

EXPERIENCIAS COMPUTACIONALES CON PROCEDIMIENTOS DE IDENTIFICACIÓN DE RESTRICCIONES PARA ALGUNOS TIPOS DE PROGRAMAS ENTEROS

JAIME BARCELÓ

UNIVERSIDAD POLITÉCNICA DE CATALUNYA

Desde los primeros trabajos de Padberg, Grötschel y otros, los procedimientos de identificación de restricciones han demostrado su utilidad en la resolución de clases especiales de -- problemas enteros de estructura combinatoria, tales como el del viajante de comercio, los de apareamientos en grafos, el de la mochila, etc., entre otros.

Por otra parte, muchos otros tipos de problemas enteros, incluyen en su estructura aspectos combinatorios, como es el caso, por ejemplo, de los problemas de localización de plantas con restricciones de capacidad y los de particionamiento de conjuntos.

Este trabajo tiene una doble intención, por una parte dar una breve panorámica de los procedimientos de identificación de restricciones, y su utilización algorítmica general, y por -- otra parte estudiar su aplicación a algunos problemas particulares, explotando algunos aspectos de su estructura combinatoria.

Los problemas de localización de plantas con restricciones de capacidad, admiten formulaciones alternativas algunas de las cuales incluyen restricciones de tipo knapsack, especialmente si se añaden las aparentemente redundantes restricciones knapsack surrogadas sobre las capacidades de las plantas. Este trabajo analiza computacionalmente el impacto que tales formulaciones y las restricciones complementarias generadas por un procedimiento de identificación de restricciones, tienen en la resolución de dichos problemas.

Los problemas de partición de conjunto pueden ser resueltos mediante algoritmos que utilicen los cortes disyuntivos que su estructura permiten generar. Balas y Padberg proponen además otras familias de cortes derivados de subproblemas planteados en el grafo de intersección -- fuerte asociado. En este trabajo se estudia computacionalmente los resultados de incorporar, según un procedimiento de tipo identificación de restricciones, planos secantes derivados de desigualdades de clique en el grafo asociado.

Keywords: INTEGER PROGRAMMING, CONSTRAINT IDENTIFICATION.

1. PROGRAMACION LINEAL ENTERA Y PLANOS SECANTES.

Un enfoque algorítmico clásico para la resolución de los problemas de programación lineal entera, ha sido el siguiente: Dado el problema lineal entero (P):

$$(P) \text{ MIN } cx, \quad x \in S \tag{1}$$

donde S, conjunto de soluciones posibles está definido por

$$S = \{x \in \mathbb{R}^n \mid Ax \leq b, \quad x \geq 0 \text{ y } x \text{ entero}\} \tag{2}$$

se suprimen las condiciones de integridad sobre los vectores x en (2) y se resuelve el problema (\bar{P}), relajación lineal, resultante. Si la solución óptima \bar{x} , del problema (\bar{P}) es entera, entonces, teniendo en cuenta las re-

laciones entre las soluciones posibles de un problema de programación matemática y las de una relajación válida del mismo, (Geoffrion and Marsten, /9/), la solución \bar{x} es una solución óptima para el problema (P). Si \bar{x} no es entera, entonces es deseable eliminar \bar{x} del conjunto $\{x \in \mathbb{R}^n \mid Ax \leq b, \quad x \geq 0\}$ añadiendo una de igualdad adicional, o plano secante, tal como

$$ax \leq a_0 \tag{3}$$

de manera que:

$$\bar{x} \notin \{x \in \mathbb{R}^n \mid Ax \leq b, \quad ax \leq a_0\} \tag{3a}$$

y

- Jaume Barceló - Universitat Politècnica de Catalunya - Facultat d'Informàtica, Dep. d'Estadística i Investigació Operativa - Jordi Girona Salgado, 31 - 08034 Barcelona.

- Article rebut 1' octubre de 1985.

$$\{x \in \mathbb{Z}^n \mid Ax \leq b, x \geq 0\} = \{x \in \mathbb{Z}^n \mid Ax \leq b, ax \leq a_0, x \geq 0\}$$

el algoritmo de Gomory, /10/, /11/, constituye un procedimiento sencillo para generar tales desigualdades a partir de la tabla óptima del simplex, correspondiente a la solución óptima \bar{x} .

Aunque se puede demostrar que el algoritmo de Gomory converge hacia una solución óptima del problema (P) en un número finito de planos secantes (3), la experiencia práctica ha demostrado que se trata de un algoritmo muy insuficiente ya que, además de toda una serie de problemas de estabilidad numérica, tendencia a la degeneración y al ciclo, etc., genera una gran cantidad de inecuaciones redundantes.

Todo ello sugiere que el planteamiento de Gomory es demasiado general y que puede ser mucho más eficaz intentar resolver no problemas generales de programación lineal entera, sino problemas particulares para los que se puedan generar los planos secantes - necesarios o los mejores, como por ejemplo, los que describen las caras de la envolvente convexa del politopo de los vectores de incidencia de las soluciones posibles. Este enfoque es, en principio, factible en tanto en cuanto un problema de optimización combinatoria, y por lo tanto un problema entero, admita una formulación equivalente en términos de la caracterización lineal de la envolvente convexa de sus vectores de incidencia.

En efecto, si definimos un problema lineal de optimización combinatoria de la manera siguiente (Grötschel, /13/):

Dado un conjunto finito E y una familia \mathcal{F} de subconjuntos de E denominada el conjunto de soluciones factibles, sea $c: E \rightarrow \mathbb{R}$ una función objetiva. Encontrar un conjunto factible $I^* \in \mathcal{F}$ tal que:

$$\sum_{e \in I^*} c(e) = \text{MIN} \left\{ \sum_{e \in I} c(e) \mid I \in \mathcal{F} \right\} \quad (4)$$

