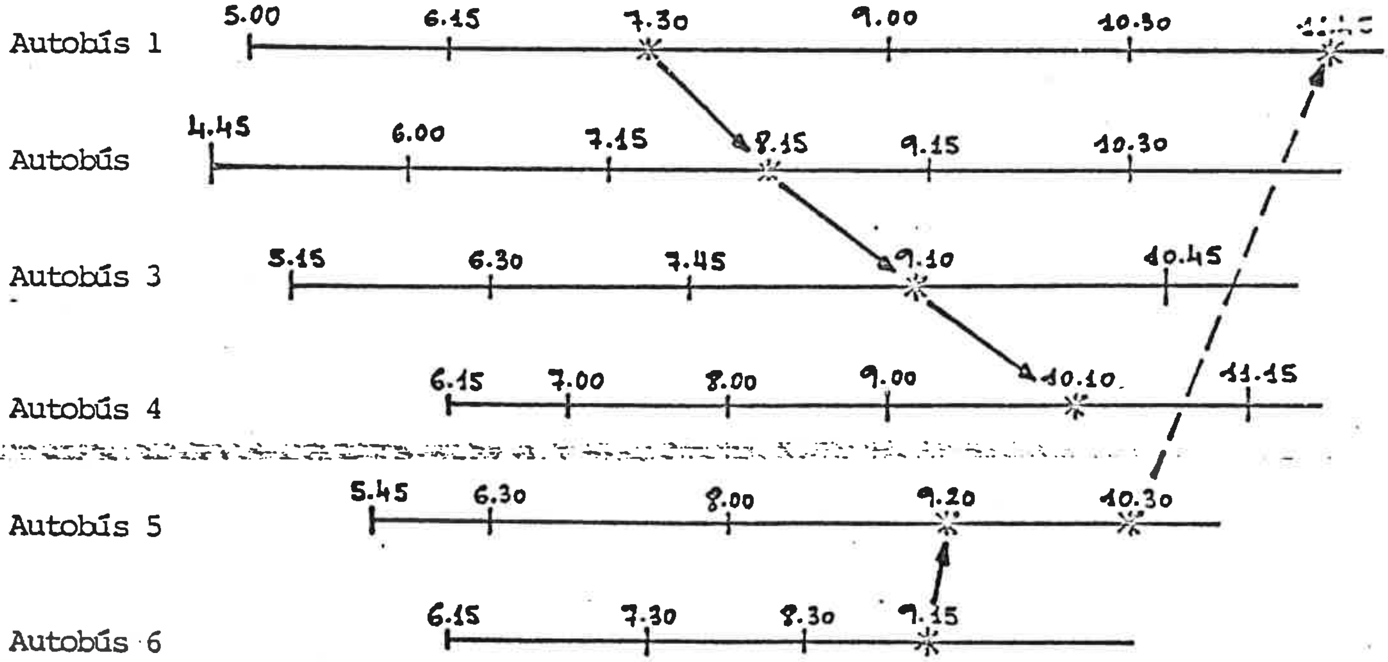

Figura 4

La figura 4 representa gráficamente el resultado del paso (a2) en el autobús 1, el punto 9:00, previamente marcado, ha sido retrasado hasta el 7.30, puesto que al no haber en los restantes autobuses un punto marcado a una distancia aproximada de 45 minutos, el tramo no podía ser parte de un servicio. El nuevo tramo del autobús 1, que empieza a las 5.00 y termina a las 7.30 puede enlazarse únicamente con el punto de relevo 8.15 del autobús 2, lo cual obliga a retrasar en el autobús 2 el punto marcado 9.15 al 8.15. En el autobús 1 podemos marcar un nuevo punto a las 11.45 que define un segundo tramo que podrá pasar a completar un servicio o iniciar un servicio partido, este nuevo punto marcado representa el último instante en que una tripulación que subió al autobús a las 7.30 puede abandonarlo.

El punto 8.15 del autobús 2 puede enlazarse con el 9.10 del autobús 3, este punto es preferible al 9.00 del autobús 4 porque aquel está marcado y este no. El punto marcado 9.10 del autobús 3 puede enlazarse con el punto marcado 10.10 en el autobús 4, por lo tanto estos puntos marcados se conservan.

El punto marcado 9.15 del autobús 4 se puede enlazar con el punto marcado 9.20 del autobús 5, sin ninguna interrupción en el servicio, y entonces se marca un nuevo punto a las 10.30 en el autobús 5, que podría enlazarse con el 11.45 del autobús 1 para un servicio partido.

Al final del proceso tendríamos definidos los siguientes servicios:

SERVICIO 1: Empieza en el autobús 1 hasta las 7.30, descansa 45 minutos y continúa en el autobús 2 a partir de las 8.15 hasta las 12.45.

SERVICIO 2: Empieza en el autobús 2 hasta las 8.15, descansa hasta las 9.10 y continúa en el autobús 3 hasta las 12.30.

SERVICIO 3: Empieza en el autobús 3 hasta las 9.10, descansa hasta las 10.10 y continúa en el autobús 4 hasta las 14.00.

SERVICIO 4: Empieza en el autobús 6 hasta las 9.15 para el autobús 5 sin interrupción a las 9.20 y continúa en él hasta las 10.30, descansa hasta las 11.45 y continúa en el autobús 1 desde las 11.45 hasta las 15.00.

SERVICIO 5: Empieza en el autobús 5 hasta las 9.20, descansa hasta las 10.30 y continúa en el mismo autobús 5 hasta las 14.15.

Al final del proceso han quedado formados 5 servicios completos de principio de jornada, y han quedado dos tramos, al comprendido entre el punto 7.30 y el 11.45 del autobús 1 y el comprendido entre las 6.15 y las 10.10 del autobús 4, que formarán parte de dos servicios partidos.

La otra tendencia que hemos mencionado antes consiste en generar servicios factibles que recubran suficientemente el trabajo total a realizar en los autobuses, y utilizar procedimientos de programación matemática que elijan de entre los servicios generados los que satisfacen los requerimientos con el mínimo coste. (Parker y Smith, /9/, Ryan y Foster, /11/).

Este planteamiento tiene el riesgo de generar un número de servicios tan grande que sea imposible de tratar, ya que el número de combinaciones posibles puede ser enorme. Para mantener el problema dentro de unas dimensiones manejables se suele imponer restricciones adicionales al proceso de generación de servicios posibles. Generalmente tales restricciones proceden del análisis y observación de las heurísticas de implementación manual en las empresas de transporte público. Ejemplos de tales reglas son los siguientes:

- . Limitar a dos o tres el número de tramos que constituyen un servicio.
- . Dar una longitud fija a los diferentes tipos de tramo (primer tramo de un servicio de principio de jornada, etc.)
- . Poner restricciones a los tiempos en que pueden comenzar o terminar los distintos tipos de servicio (un servicio completo de principio de jornada no puede empezar después de las 8.00 y ha de terminar antes de las 15.00, etc.)
- . Fijar las longitudes de los descansos.
- . Intentar identificar aquellos puntos de relevo más adecuados para generar servicios que conduzcan a una buena planificación. El procedimiento básico consiste en marcar aquellos puntos de relevo en cada uno de los autobuses separados entre sí por tramos de longitud máxima. Así por ejemplo, para un vehículo dado se supone que la tripulación que lo saca del depósito va a estar en él el máximo tiempo permitido por las condiciones laborales, se busca el punto de relevo más próximo a dicho instante y se marca; se repite el proceso con las siguientes tripulaciones hasta que el vehículo regresa al depósito. Si solo se autoriza a las tripulaciones a relevarse en los puntos marcados, entonces el servicio del autobús será cubierto por el mínimo número de tramos.
  Análogamente se marcan los puntos de relevo en sentido contrario, es decir, partiendo del momento en que el vehículo regresa al depósito, hasta el momento en que sale de él.
  En general los puntos marcados "progresivamente" y los marcados "regresivamente" no coincidirán sino que definirán un intervalo de tiempo en que se deberán efectuar los relevos de tripulaciones para cubrir el horario del vehículo con el número mínimo de tramos posible.

La figura adjunta (Fig. 5) presenta el caso de un autobús que sale del depósito a las 7.00 y regresa a las 21.00, pasando por un punto de relevo cada hora, para el que el tramo de máxima longitud es de 4 1/2 horas:

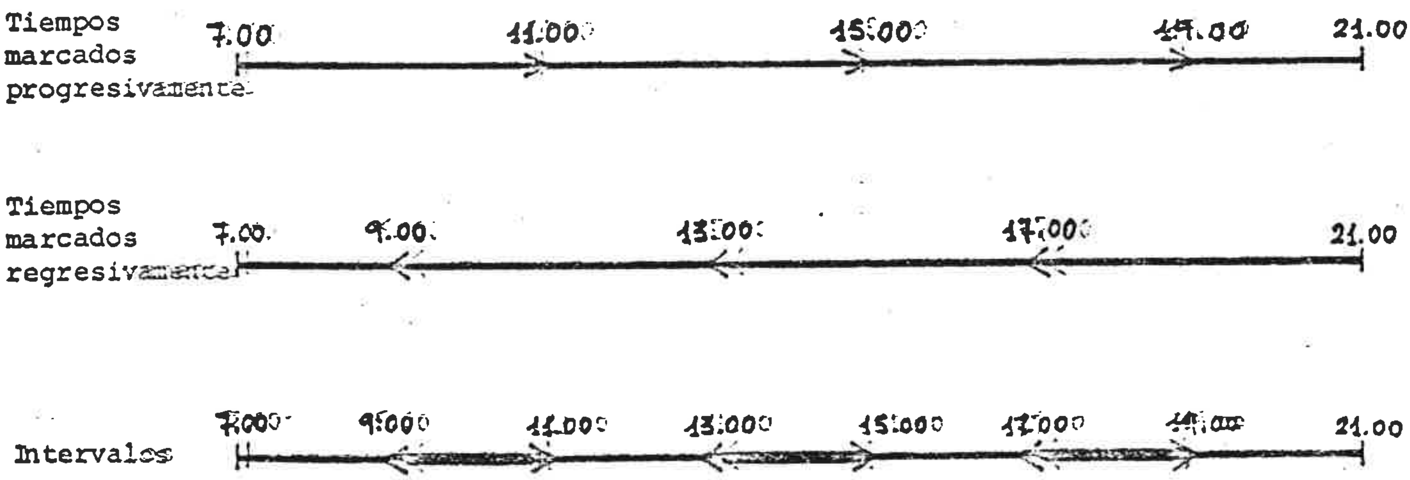

Figura 5

Admitiendo entonces únicamente relevos en los puntos de relevo dentro de los intervalos marcados, se inicia un proceso de exploración similar al descrito anteriormente, para ir enlazando los diferentes tramos de manera que configuren servicios factibles.