Todo problema de optimización combinatoria queda, pues, definido por la terna (E, \mathcal{F}, c) , donde E es el conjunto base, $\mathcal{F} \subseteq \mathcal{P}(E)$ es el conjunto de soluciones factibles, c es una función lineal objetiva, y se ha de en-

contrar una solución factible I^* tal que

$\sum_{e \in I^*} c(e)$ sea mínima, y a casi todo problema de optimización combinatoria (Grötschel, /14/) se le puede asociar un poliedro definido de la siguiente forma: Denotemos por \mathbb{R}^E el espacio vectorial real de $|E|$ dimensiones, en el que cada vector $x \in \mathbb{R}^E$ tiene sus componentes indexadas por los elementos de E. Para todo subconjunto $I \subseteq E$ definimos su vector de incidencia x^I , como el vector de componentes

$$x_e^I = 1 \quad \text{si } e \in I$$

$$x_e^I = 0 \quad \text{si } e \notin I$$

Al conjunto \mathcal{F} de soluciones posibles le asociamos la envolvente convexa de los vectores de incidencia de los elementos de \mathcal{F} , es decir, el politopo $P_{\mathcal{F}} \subseteq \mathbb{R}^E$ definido por:

$$P_{\mathcal{F}} = \text{conv} \{x^I \in \mathbb{R}^E \mid I \in \mathcal{F}\} \quad (5)$$

y el problema de optimización combinatoria (4) se puede formular como

$$\text{MIN} \{cx \mid x \in P_{\mathcal{F}}\} \quad (6)$$

que es un problema de programación lineal dado que $P_{\mathcal{F}}$ es un poliedro. Ahora bien, desde un punto de vista algorítmico la forma de (6) no es apta para que el problema sea resuelto numericamente por ninguno de los algoritmos habituales, diseñados para trabajar con politopos representados por sistemas de desigualdades lineales. Este inconveniente puede resolverse, desde una perspectiva teórica, gracias al teorema de Weyl, /28/, que establece que para todo poliedro $P_{\mathcal{F}}$ de la forma (5) existen una matriz A, de dimensiones $(m, |E|)$, y un vector $b \in \mathbb{R}^m$ tales que:

$$P_{\mathcal{F}} = \{x \in \mathbb{R}^E \mid Ax \leq b\} \quad (7)$$

sin embargo, desde un punto de vista práctico aparece un nuevo inconveniente debido a que el número de filas de la matriz A puede ser exponencial en $|E|$ aunque el problema (4) sea soluble en tiempo polinómico. Afortunadamente Grötschel et al. /15/, demuestran que en la resolución de problemas lineales de la forma (6) lo que es determinante no es el número de desigualdades, sino la estructura del sistema (7) de desigualdades, por lo cual un procedimiento que permita generar las restricciones adecuadas para la re-

solución del problema combinatorio, a medida que se necesitan, permite resolver el problema (6) sin necesidad de identificar explícitamente todas las desigualdades que define (7).

Este planteamiento general adquiere, en el caso de los problemas enteros, la siguiente forma: una formulación típica de un problema entero en forma de variables, 0-1 es:

$$\text{MIN } cx, \quad x \in D \quad (8)$$

$$(P) D = \{x \in \mathbb{R}^n \mid Ax \leq b, \quad x_j \in \{0, 1\}, \quad \forall j \in \{1, \dots, n\}\}$$

para el cual la familia \mathcal{F} puede definirse como:

$$\mathcal{F} = \{I \subseteq \{1, \dots, n\} \mid x^I \in D\}$$

con lo cual el politopo $P_{\mathcal{F}}$ es:

$$P_{\mathcal{F}} = \text{conv}(D) \quad (9)$$

La relajación lineal del problema (8) consistente en la supresión de las condiciones de integridad, define el poliedro

$$P_R = \{x \in \mathbb{R}^n \mid Ax \leq b, \quad 0 \leq x_j \leq 1, \quad \forall j \in \{1, \dots, n\}\} \quad (10)$$

tal que:

$$P_{\mathcal{F}} \subseteq P_R$$

y

$$\text{MIN } \{cx \mid x \in P_{\mathcal{F}}\} \geq \text{MIN } \{cx \mid x \in P_R\} \quad (11)$$

pudiéndose demostrar, Grötschel, /14/, que todo punto extremo de $P_{\mathcal{F}}$ es un punto extremo de P_R pero no al contrario, sin embargo, si el segundo problema de optimización de (11) tiene un vector solución cero-uno, entonces queda resuelto el problema (8).

Dado que, como hemos comentado antes, aunque se conozca una descripción de P como sistema de desigualdades, no puede usarse en la práctica dado el gran número de ellas, no puede pensarse en una utilización directa de la relación (11). Sin embargo, si es posible resolver el problema de separación para $P_{\mathcal{F}}$ - (Grötschel et al. /15/, Grötschel, /13/):

Problema de separación para $P_{\mathcal{F}}$:

Dado un vector $y \in \mathbb{R}^n$, el problema de separación para $P_{\mathcal{F}}$ consiste en demostrar que $y \in P_{\mathcal{F}}$, o encontrar un vector $d \in \mathbb{R}^n$ tal que

$$\forall x \in P_{\mathcal{F}} : dx < d_0$$

entonces a partir del vector d puede construirse una desigualdad válida

$$dx \leq d_0$$

para el politopo $P_{\mathcal{F}}$ de la que, eventualmente puede derivarse una faceta, es decir, una desigualdad tal que: (Pulleyblank, /23/).

$$1) P_{\mathcal{F}} \subseteq \{x \in \mathbb{R}^n \mid dx \leq d_0\}$$

$$2) \dim(P_{\mathcal{F}} \cap \{x \in \mathbb{R}^n \mid dx = d_0\}) = \dim P_{\mathcal{F}} - 1$$

se trata, por lo tanto, de un hiperplano de soporte de $P_{\mathcal{F}}$ generado por puntos extremos de $P_{\mathcal{F}}$.

La adición de tal desigualdad al sistema de desigualdades que define (8) permite una nueva relajación (10) que se aproxime mejor a las condiciones de (11). La situación mejora si en lugar de plantearnos el problema de separación para un punto $y \in \mathbb{R}^n$ cualquiera, nos lo planteamos, inspirándonos en el procedimiento de Gomory, para el punto \bar{x} -- proporcionado por la solución del programa lineal asociado a la relajación (10) de manera que se verifiquen las condiciones (3a), se trata pues de buscar una desigualdad válida que genere una faceta de P que elimine la solución de la relajación continua, sin eliminar ninguna solución factible entera. Esta reformulación del problema de separación se conoce como problema de identificación de facetas (Grötschel, Padberg, /16/, - Padberg, Rao, /21/):

Problema de identificación de facetas:

Dado un punto $\bar{x} \in \mathbb{R}^n$, y un politopo $P_{\mathcal{F}}$ encontrar una desigualdad lineal

$$dx \leq d_0$$

válida para $P_{\mathcal{F}}$ que defina una faceta de $P_{\mathcal{F}}$ tal que sea violada por \bar{x} , es decir, tal que se verifique $d\bar{x} > d_0$ o demostrar que no existe tal desigualdad, es decir, que se verifica $\bar{x} \in P_{\mathcal{F}}$.