Una vez generado un número suficiente de servicios posibles, la cuarta y última etapa del proceso de generación automática de servicios descrito anteriormente, consiste en la selección de los servicios que satisfacen las necesidades de la planificación.

Generalmente tal selección se realiza por procedimientos de programación matemática que determinen el conjunto de servicios óptimo, según el criterio de optimización elegido, que habitualmente es el de la minimización del coste del servicio.

Los procedimientos de programación matemática se agrupan en dos categorías:

a) Procedimientos de análisis de redes (Bodin et al., /2/, Lessard et al. /6/).
b) Métodos de partición y/o recubrimiento (Pierce y Lasky, /10/ Mitra y Welsch, /8/, Ryan y Foster, /11/).

a) <u>Procedimientos de análisis de redes</u>.

Plantean el problema del enlace de los tramos elementales y de los tramos para formar servicio, como una variable de problemas de búsqueda de flujos posibles en grafos. Un esquema de este enfoque sería el siguiente, se asocia al problema de generación de servicios un grajo cuyo conjunto N de nodos está definido por el conjunto de tramos elementales, de manera que cada nodo representa un tramo elemental, y cuyo conjunto arcos A está particionado en dos subconjuntos:

A1 = conjunto de arcos "progresivos", un arco $(i,j)\in A1$ si una tripulación puede cubrir de forma continuada los tramos elementales i y j es decir si los tramos elementales i y j pueden formar parte de un tramo, primero el i y después el j. A cada arco (i,j) se le asocia una variable binaria x_{ij}, tal que $x_{ij}=1$ si el arco (i,j) entra en la solución y $x_{ij}=0$ en caso contrario.

A2 = Conjunto de arcos "regresivos", el arco $(j,i)\in A2$ si un tramo que empieza en i y termina en j es factible, a cada arco (j,i) se le asocia una variable binaria y_{ij} que vale 1 si el arco (j,i) interviene en la solución y 0 en caso contrario.

La formulación se completa describiendo las formas en que los tramos pueden combinarse para formar servicios. Denotaremos por ρ el conjunto de todos los servicios potenciales (secuencias de tramos factibles) y por $\rho(i,j)$ el conjunto de los servicios que incluyen un tramo que empieza en i y termina en j, y asociaremos un coste c_l con cada secuencia de tramos $l\in\rho$ y un coste c_{ij} con cada arco $(i,j)\in A = A_1\cup A_2$, con lo cual el problema de flujo en grafo que busca circulaciones posibles, cuyas soluciones determinan los servicios elejidos, es:

$$[\mathrm{MIN}] \sum_{l\in\rho} c_l z_l + \sum_{(i,j)\in A_1} c_{ij} x_{ij} + \sum_{(i,j)\in A_2} c_{ij} y_{ij} \qquad (1)$$

sometida a:

$$\sum_{i:(i,j)\in A_1} x_{ij} + \sum_{i:(i,j)\in A_2} y_{ij} - \sum_{i:(j,i)\in A_1} x_{j,i} - \sum_{i:(j,i)\in A_2} y_{ji} = 0 \qquad \forall j\in N \quad (3)$$

$$\sum_{i:(i,j)\in A_1} z_{ij} + \sum_{i:(i,j)\in A_2} y_{ij} = 1, \qquad \forall j\in N \quad (3)$$

$$\sum_{(i,j)\in A_1\cap C} x_{ij} \quad \sum_{i:(i,j)\in A_2\cap C} y_{ij} \leq |C| - 1, \forall \text{ ciclo} C: |A_2\cap C| \geq 2 \quad (4)$$

$$\sum_{l\in p(i,j)} z_l - y_{ji} = 0 \quad . \forall (j,i)\in A_2 \qquad (5)$$

$$0 \leq x_{ij} \leq 1 \quad y \quad \text{entero}, \ \forall (i,j)\ A_1 \qquad (6)$$

$$0 \leq y_{ij} \leq 1 \quad y \quad \text{entero}, \ \forall (i,j)\ A_2 \qquad (7)$$

$$0 \leq z_l \leq 1 \quad y \quad \text{entero}, \ \forall l \in \rho \qquad (8)$$

donde las restricciones (5) y (8) aseguran que todo tramo elemental queda recubierto por un servicio, evidentemente $z_1 = 1$ si el servicio l se incluye en la xolución $z_1 = 0$ en caso contrario.

b) <u>Métodos de partición y/o recubrimiento</u>

Sea I el conjunto de tramos elementales y J el conjunto de servicios factibles generados, a partir de los conjuntos I y J podemos construir la matriz A, de incidencia, de dimensiones $|I| \times |J|$ cuyos elementos a_{ij} se definen de la manera siguiente:

$$a_{ij} = \begin{cases} 1 \text{ si el tramo elemental i forma parte del servicio j} \\ \\ 0 \text{ En caso contrario} \end{cases}$$

Cada fila de la matriz A corresponde a un tramo elemental y los elementos no nulos de la misma identifican los servicios en que interviene dicho tramo elemental, y cada columna se corresponde con un servicio factible y sus elementos no nulos identifican los tramos elementales que componen el servicio en cuestión. A cada columna (servicio) le podemos asociar una variable de decisión x_j cuyos valores son:

$$x_i = \begin{cases} 1 & \text{si el servicio j forma parte de la solución} \\ 0 & \text{En caso contrario} \end{cases}$$