Utilizando este procedimiento se pueden generar desigualdades que definan caras a medida que sean necesarias, para eliminar los óptimos fraccionales que proporciona la relajación lineal del problema entero, tal como indica el esquema algorítmico siguiente: (Grötschel and Padberg, /16/):

METODO DE IDENTIFICACION DE RESTRICCIONES PARA PROBLEMAS ENTEROS.

1. INICIALIZACION

Sea (LP_0) una relajación lineal válida del problema de programación lineal entera que se pretende resolver.

Hacer $k=0$, e ir a 2.

2. RESOLUCION DEL PROBLEMA LINEAL LP_k .

Resolver (LP_k) . Sea x^k una solución óptima de (LP_k) . Si x^k es entera, STOP, en caso contrario.

Ir a 3.

3. IDENTIFICACION DE FACETAS.

Resolver el problema de identificación de facetas para x^k y el politopo P_F .

3.1 Si se encuentran una o más desigualdades lineales definidoras de facetas para P_F violadas por x^k , definir (LP_{k+1}) , añadiendo a (LP_k) las desigualdades obtenidas.

Hacer $K = K+1$, e ir a 2.

3.2 Si no existe ninguna desigualdad lineal definidora de facetas violadas por x^k , STOP.

Suponiendo que en el paso 2 se utilice un método finito para resolver el programa lineal, el procedimiento propuesto es finito ya que el número de desigualdades que definen caras de un politopo P_F es finito.

En los restantes apartados de este trabajo se describen adaptaciones particulares de este procedimiento general a los problemas de localización de plantas con restricciones de capacidad, y a los problemas de particionamiento de conjuntos.

2. CONSIDERACIONES SOBRE LA FORMULACION DE LOS PROBLEMAS DE LOCALIZACION CON RESTRICCIONES DE CAPACIDAD.

Este estudio computacional es el resultado de una combinación de hechos teóricos y empíricos sobre los problemas de localización de restricciones de capacidad. Entre los hechos empíricos debemos mencionar la potencia de las formulaciones desagregadas de los problemas de localización de plantas, como por ejemplo, las que incluyen restricciones desagregadas de la forma

$$x_{ij} \leq b_j v_j, \quad \forall i \in I, \quad \forall j \in J \quad (12)$$

donde las y_j son las variables de decisión 0-1 asociadas con las plantas, $J=\{1, \dots, m\}$, es el conjunto de los emplazamientos potenciales para las plantas, y x_{ij} son las variables asociadas con los suministros al centro i desde la planta j , siendo $I = \{1, 2, \dots, m\}$, el conjunto de los centros cuyas demandas d_i han de ser satisfechas, y $b_j, j \in J$, las capacidades de las plantas. En un trabajo pionero, Spielberg, /26/, planteó la importancia de tales restricciones y posteriormente, Schrage, /24/, experimentó ampliamente los efectos computacionales de la adición de las restricciones (12). Geoffrion y McBride, /8/, han explicado teóricamente porque estas restricciones son computacionalmente eficientes, al demostrar que son caras de la envolvente convexa del problema de localización de plantas con restricciones de capacidad.

Sin embargo, desde la perspectiva de la programación lineal resolver problemas de localización de plantas con restricciones de capacidad, incluyendo restricciones (12) en la formulación del problema como problema de programación lineal, tiene un elevado coste tanto por lo que al esfuerzo de formulación se refiere como al computacional dado el incremento de las dimensiones del problema, -- esfuerzo que puede anular las otras ventajas, en caso de no disponer de un código de programación lineal modificado, capaz de tratar implícitamente este tipo de restricciones, como un caso especial de acotación superior generalizada, (Schrage, /24/, /25/).

Otro hecho empírico a tener en cuenta es la

experiencia positiva de incluir además restricciones agregadas del tipo:

$$\sum_{j \in J} b_j y_j \geq d$$

donde $d = \sum_{i \in I} d_i$ es la demanda agregada, - en la formulación del problema de localización de plantas con restricciones de capacidad, como problema de programación entera mixta. Experiencia computacional sobre este hecho reportan Guignard y Spielberg, /17/, y Van Roy /27/, en un contexto algorítmico diferente, demostrando que las restricciones (13) ayudan a calcular planos de Benders mas potentes en el procedimiento de "cross-decomposition".

En el trabajo referenciado, Guignard y Spielberg también recomiendan la adición de restricciones "lógicas", o cortes, de la forma:

$$\sum_{j \in S} y_j \geq L, \quad S \subseteq J \quad (14)$$

pero no resuelven el problema de como construir sistemáticamente las mejores de tales restricciones lógicas, es decir, como identificar el mejor subconjunto S y calcular la correspondiente cota inferior L.

Teniendo en cuenta todas estas consideraciones, intentamos estudiar el comportamiento computacional de los problemas de localización de plantas con restricciones de capacidad, formuladas como problemas de programación entera mixta, cuando no se incluye restricciones desagregadas (12) dado que sólo se dispone de un código normal de programación lineal (el XMP de Roy Marsten en nuestro caso), pero basándonos en la teoría general desarrollada en el apartado 1., podemos diseñar un procedimiento automático para generar restricciones (14).

3. GENERACION AUTOMATICA DE RESTRICCIONES PARA PROBLEMAS DE LOCALIZACION DE PLANTAS CON RESTRICCIONES DE CAPACIDAD.

Cuando en la formulación de problemas de localización con restricciones de capacidad como problemas de programación entera mixta, se incluye el problema de Knapsack surrogado

$$\sum_{j \in J} b_j y_j \geq d, \quad y_j \in \{0,1\}, \quad \forall j \in J \quad (15)$$

y el politopo de knapsack asociado

$$K = \text{conv}\{y \in R^m \mid \sum_{j \in J} b_j y_j \geq d, y_j \in \{0,1\}, \forall j \in J\} \quad (16)$$

si P es el politopo asociado con el problema de localización con restricciones de capacidad, entonces Padberg, Van Roy y Wolsey, /22/, demuestran que se verifica la siguiente proposición:

Proposición 3.1

Toda desigualdad válida para K es una desigualdad válida para P.