Entonces si c_j es el coste del j-ésimo servicio el problema de elegir un conjunto de servicios, de coste mínimo, que satisfaga los requerimientos de la planificación se puede formular como el problema de programación matemática

$$[\text{MIN}] \sum_{j \in J} c_j \, x_j \tag{9}$$

sometido a

$$\sum_{j \in J} a_{ij} \, x_j = (\geqslant) \; 1 \quad \forall \, i \in I \tag{10}$$

$$x_j \in \{0,1\}, \; \forall \, j \in J \tag{11}$$

El conjunto de restricciones (10) impone la condición de que todo tramo elemental forme parte de algún servicio y no queden tramos elementales sin recubrir. La formulación de las restricciones (10) como restricciones de igualdad estricta, corresponde a la formulación como problema de PARTICION DE CONJUNTOS, en la cual cada tramo elemental interviene únicamente en la formación de un único servicio y por lo tanto no se solapan dos tripulaciones en un mismo tramo. Si por el contrario las restricciones (10) se formulan como restricciones de desigualdad $\geqslant$, el problema queda planteado como un problema de RECUBRIMIENTO DE CONJUNTOS, en el cual se admite la posibilidad de que un tramo elemental intervenga en la formación de más de un servicio solapando, por lo tanto, las tripulaciones durante el período de tiempo correspondiente.

La formulación anterior, tanto en el caso de partición como en el de recubrimiento puede completarse con restricciones adicionales que representen condiciones complementarias de operación, como por ejemplo restricciones del tipo

$$\sum_{j \in J_k} x_j \leqslant n_k \quad , \quad k \in K$$

donde K es el conjunto de los diferentes tipos de servicio, n_k es el número máximo de servicios de tipo k admitidos en la planificación y $J_k \subset J$, es el conjunto de servicios factibles de tipo k generados.

2. ESTUDIO DEL CASO DE LA EMPRESA FERROCARRIL METROPOLITANO DE BARCELONA S.A.

En el mes de Mayo de 1983 las Empresas Transportes de Barcelona S.A. (Autobuses) y Ferrocarril Metropolitano de Barcelona S.A. (Metro), firmaron un convenio con el Departamento de Investigación Operativa de la Facultad de Informática de la Universidad Politécnica de Barcelona para proceder a un estudio piloto sobre la generación automática de horarios de servicio de conductores en una de las líneas del ferrocarril metropolitano, estudio piloto cuyos resultados serían extendidos posteriormente a las restantes líneas de metro y autobuses.

Después de una etapa de documentación se procedió a elegir una línea piloto por sus características de representatividad y complejidad, y un enfoque para plantear el problema.

El enfoque elegido fue el siguiente:

a) Generación de servicios factibles mediante un procedimiento heurístico. El procedimiento heurístico elegido fue una versión ad hoc de la segunda variante descrita en el apartado anterior, consistente en fijar unas longitudes de tramo en función de las condiciones laborales estipuladas por el convenio colectivo correspondiente y, a partir de ellas definir intervalos de relevo a partir del doble proceso de marcaje progresivo y regresivo de puntos de relevo. Los puntos de relevo incluidos en cada intervalo obtenido señalaban los extremos de los tramos a considerar, a partir de los cuales se inicia un proceso de exploración secuencial que enlaza cada tramo con los restantes tramos con los que puede configurar un servicio factible según la reglamentación de trabajo.
Para la línea piloto elegida, (la número 1), con 16 trenes en servicio y un paso por punto de relevo cada hora aproximadamente, los resultados obtenidos fueron de 460 tramos elementales y 6065 servicios totales, entre servicios completos y servicios partidos.

b) Selección de un subconjunto de servicios factibles de coste mínimo, mediante un procedimiento de partición.
En la primera versión implementada, cuyos resultados se reportan en el presente informe, el modelo elegido fue uno de partición de conjuntos correspondiente al definido por (9), (10) y (11), con las restricciones (10) formuladas como restricciones de estricta igualdad. En función de los resultados de la heurística (a), las dimensiones del problema tipo de partición de conjuntos fueron de 460 x 6065, (460 filas y 6065 columnas).

2.1 ALGORITMO DE PARTICION DE CONJUNTOS

Desde un principio decidimos desarrollar dos líneas paralelas de trabajo, siguiendo el planteamiento de Balas y Padberg, /1/, una de tipo enumerativo y otra de tipo branch and bound incluyendo procedimientos de acotación condicional de corte disyuntivo y métodos subgradiente. La primera línea, de tipo enumerativo, perseguía el objetivo de disponer lo más rápidamente posible de una herramienta práctica, q permitiese obtener resultados, a partir de los cuales evaluar y mejorar los procedimientos heurísticos de generación de servicios posibles al tiempo que se ponía a disposición de las Empresas contratantes un software operativo. La segunda línea persigue unos objetivos de investigación fundamental y profundización en el diseño de algoritmos para particionamiento incluyendo restricciones adicionales.

Los resultados computacionales a que hace referencia este informe corresponden a la primera línea, estando la segunda todavía en curso de desarrollo.

El algoritmo implementado consta de las siguientes fases:

1) Preprocesado de la matriz A de incidencia (Balas y Padberg /1/, Garfinkel y Nemhauser, /3/, Marsten, /7/, etc.)
2) Implementación de un esquema enumerativo básico de explotación de la estructura especial de la matriz A, (Pierce y Lasky, /10/, Garfinkel y Nemhauser, /3/, Marsten, /7/)
3) Mejoras del proceso de exploración y acotación
   3.1) Modificaciones del vector de costes (Pierce y Lasky, /10/, Gondran y Laurière, /4/)
   3.2) Utilización de restricciones de sustitución, (Pierce y Lasky, /10/)
   3.3) Explotación adicional de la estructura de la matriz A.