(Que eventualmente si se verifican algunas condiciones adicionales, tal como se especifican en la referencia anterior, puede ser una cara de P).

Es bien sabido que el concepto de desigualdad de recubrimiento para los problemas de knapsack, (Balas, /1/), proporciona un procedimiento para calcular desigualdades válidas, y caras, para el politopo de knapsack K. Así pues, cuando en la formulación de un problema de localización de plantas con restricciones de capacidad se incluye una restricción surrogada de tipo knapsack como (15), el procedimiento para calcular desigualdades válidas a partir de los recubrimientos minimales proporciona un método sistemático para generar restricciones lógicas de la forma

$$\sum_{j \in S} \pi_j y_j \geq \pi_0 \quad (17)$$

que incluyen la familia de restricciones -- (14) como un caso particular, dado que en ciertas situaciones se pueden generar restricciones (17) con coeficientes $\pi_j > 1$ que, consecuentemente, son más potentes que las restricciones (14).

Este método da respuesta a la primera de las preguntas planteadas al final del apartado 2. La segunda, la de identificar la mejor de tales desigualdades, fue resuelta por Padberg al demostrar que la mejor de las desigualdades de recubrimiento minimales era la más -- violada por la solución \bar{y} de la relajación

lineal del problema.

Sea \bar{y} la solución de la relajación lineal - de la restricción surrogada de tipo knapsack (15), entonces, Growder, Johnson y Padberg, /17/, demuestran la siguiente proposición:

Proposición 3.2

Sea (z, s) una solución óptima del problema de knapsack

$$z = \text{MIN} \left\{ \sum_{i \in J} \bar{y}_j s_j \mid \sum_{j \in J} b_j s_j > \sum_{j \in J} b_j - d, \right. \\ \left. s_j \in \{0, 1\}, \forall j \in J \right\}$$

donde $S \subseteq J$ es el soporte de s . Entonces:

Si $z < 1$ la desigualdad válida

$$\sum_{j \in S} y_j \geq 1 \tag{18}$$

para K (y por lo tanto para P) es violada por \bar{y} .

Sin embargo, en caso de que exista una desigualdad del tipo (18), será una faceta del politopo K^S definido por

$$K^S = \text{Kn} \{ y \in \mathbb{R}^m \mid y_j = 0, \forall j \in S \} \tag{19}$$

faceta que puede potenciarse, es decir proyectarse en el espacio de m dimensiones, para obtener una faceta de K , mediante el denominado procedimiento de "lifting", /7/, basado en los teoremas de lifting, /19/, demostrados en Padberg, /20/, y Balas y Zemel, /5/, cuyo esquema algorítmico es el siguiente:

Procedimiento secuencial de Lifting:

$$\text{Hacer } \pi_j = 1, \forall j \in S \\ f_0 = |S| - 1$$

Paso iterativo

Para $k \in J \setminus S$

Calcular:

$$z_k = \text{MAX} \sum_{j \in S} s_j \pi_j$$

$$\sum_{j \in S} b_j s_j \leq \sum_{j \in J} b_j - d - b_k, s_j \in \{0, 1\}, \forall j \in S$$

Definir

$$\pi_k = f_0 - z_k$$

$$S = S \cup \{k\}$$

Repetir hasta que $J \setminus S = \emptyset$

Entonces definir:

$$\pi_0 = \sum_{j \in J} \pi_j - f_0$$

la desigualdad

$$\sum_{j \in J} \pi_j y_j \geq \pi_0 \tag{20}$$

es la faceta de K más violada por la solución en curso \bar{y} de la relajación lineal, y como es una desigualdad válida para P (que eventualmente puede ser una faceta de P), elimina dicha solución.

La experiencia computacional que este trabajo presenta, está basada en el enfoque algorítmico descrito en el apartado 1., cuyo diagrama de flujo queda representado en la Fig. 1, que esencialmente consiste en, dada una formulación del problema de localización de plantas con restricciones de capacidad, como problema de programación entera mixta, incluyendo restricciones surrogadas de tipo knapsack como (15), se calculan, a partir de las soluciones a las relajaciones lineales, tantas restricciones (20) como sea posible y se añaden a la formulación del problema.

Cuando ya no sea posible añadir más restricciones (20) y no se haya llegado a una solución entera, se continua mediante un procedimiento de branch and bound. En nuestras experiencias, como utilizábamos el XMP, hicimos uso de la heurística de "pivot and -- complement" de Balas y Martín, /3/, para generar una cota superior inicial.

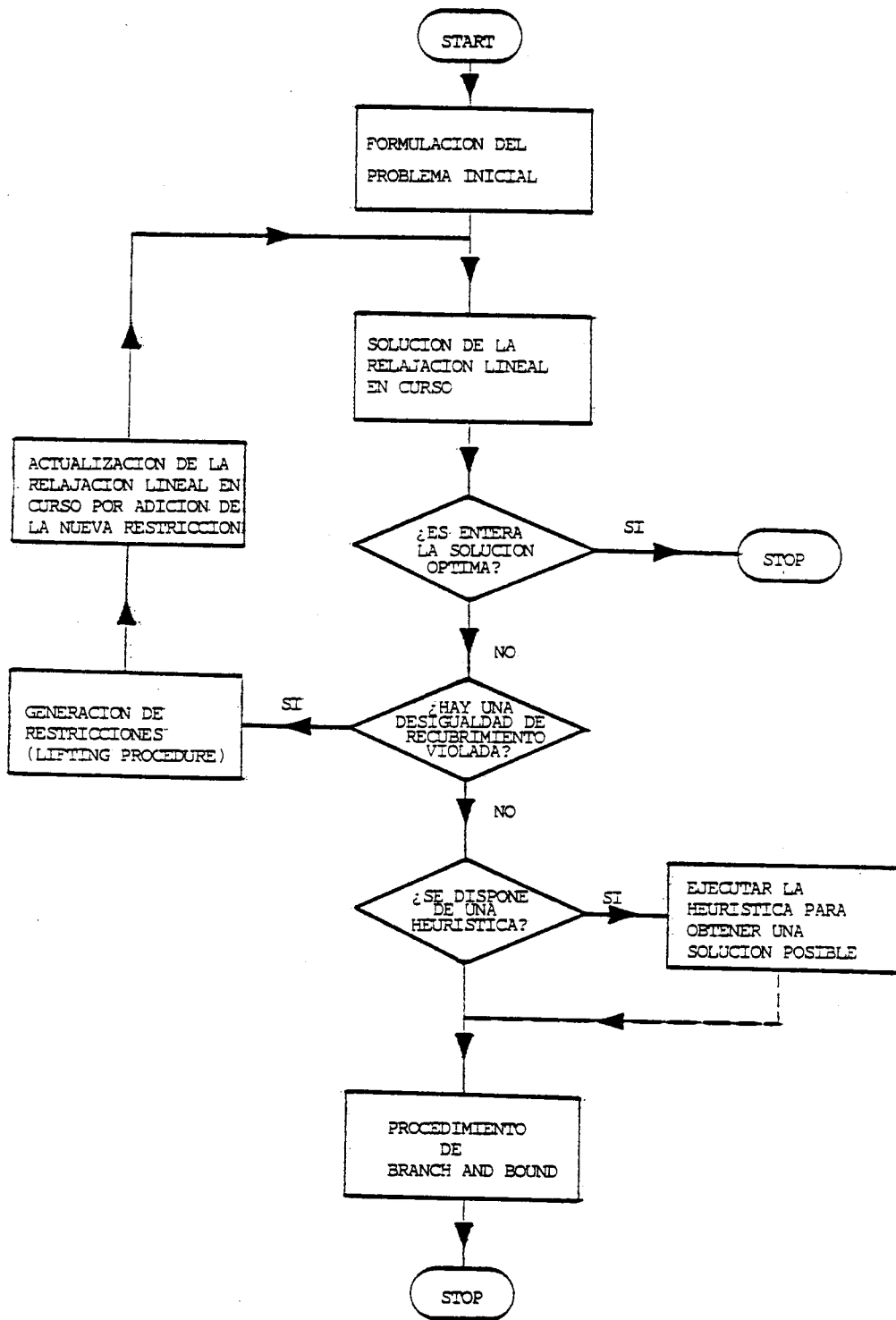


FIGURA 1.

4. FORMULACION DEL PROBLEMA.

La formulación del problema empleada en las experiencias computacionales que reportamos en este trabajo, ha sido la siguiente:

$$z = \text{MIN} \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} + \sum_{j \in J} f_j y_j$$

$$\sum_{j \in J} x_{ij} = 1, \quad \forall i \in I \quad (21)$$

$$\sum_{j \in J} y_j \leq K$$

$$\sum_{j \in J} b_j y_j \geq d, \quad d = \sum_{i \in I} d_i \quad (23)$$

$$\sum_{i \in I} d_i x_{ij} \leq b_j y_j, \quad \forall j \in J \quad (24)$$

$$0 \leq x_{ij} \leq 1, \quad i \in I, \quad \forall j \in J \quad (25)$$

$$y_j \in \{0, 1\}, \quad \forall j \in J \quad (26)$$

donde, como de costumbre, c_{ij} son los costes de transporte desde la planta j hasta el centro i , y f_j son los costes fijos de la decisión de abrir una planta en la ubicación potencial j . El conjunto de restricciones (21) asegura que la demanda de todos los centros será satisfecha, la restricción (22) limita el número de plantas a abrir a un máximo de K , y la restricción (23) es del tipo de restricciones de knapsack surrogadas que hemos discutido antes. El conjunto de restricciones (24) evita el intento de aprovisionar un centro desde una planta que no ha sido abierta ($y_j=0 \Rightarrow x_{ij}=0, \forall i \in I$), y no permite que el aprovisionamiento desde una planta supere la capacidad de la misma cuando ésta está abierta. Estas restricciones pueden interpretarse como una agregación de restricciones de tipo (12), que conduce a una relajación lineal -- más débil pero que requiere un menor esfuerzo computacional, como hemos comentado en el apartado 2., cuando no se dispone de un código para programación lineal modificado.

Las restricciones (26) se sustituyen por

$$0 \leq y_j \leq 1, \quad \forall j \in J \quad (27)$$

para obtener la relajación lineal ordinaria, en cuyo caso las restricciones (25) y (27) pueden ser tratadas como restricciones ordinarias de acotación superior de variables.

Después de cada solución de la relajación lineal, la restricción (23) genera una restric-

ción de la forma

$$\sum_{j \in J} \pi_j y_j \geq \pi_0 \quad (20)$$

según el procedimiento descrito en el apartado 3.

Una alternativa que también se puede estudiar con esta formulación es la de los problemas enteros puros de localización de -- plantas con restricciones de capacidad, que corresponde al caso en que cada centro ha de ser aprovisionado solo desde una planta. Este problema se obtiene sustituyendo las restricciones (25) por

$$x_{ij} \in \{0, 1\}, \quad \forall i \in I, \quad \forall j \in J \quad (28)$$

En este caso las restricciones (24) son también restricciones de tipo knapsack y pueden generar restricciones adicionales de la forma

$$\sum_{i \in I} \pi_i x_{ij} \leq b_j y_j, \quad j \in J^* \quad (29)$$

siendo J^* el conjunto de plantas abiertas, de la misma forma que se generan las restricciones (20) a partir de (23).

5. EXPERIENCIA COMPUTACIONAL.

Se han resuelto seis series de problemas de dimensiones y características diferentes, y los resultados se exponen en las tablas 1 a 6. Todos ellos fueron generados utilizando un generador de problemas test de las siguientes características. Dados los parámetros DMIN, valor mínimo de la demanda, DPROM, valor medio aproximado de las demandas, CMIN, mínimo coste de transporte, CPROM, valor medio aproximado de los costes de transporte, y DCOEF, relación media entre las capacidades de las K plantas y la demanda total d , los datos del problema se generan de la manera siguiente: el número límite de plantas a abrir, K , se genera en el intervalo ---- $[0, \ln, n]$, siendo n el número de centros de finidos por el usuario; las demandas d_i se generan suponiéndolas uniformemente distribuidas en un intervalo centrado en DPROM -- que tiene DMIN como límite inferior; una vez se han generado las demandas, se calcula -- BPROM, capacidad media de las plantas, como (DEMANDA TOTAL)/(DCOEF *K), y entonces se -

generan las capacidades b_j suponiendo que están uniformemente distribuidas en el intervalo $[BMIN, 2^* BMIN]$, fijando BMIN al valor de DMIN. Los costes de transporte se generan suponiendo que responden a la estructura definida por

$$c_{ij} = a_{ij} + r_{ij} d_i$$

con dos componentes, a_{ij} componente fijo que depende de la planta, determinada aleatoriamente, y una segunda componente $r_{ij} d_i$ proporcional a la demanda. Los costes fijos f_j se pueden calcular de dos maneras, bien como directamente proporcionales a las capacidades o como proporcionales a las capacidades mas una componente aleatoria, pero en cualquier caso, suponiendo una cierta relación de proporcionalidad con los costes de transporte.