2.1.1 Preprocesado de los datos del problema

Los resultados referenciados en la bibliografía citada aconsejan el preprocesado previo de los datos del problema, antes de aplicar ningún tipo de algoritmo, por la reducción sustancial de las dimensiones del mismo que puede proporcionar. El preprocesado típico de la matriz 0-1, de incidencia, para un problema de particionamiento consta de las siguientes reglas de reducción:

Denotando por I el conjunto de filas y por J el de columnas, por a^i la i-ésima fila y por a_j la j-ésima columna de A siendo $|I| = m$ y $|J| = n$.

1) Si para algún $i \in I$ y $k \in J$: $a_{ik} = 1$, $a_{ij} = 0$, $\forall j \in J-\{k\}$, entonces:
   1.a Hacer $x_k = 1$ y eliminar la columna k
   1.b Suprimir todas las columnas $j \in J-\{k\}$ para las que $a_k \cdot a_j \geq 1$
   1.c Suprimir todas las filas $h \in I$ tales que $a_{hk} = 1$
2) Si para algún $i,k \in I$, $a^i \leq a^k$, entonces se puede
   2.a) Suprimir la fila k
   2.b) Suprimir todas las columnas $j \in J$ tales que $a_{ij}=0$, $a_{kj}=1$,
3) Si para algún $k \in J$ y algún subconjunto $J' \subset J$

$$\sum_{j \in J'} a_j = a_k \quad y \quad \sum_{j \in J'} c_j \leq c_k$$

   se puede suprimir la columna k
4) Para cada $i \in I$, sea $J_i = \{j \in J \mid a_{ij} = 1\}$
   Entonces cualquier columna a_k tal que $a_k\, a_j \geq 1$, $\forall j \in J_i$, para algún $i \in I$, puede eliminarse.

2.1.2 Esquema enumerativo básico

En esencia el proceso implementado ha sido, siguiendo las referencias mencionadas, el de explotar la estructura de la matriz A particionándola en t bloques no vacíos, B_j, $j=1,\ldots,t$, tales que el bloque B_j satisface la condición $a_{ik}=1$, $k \in B_j$ y $a_{ik}=0$ para

$$k \in \bigcup_{l=j+1}^{t} B_l$$

para alguna fila i de la matriz A. Las filas de la matriz A se reordenan de manera que la fila que define el bloque B_j se convierta en la fila j, para $j=1,\ldots,t$. si $t<m$, se definen los bloques va-

cíos B_j para $j=t+1,...,m$. Dentro de cada bloque se ordenan las columnas por medio de algún criterio heurístico, en la versión básica el criterio empleado ha sido el de Garfinkel y Nemhauser, /3/. Las mejoras referidas en el punto 2.1.3 proceden, en parte, de la utilización de otros criterios que serán discutidos en dicho apartado.

Denotando por S el conjunto de índices de una solución parcial por S^+ el subconjunto de variables fijadas a 1, por z(S) el valor de la función objetivo para dicha solución parcial y por R(S) el conjunto de filas satisfechas por S y por Z la solución incumbente (mejor solución posible obtenida hasta el momento, si se ha obtenido alguna), el algoritmo enumerativo básico procede de la manera siguiente:

Paso 1 (Inicialización)
$S = \emptyset$, $R(S) = \emptyset$, $\bar{z} = \infty$ y $z(S) = 0$
Ir al Paso 2.

Paso 2 (Selección del bloque siguiente)
Hacer $r = \min \{i \mid i \notin R(S)\}$
Marcar el principio del bloque r (elemento de mínimo coste)
Ir al Paso 3.

Paso 3 (Test para aumentar una variable)
Empezando por la posición marcada en el bloque r, examinar las columnas de A en el bloque r.
Si se encuentra una columna j tal que $a_{ij}=0$, $\forall i \in R(S)$, y $z(S)+c_j<\bar{z}$,
Ir al Paso 4. Si se encuentra una columna j para la que $z(S)+c_j \geq \bar{z}$, o si se ha terminado de explorar el bloque r, ir al Paso 5.

Paso 4 (Test de una nueva solución)
Redefinir S^+: $S^+ \leftarrow S^+ \cup \{j\}$, $R(S)$:$R(S) \leftarrow R(S) \cup \{i \mid a_{ij}=1\}$
y $z(S)$:$z(S) \leftarrow z(S)+c_j$. Si $R(S)=\{1,2,...,m\}$ se ha encontrado una nueva solución posible mejor, registrarla como incumbente y actualizar $\bar{z}$,
ir al Paso 5.
En caso contrario ir al Paso 2.

Paso 5 (Retroceso)
Si $S^+ = \emptyset$ (se ha examinado todo el bloque 1), fin del algoritmo.
La mejor solución encontrada hasta el momento (si se ha encontrado alguna) es la óptima.
En caso contrario, si k es el último índice incluido en S^+
Redefinir S^+: $S^+ \leftarrow S^+ - \{k\}$.
Sea B_r el bloque al que pertenece la columna a_k. Poner una marca en la columna siguiente del bloque r, eliminar la marca previa del bloque r e ir al Paso 3.