Los resultados computacionales de cada serie de experiencias se recogen en la tabla correspondiente, y son los siguientes: la primera columna de cada tabla identifica el problema y la segunda el estado final del problema. El estado a significa que para dicho problema la técnica de identificación de restricciones ha obtenido un resultado satisfactorio y ha reducido significativamente el esfuerzo de cálculo requerido para resolver el problema, mientras que el estado b lo representa la situación contraria, para esta clase de problemas la técnica de identificación de restricciones no sólo no ha tenido éxito, sino que de hecho ha incrementado la cantidad de cálculos necesarios para resolver el problema. Algunos problemas no tienen definido el estado, son problemas a los que no se puede aplicar esta técnica porque no es posible calcular -- planos secantes para los mismos. La tercera y cuarta columnas dan información sobre la estructura del problema, contienen respectivamente, los ratios entre la demanda total, la capacidad total $\sum_{j \in J} b_j$ ofrecida por las plantas y entre los costes fijos f_j y los costes totales de transporte $\sum_{i \in I} c_{ij}$ para cada planta. Las columnas cinco y seis informan sobre el esfuerzo computacional requerido para resolver el problema utilizando un código branch and bound estandar, sin utilizar planos secantes. La columna cinco indica el número de nodos explorados y la

columna seis el número total de transformaciones pivotaes del simplex. La columna siete proporciona más información sobre las características del problema, los valores de esta columna representan el salto de discontinuidad (gap):

(valor óptimo entero - valor óptimo continuo) x 100 / valor óptimo continuo

Tal como dijimos anteriormente, la estrategia seguida ha sido la de calcular el mayor número de planos secantes posibles antes de proceder al branch and bound. La columna -- ocho proporciona el número de planos secantes calculados para cada problema. Las columnas nueve y diez informan sobre el esfuerzo computacional requerido para resolver el problema por el procedimiento estandar de branch and bound, después de añadir los planos secantes. La columna nueve da el número total de transformaciones pivotaes del simplex, incluyendo las requeridas para restablecer la posibilidad primal después de la adición de los planos secantes. La última columna contiene una estimación del porcentaje de la reducción del salto de discontinuidad como consecuencia de los planos secantes añadidos.

Los problemas de las tablas 1, 2 y 3 son de dimensión 20x10, es decir 20 centros y 10 plantas. La tabla 1 contiene 15 problemas generados con la hipótesis de que los costes fijos f_j eran proporcionales a las capacidades b_j , mientras que los problemas de las tablas 2 y 3 suponen que los costes f_j , son proporcionales a las capacidades b_j más una componente aleatoria, todos ellos fueron generados utilizando la misma simiente para el generador de números aleatorios que la usada en los problemas 1.3 a 1.13 de la tabla 1. En la tabla 3 se supuso además que las relaciones entre f_j y $\sum_{i \in I} c_{ij}$ eran variables.

La tabla 4 incluye 18 problemas de dimensiones 30x15, los problemas 4.1 a 4.9 fueron generados con las mismas hipótesis que los de la tabla 1, y los 4.10 a 4.18 con los criterios de las tablas 2 y 3, que a su vez fueron los utilizados en los 14 problemas de dimensiones 40x20 de la tabla 5 y en los 12 problemas de dimensiones 40x30 de la tabla 6, que incluye además, un problema comple-

mentario de dimensiones 50x25.

Todas las experiencias fueron realizadas en un VAX 785 utilizando el código XMP de Roy Marsten, con la heurística de "pivot and complement" para obtener soluciones posibles iniciales.

6. CONCLUSIONES.

Como hemos indicado anteriormente, los problemas señalados con una a son aquellos para los que la técnica de generación de restricciones tuvo éxito. Comparando las columnas cinco y nueve, y seis y siete, podemos apreciar la reducción en el coste computacional. Por ejemplo, para el problema 1.1 fué necesario explorar 48 nodos y realizar 1130 transformaciones pivotaes de la tabla del simplex para resolverlo directamente que, tras la adición de uno de los planos secantes propuestos quedarán reducidos a 16 nodos y 199 transformaciones pivotaes. El problema 5.7 representa uno de los casos más espectaculares ya que, en el intento de resolverlo directamente, no se obtuvo ninguna solución posible después de explorar 362 nodos y realizar más de 10.000 transformaciones pivotaes, y tras añadir cuatro planos secantes se obtuvo la solución óptima explorando 117 nodos y realizando 3329 transformaciones pivotaes.

Dentro de las clases a y b existen dos subclases, identificadas respectivamente por a* y b. en las tablas. Los problemas cuyo estado final es a* son aquellos para los cuales a pesar de haber sido fructífera la aplica-

ción de la técnica de generación de restricciones, los resultados hubiesen sido mejores si en lugar de generar directamente el mayor número de planos secantes posible, se hubiese parado antes el proceso de generación. Por ejemplo, el problema 2.7 necesitó 66 nodos y 1188 pivots para ser resuelto directamente y 43 nodos y 362 pivots para ser resuelto después de añadir 5 planos secantes, pero si el proceso de generación de planos se hubiese detenido después de añadir el tercero, entonces la resolución del problema hubiese requerido únicamente la exploración de 19 nodos y la realización de 263 pivots, es decir un resultado todavía mejor que el primero.

La subclase b. tiene un comportamiento análogo. El problema 6.9, por ejemplo, que es el que presenta los peores resultados, necesitó 241 nodos y 7341 pivots para ser resuelto directamente, números que se incrementaron a 357 nodos y 10049 pivots después de añadir 8 planos, sin embargo si se hubiesen añadido los planos uno a uno, hubiésemos visto que tras la adición del primer plano secante se hubieran necesitado solo 160 nodos y 5140 pivots, y que el segundo y tercer plano proporcionaban resultados similares, mientras que a partir del tercer plano se iniciaba un proceso de empeoramiento de los resultados que terminaba después del octavo plano secante con los resultados referidos.

La tabla siguiente resume los resultados de los 80 problemas presentados en las tablas 1 a 6:

Tabla Clase	1	2	3	4	5	6	Número total de problemas en la clase
a	6	5	5	7	12	6	41
b.	1	3	2	1	0	5	12
b	2	2	2	4	2	2	14
sin planos secantes	6	0	1	6	0	0	13
	15	10	10	18	14	12	80

TABLA 1

DIMENSION 20 X 10 , $f_j = K_j b_j$

PROBLEMA	estado	RATIO	RATIO	BRANCH AND BOUND		GAP %	NUMERO DE PLANOS SECANTES	BRANCH AND BOUND		REDUCCION DEL GAP	
		$r = \bar{d}/\bar{b}$	f/c	NODOS	PIVOTS LP			NODOS	PIVOTS LP		
1.1	a	0.70	1.40	48	1130	10.75	1	16	199	99.71	
1.2	a	0.75	1.40	48	466	5.76	1	0	132	100.00	
1.3	-	0.47	4.45	44	512	9.83	-	-	-	--	
1.4	a	0.58	3.18	16	276	1.34	2	16	267	1.65	
1.5	a	0.29	7.26	58	646	2.34	-	-	-	--	
1.6	-	0.23	6.99	43	587	6.50	-	-	-	--	
1.7	b	0.58	3.23	53	1407	0.57	1	53	1564	1.83	
1.8	-	0.43	4.77	44	526	4.04	-	-	-	--	
1.9	-	0.38	4.58	44	552	7.11	-	-	-	--	
1.10	-	0.29	6.57	53	1714	1.60	-	-	-	--	
1.11	b.	0.81	1.27	54	469	1.56	3	51	505	22.04	
1.12	a	0.75	1.33	40	417	1.35	1	41	394	18.44	
1.13	a	0.77	0.63	51	461	0.68	1	16	269	4.72	
1.14	b	0.64	0.61	67	523	6.65	1	67	580	0.015	
1.15	a	0.77	0.53	22	350	5.92	3	16	172	10.36	

TABLA 2

PROBLEMA	estado	DIMENSION 20×10 , $f_j = K_j b_j + r_j$ (r_j aleatorio)		RATIO		BRANCH AND BOUND		GAP %	NUMERO DE PLANOS SECANTES	BRANCH AND BOUND		REDUCCION	
		$r = \sum d / \sum b$	$f / \sum c$	NODOS	PIVOTS LP	NODOS	PIVOTS LP			DEL GAP			
2.1	a	0.64	0.79	46	785	16	301	6.19	5	16	301	17.41	
2.2	b.	0.47	4.77	37	399	19	409	3.00	5	19	409	72.64	
2.3	a	0.58	3.67	63	763	51	451	3.40	2	51	451	32.17	
2.4	b	0.29	6.95	41	393	46	681	25.48	3	46	681	14.48	
2.5	b.	0.23	6.49	19	240	16	251	7.25	4	16	251	95.40	
2.6	b	0.58	2.82	49	442	52	538	5.24	3	52	538	11.78	
2.7	a*	0.43	5.57	66	1188	43	362	2.13	5	43	362	23.84	
2.8	b.	0.38	4.13	34	346	19	357	1.53	2	19	357	36.68	
2.9	a	0.29	6.70	37	409	16	195	11.05	4	16	195	41.81	
2.10	a	0.64	1.94	64	1161	64	1078	1.00	1	64	1078	0.87	

TABLA 3

DIMENSION 20 X 10 , $f_j = K_j b_j + r_j$ (r_j aleatorio)

PROBLEMA	estado	RATIO		RATIO		BRANCH AND BOUND		CAP %	NUMERO DE PLANOS SECANTES	BRANCH AND BOUND		REDUCCION DEL GAP	
		$r = d/yb$	$f/\Sigma c$	NODOS	PIVOTS LP	NODOS	PIVOTS LP						
3.1	b	0.66	1.42	44	329	2.59	3	43	408	5.41			
3.2	a	0.68	1.99	37	390	1.78	3	36	293	8.50			
3.3	a	0.73	1.75	46	498	4.90	4	16	221	96.13			
3.4	b.	0.64	1.54	35	363	7.03	2	36	409	39.43			
3.5	a	0.59	1.77	46	438	2.36	7	16	230	89.28			
3.6	b.	0.68	2.26	35	325	2.05	6	34	374	49.92			
3.7	b	0.72	1.44	16	254	3.57	3	19	287	26.32			
3.8	a	0.72	1.72	35	565	5.28	4	16	248	97.51			
3.9	a	0.72	1.55	41	526	2.50	1	41	519	4.25			
3.10	-	0.79	1.29	45	461	0.29	-	-	-	-			

TABLA 4

DIMENSION 30 x 15

PROBLEMA	estado	RATIO $r = \bar{r}_d / \bar{r}_b$	RATIO f / \bar{r}_c	BRANCH AND BOUND		GAP %	NUMERO DE PLANOS SECANTES	BRANCH AND BOUND		REDUCCION DEL GAP	
				NODOS	PIVOTS LP			NODOS	PIVOTS LP	NODOS	PIVOTS LP
4.1	-	0.66	1.00	100	1482	0.87	-	-	-	-	-
4.2	b	0.70	0.98	82	1289	0.53	2	90	1257	8.34	-
4.3	a	0.87	0.36	43	746	0.76	6	37	590	16.16	-
4.4	-	0.67	0.46	98	1350	1.91	-	-	-	-	-
4.5	-	0.77	0.40	66	850	0.35	-	-	-	-	-
4.6	-	0.67	0.44	72	3825	0.93	-	-	-	-	-
4.7	-	0.70	0.43	84	1245	2.23	-	-	-	-	-
4.8	-	0.78	0.54	77	1167	1.65	-	-	-	-	-
4.9	b	0.81	0.54	16	387	0.09	3	37	756	99.26	-
4.10	a	0.89	0.63	48	640	4.83	4	36	423	90.77	-
4.11	a	0.67	0.36	84	1548	1.70	2	49	838	12.08	-
4.12	b	0.67	0.48	73	1749	2.70	1	214	4956	6.77	-
4.13	a	0.78	0.58	70	791	2.04	4	48	689	14.83	-
4.14	b	0.71	1.29	16	385	0.71	2	16	424	54.67	-
4.15	a	0.39	1.84	138	2316	3.30	3	72	818	51.27	-
4.16	b.	0.33	1.77	49	779	4.30	4	40	815	43.43	-
4.17	a*	0.44	1.90	96	2613	4.60	5	44	628	49.75	-
4.18	a*	0.38	1.75	113	1272	5.02	6	64	825	61.39	-