2.1.3 Mejoras del proceso de exploración y acotación

El esquema enumerativo básico descrito en el apartado anterior puede mejorarse introduciendo diferentes modificaciones en el proceso entre las variantes a considerar en nuestro trabajo hemos tenido en cuenta únicamente tres modalidades: modificación del vector de costes, y por lo tanto actuación sobre el orden de las columnas en los bloques que estructuran la matriz A, mejora del proceso de exploración a base de calcular cotas inferiores del coste de complementación de cada una de las soluciones parciales mediante la utilización de restricciones de sustitución, y finalmente un procedimiento original de explotación adicional de la estructura de la matriz A.

2.1.3.1. Modificación del vector de costes

Pierce y Lasky, /10/ y Delorme y Heugron, /12/, entre otros, reportan importantes reducciones del tiempo de exploración cuando se sustituye el vector de costes c de la función objetivo, por el vector de costes reducidos $c - c_B B^{-1} A$, de la tabla óptima del simplex correspondiente a la relajación lineal ordinaria del problema de partición, obtenido sustituyendo las restricciones $x_j \in \{0,1\}$ $\forall j \in J$, por $0 \leq x_{ij} \leq 1$; $\forall j \in J$.

Esto es posible puesto que $(c - c_B B^{-1} A)\, x = cx - c_B B^{-1} e$, pues $Ax=e$, siendo e un vector cuyas componentes son todas iguales a la unidad, para todo vector x solución posible y por lo tanto el valor de la función objetivo modificada por los costes reducidos difiere del valor de la función objetivo original sólo en una constante para toda solución posible.

Con esta sustitución las variables se ordenarían dentro de los bloques B_j de la matriz A según el orden no decreciente de sus costes reducidos, ordenación que proporciona una función de elección en el proceso enumerativo, más fina que la de los costes reales, que limita de forma notable la profundidad de la exploración en el árbol enumerativo.

Esto es así porque en la medida en que, en esta categoría de problemas, el óptimo continuo y el óptimo entero están muy próximos y la evaluación de una solución parcial utilizando los costes reducidos iguala muy rápidamente el coste de la última solución obtenida.

El problema de cálculo más grave que plantea esta modificación es el de resolver el programa lineal contínuo asociado al problema del programa lineal a resolver y, lo que es peor, en la tendencia a la inestabilidad numérica y a la degeneración debido a la baja densidad de la matriz A, típica de estos problemas de partición (Marsten, /7/, Delorme y Hengon, /12/).

Afortunadamente esta desventaja puede paliarse, en parte, utilizando una versión ad hoc del algoritmo del simplex dual con variables acotadas. Para ello se puede reformular el problema de la manera siguiente: las restricciones

$$\sum_{j \in J} a_{ij} \, x_j = 1, \; \forall i \in I$$

$$0 \leq x_j \leq 1 \quad , \forall j \in J$$

de la relación lineal ordinaria del problema de partición, pueden sustituirse por el conjunto de restricciones equivalente:

$$\sum_{j \in J} a_{ij} \, x_j \leq 1 \quad , \forall i \in I \tag{13}$$

$$\sum_{j \in J} n_j \, x_j \geq m \tag{14}$$

$$0 \leq x_j \leq 1 \tag{15}$$

donde

$$n_j = \sum a_{ij} \quad , \forall j \in J \quad \text{y} \quad m = |I|$$

es el número de elementos no nulos de la columna j de la matriz A. Introduciendo variables de holgura en las restricciones (13) y (14) queda:

$$\sum_{j \in J} a_{ij}\, x_i + x_{n+i} = 1 \quad , \forall i \in I \tag{16}$$

$$\sum_{j \in J} - n_j\, x_j + x_{n+m+1} = -m \tag{17}$$

de donde se deduce que:

$$x_{n+i} = 1 \quad , \forall i \in I$$

y

$$x_{n+m+1} = -m$$

es una solución básica posible dual, no posible primal que sirve de solución inicial para el algoritmo del simplex dual, que presenta menos problemas de degeneración en este tipo de problemas.

Una alternativa más sencilla y rápida de mejorar la estructura de costes sin tener que resolver un programa lineal, es la establecida por Gondran y Lauriere, /4/, consiste en elegir una fila cualquiera a^i de A y sustituir c por $c - c_{j_i}\, a^i$, donde

$$c_{j_i} = \text{MIN}\ \{c_j \mid a_{ij} = 1\}$$

y repetir el proceso para las demás filas hasta haberlas tenido en cuenta todas. La justificación de este procedimiento se deduce del mismo argumento utilizado en el caso de la sustitución de los costes por los costes reducidos.

La experiencia computacional obtenida por nosotros con este último procedimiento ha sido excelente, aunque presenta algunos inconvenientes que, como comentaremos en el apartado siguiente, pueden hacer más interesante la primera alternativa, según los procedimientos que se implementen.

2.1.3.2. Utilización de restricciones de sustitución

Otra forma de mejorar la eficiencia del esquema enumerativo es calcular cotas inferiores del coste de la completación de cada solución parcial, o en el caso de estar utilizando los costes reducidos, restringir la exploración a aquellas variables j para las que

$$c'_j \leq z_u - \sum_{i \in I} u_i$$

donde u_i son las soluciones del problema dual y $c'_j = c_j - c_B\, B^{-1}\, a_j$ y z_u una cota superior válida del valor óptimo Z_o , o lo que es equivalente, fijar a 0 aquellas variables para las que

$$c'_j > z_u - \sum_{i \in I} u_i$$

Analogamente podemos restringir la exploración a aquellas soluciones potenciales x, para las que

$$\sum_{j \notin S} c_j x_j < \bar{z} - \sum_{j \in S} c_j\, \bar{x}_j$$

siendo $\bar{z}$ la solución incumbente. Ello implica que podemos hacer

$$x_j = 0 \quad \text{si } c_j > \bar{z} - \sum_{j \in S} c_j\, \bar{x}_j \qquad j \notin S$$

Pero de todos estos procedimientos el más potente es el de las restricciones de sustitución, (Geoffrion, /13/). En el nodo k-ésimo del árbol enumerativo, definida la solución parcial S, el subproblema planteado puede formularse como:

$$[\mathrm{MIN}] \quad z_k = \sum_{j \notin S} c_j x_j + z(S)$$

$$\sum_{j \notin S} a_{ij} x_j = 1, \quad \forall i \notin R(S)$$

$$x_j \in \{0,1\}, \quad \forall j \notin S$$

Toda combinación lineal de las restricciones $i \notin R(S)$, con coeficientes u_i no negativos, es una restricción de sustitución, que nos proporciona el siguiente subproblema relajado, del subproblema k-ésimo:

$$[\mathrm{MIN}] \quad \bar{z}_k = \sum_{j \notin S} c_j x_j + z(S)$$

$$\sum_{i \notin R(S)} u_i \sum_{j \notin S} a_{ij} x_j = \sum_{i \notin R(S)} u_i$$

$$x_j \in \{0,1\}, \quad \forall j \notin S$$

la solución del subproblema relajado proporciona una cota inferior, Geoffrion, /14/, de la completación de la solución parcial S en dicho nodo, cota inferior cuyo valor viene dado por:

$$z_I(S) = \bar{z}_k + z(S)$$

Cada selección de un conjunto de coeficientes u_i nos proporciona una restricción de sustitución diferente y con élla una relajación distinta y, por consiguiente una nueva cota inferior. De entre todos los conjuntos de coeficientes u_i posibles, los que proporcionan la restricción de sustitución más potente, y por lo tanto la mejor cota inferior, son aquellos u_i solución del programa lineal dual del subproblema dado. Es en este sentido en el que, en el apartado anterior, comentábamos que en algunos casos podría ser más interesante la resolución del dual a pesar de la desventaja que representaba el tener que resolver el programa lineal correspondiente, ya que entonces, además de la ventaja de disponer de los costes reducidos para redefinir el vector c, se dispone de coeficientes que permiten construir una restricción de sustitución que proporciona la mejor cota inferior posible.

Una restricción de sustitución alternativa, propuesta por Pierce y Lasky, /10/, que no requiere de la resolución de un programa lineal es:

$$[\mathrm{MIN}] \quad z_k = \sum_{j \notin S} c_j x_j + z(S)$$

$$\sum_{j \notin S} v_j x_j = t$$

$$x_j \in \{0,1\}, \quad \forall j \notin S \qquad \text{(SSk)}$$

donde $v_j = \sum_{i \notin R(S)} a_{ij}$, y $t = m - |R(S)|$. Esta restricción de sustitución corresponde a hacer $u_i = 1, \forall i \notin R(S)$ y $u_i = 0, \forall i \in R(S)$. El subproblema de sustitución en el nodo k-ésimo (SSk), es en cada caso un problema de knapsack que o bien se resuelve exactamente por medio de algún algoritmo especializado, (Martello y Toth, /15/), o bien aproximadamente a través de un algoritmo rápido (Ibarra y Kim, /16/).

Esta última alternativa es particularmente sencilla de implementar y proporciona rápidamente buenas soluciones.

2.1.3.3 Explotación adicional de la estructura de la matriz A

En el curso de las experiencias realizadas durante la implementación de las alternativas descritas en los apartados anteriores surgieron, por observación, dos ideas sobre las posibilidades de reducir los tiempos de exploración teniendo en cuenta la estructura de bloques dada a la matriz y la ordenación dentro de cada bloque.

Una primera observación empírica fue que en general la exploración de cada rama del árbol enumerativo no es muy profunda, y como tal exploración, según el esquema enumerativo, descrito, está determinada por el número de elementos de los primeros bloques. Consecuentemente al definir los bloques, una reordenación de las filas que minimice el número de elementos de los primeros bloques (y no sólo del primero como preconizan otros autores), reduce notablemente los tiempos de ejecución, como hemos podido comprobar en las experiencias realizadas.

Por otra parte si en la exploración de la completación de una solución parcial se puede detectar una incompatibilidad, por falta de recubrimiento de una de las filas, antes de llegar a explorar un retroceso puesto que podemos garantizar, por la incompatibilidad, que entre los descendientes del nodo que dio origen a la rama que se explora, no hay ninguna solución posible del problema. La detección se puede realizar definiendo un contador por cada fila, que indique el número de columnas que pueden recubrirla (número de elementos de la fila). Al elegir una columna de un bloque aún no considerado, como en toda solución participará como máximo una columna de cada bloque, las restantes columnas del bloque en cuestión ya no pueden intervenir en la formación de la solución, por lo tanto, podemos decrementar, en las cantidades correspondientes, los contadores de las filas que recubrían dichas columnas. Si algún contador se anula, hemos detectado una fila que quedaría sin recubrir en la completación de la solución parcial en curso, por lo que podemos proceder a un retroceso sin necesidad de profundizar más la exploración. Si por el contrario ningún contador se anula, las filas que la columna recubre, ya quedan consideradas en la partición, por lo tanto las columnas de los bloques posteriores al bloque en curso de tratamiento, (bloques aún no tratados), que recubran alguna de dichas filas, no podrán formar parte de la solución y por lo tanto podemos decrementar los contadores de las filas correspondientes. En caso de que alguno se anule se procede al retroceso.

3. EXPERIENCIA COMPUTACIONAL Y CONCLUSIONES

En la puesta a punto del programa que incorpora las diferentes mejoras discutidas, se han ido resolviendo diferentes series de problemas test. Las más significativas han sido las dos últimas consistentes una en un conjunto de 10 problemas test, generados aleatoriamente, de 200 filas y 2500 columnas (variables), de una densidad de entre el 4% y el 8%, y dos problemas reales, generados por la heurística descrita, de 460 filas y 6065 columnas y densidad 2.4%, y 460 filas y 2425 columnas y densidad 2.5% respectivamente.

Test que fueron pasados en un VAX 750 en la Facultad de Informática de la U.P.B.. Los problemas no fueron resueltos hasta obtener el óptimo, las ejecuciones se interrumpieron cuando la mejora entre las sucesivas incumbentes eran inferiores a un 1%, y los tiempos que se han tenido en cuenta no han sido los de CPU pura, sino los de "elapsed time". Las razones para hacerlo así han sido fundamentalmente por considerar que la herramienta informática que estábamos desarrollando debía ser, para la Empresa del Ferrocarril Metropolitano de Barcelona, una herramienta de gestión, a utilizar interactivamente por el gabinete técnico responsable de la elaboración de horarios y para ellos lo más importante era disponer rápidamente del máximo número de soluciones aceptables, entre las cuales elegir con otros criterios no fácilmente cuantificables, o a partir de las cuales, por procedimientos manuales, poder retocarlas para obtener una solución satisfactoria cuyo coste está acotado previamente.

La primera serie de problemas test produjo soluciones, en las condiciones citadas, en tiempos del orden, en promedio, de los 3 minutos y los segundos entre 3 y 5 minutos de "elapsed time".

A partir de estos resultados en estos momentos estamos procediendo a la extensión de la experiencia a las restantes líneas.

4. NOTA ADICIONAL

La tarea de programación, así como la paternidad de las mejoras detalladas en el apartado 2.1.3.3 ha sido desarrollada por el equipo de alumnos de 5º curso de la FIB, Montserrat Montiel, Francesc Cuatrecasas y Eladio Valencia.

La heurística de generación de servicios posibles ha sido programada por Montserrat Carbó de la Empresa Ferrocarril Metropolitano S.A.

5. REFERENCIAS BIBLIOGRAFICAS

/1/ E. Balas and M. W. Padberg
Set Partitioning: A Survey, in Combinatorial Optimization, eds. N. Christofides, A. Mingozzi, P. Toth and C. Sandi, John Wiley and Sons, (1979)

/2/ L. Bodin, B. Golden, A. Assad and M. Ball
Routing and Scheduling of Vehicles and Crews: The state of the Art
Comp. and Oper. Res., 10, 2, (1983)

/3/ R.S. Garfinkel and G.L. Nemhauser
Integer Programming
John Wiley and Sons, (1972)

/4/ M. Gondran et J.L. Laurière
Un Algorithme pour le problême de partitionement
RAIRO, 8, (1974), 27-50

/5/ J. Hoffstadt
Computerized Vehicle and Crew Scheduling for the Hamburger Hochbahn Aktiengesellschaft in: Computer Scheduling of Public Transport: Urban Passenger and Crew Scheduling.
Edited by A. Wren, North Holland, (1981)

/6/ R. Lessard, J.M. Rousseau and D. Dupuis
Hastus I: A Mathematical Programming Approach to the Bus Driver Scheduling Problem
in: Computer Scheduling of Public Transport, Ed. by A. Wren, North Holland, (1981)

/7/ R.E. Marsten
An Algorithm for Large Set Partitioning Problems
Management Science, 20, (1974), 779-787

/8/ G. Mitra and A. Welsch
A Computer Based Crew Scheduling System Using a Mathematical Programming Approach
in: Computer Scheduling of Public Transport, Ed. by A. Wren, North Holland, (1981)

/9/ M.E. Parker and B.M. Smith
Two Approaches to Computer Crew Scheduling
in: Computer Scheduling of Public Transport, Ed. by A. Wren, North Holland, (1981)

/10/ J.F. Pierce and J.S. Lasky
Improved Combinatorial Programming Algorithms for a Class of All Zero-One Integer Programming Problems
Management Science, 19, (1973), 528-543

/11/ D. Ryan and B. Foster
An Integer Programming Approach to Scheduling
in Computer Scheduling of Public Transport, ed. by A. Wren, North Holland, (1981)

/12/ J. Delorme et E. Heugron
Problemes de Partitionement: Exploration Arborescente ou Methode Troncatures?
R.A.I.R.O., 9, v-2, (1975), 53-65

/13/ A.M. Geoffrion
An Improved Implicit Enumeration Approach for Integer Programming
Operations Research, 17, (1969), 437-454

/14/ A.M. Geoffrion and R.E. Marsten
Integer Programming: A framework and State-of-the-Arth Survey
Management Science, 18, (1972), 465-491

/15/ S. Martello and P. Toth
The 0-1 Knapsack Problem
in: Combinatorial Optimization, Ed. by N. Christofides, A. Mingozzi, P. Toth and C. Sandi, John Wiley and Sons, (1979)

/16/ O.H. Ibarra and C.E. Kim
Fast Approximation Algorithms for the knapsack and Subset Sum Problems
J. Assoc. Comput. Mach., 22, (1975), 463-468