TABLA 5

PROBLEMA	estado	DIMENSION 40 x 20		RATIO		RATIO		BRANCH AND BOUND		GAP %	NUMERO DE PLANOS SECANTES	BRANCH AND BOUND		REDUCCION	
		$r = \frac{I_d}{I_b}$	f/x_c	NODOS	PIVOTS LP	NODOS	PIVOTS LP	NODOS	PIVOTS LP			DEL GAP			
5.1	a	0.78	0.47	103	2118	74	1219	3	74	2.42	3	74	1219	6.77	
5.2	a*	0.75	0.84	122	1779	42	958	6	42	1.26	6	42	958	27.77	
5.3	a*	0.81	0.92	84	2032	48	1389	5	48	1.13	5	48	1389	15.22	
5.4	a*	0.77	0.98	97	2078	75	1434	3	75	1.02	3	75	1434	22.04	
5.5	a	0.8	0.80	93	3151	75	2160	2	75	2.76	2	75	2160	3.40	
5.6	a*	0.76	1.00	222	5123	95	1777	5	95	1.87	5	95	1777	38.61	
5.7	a*	0.64	1.36	362	> 10000	117	3329	4	117	4.33	4	117	3329	19.97	
5.8	b	0.7	1.16	74	1479	83	1615	2	83	1.05	2	83	1615	2.95	
5.9	a	0.63	1.30	44	771	16	492	1	16	0.33	1	16	492	79.42	
5.10	b	0.57	1.48	101	1790	116	2052	1	116	1.90	1	116	2052	18.80	
5.11	a	0.19	3.30	73	1560	35	760	8	35	9.05	8	35	760	35.07	
5.12	a	0.42	1.75	44	889	39	869	4	39	2.90	4	39	869	7.24	
5.13	a	0.49	1.82	84	1533	62	1217	5	62	6.10	5	62	1217	80.04	
5.14	a	0.17	3.64	56	886	38	753	3	38	3.73	3	38	753	20.43	

TABLA 6

PROBLEMA	estado	DIMENSION 40 x 30		RATIO		BRANCH AND BOUND		GAP %	NUMERO DE PLANOS SECANTES	BRANCH AND BOUND		REDUCCION	
		$r = \sum d / \sum b$	$f / \sum c$	NODOS	PIVOTS LP	NODOS	PIVOTS LP			DEL GAP			
6.1	b.	0.39	1.47	48	680	0.66	3	35	1030	10.84			
6.2	b.	0.69	0.83	57	1360	0.28	5	49	1418	13.69			
6.3	b	0.48	0.88	216	3737	1.48	1	172	4930	0.39			
6.4	a	0.3	2.21	94	1958	0.54	5	41	585	26.40			
6.5	a	0.48	0.5	130	3124	0.75	1	43	812	71.83			
6.6	a	0.69	0.72	109	2160	0.26	2	22	765	6.46			
6.7	b	0.51	1.25	22	563	0.13	4	25	781	14.71			
6.8	a*	0.67	1.23	165	3460	0.47	5	16	659	94.78			
6.9	b.	0.4	1.14	241	7341	2.28	8	357	10049	8.87			
6.10	b.	0.71	0.58	72	1394	0.25	3	72	1649	12.30			
6.11	a*	0.59	0.66	-	---	1.31	5	148	3185	26.60			
6.12	b.	0.30	1.60	117	2560	4.28	3	125	2545	31.58			
(50 x 25)													
7.1	a	0.40	1.58	151	3091	1.86	2	135	2843	41.93			

A primera vista resulta que la técnica de generación de restricciones, ha proporcionado buenos resultados en aproximadamente el cincuenta por ciento de los casos (41 de 80), pero es obvio que este número se puede incrementar si se dispone de una regla de decisión para parar el proceso de generación -- cuando empieza el empeoramiento, y si se puede identificar a priori cuando un problema pertenece a la clase b, para no aplicar la técnica.

En un intento de obtener tal regla de decisión se han resuelto nuevamente los problemas de las clases a* y b. añadiendo los planos uno a uno en vez de añadirlos todos de una vez. El comportamiento mostrado en todos los casos analizados fué el siguiente: los primeros planos secantes, los que proporcionan una mejora real, suelen producir una gran disminución del salto de discontinuidad, mientras que los siguientes proporcionaban una reducción significativamente menor. Consecuentemente, comparando las reducciones del salto de discontinuidad que se producen cada vez que se añade un nuevo plano, con las reducciones anteriores se puede decidir si continuar o no añadiendo planos y, por lo tanto, evitando las situaciones b. y a*.

En un intento de responder a la pregunta de si era posible o no, identificar a priori los problemas pertenecientes a la clase b, hemos realizado un análisis de las agrupaciones (cluster analysis) en el espacio de las componentes principales. El análisis se ha realizado utilizando para cada problema la información a priori representada por los valores de las cinco variables siguientes: ratio $R1 = \sum_{i \in I} d_i / \sum_{j \in J} b_j$ (valores de la columna 3 en las tablas 1 a 6), ratio medio $R2 = f_j / \sum_{i \in J} c_{ij}$ (valores de la columna 4 de las tablas 1 a 6), $RSIZ = \text{Número de centros} / \text{Número de plantas}$, $IND = (ZHEU - ZLP) \cdot 100 / ZLP$, donde ZHEU es el valor de la función objetivo para la heurística de "pivot and complement", y ZLP es el valor óptimo de la relajación lineal. IND puede interpretarse como la estimación de una cota superior del salto de discontinuidad dual (duality gap). La última variable utilizada en el análisis fué $GR = (CUT - ZLP) \cdot 100 / ZLP$, donde ZCUT es el valor óptimo de la relajación lineal después

del primer plano secante.

La figura 2 representa gráficamente los resultados del análisis en el plano determinado por las dos primeras componentes principales, que explicaban un 65.66% de la variancia total. Aparecen en ella cuatro clases significativas. La clase 1 reúne los problemas que pueden considerarse como "outlayers", mientras que la clase 2 incluye -- con toda evidencia los problemas con altos valores de R2 e IND, es decir, problemas -- con valores grandes del ratio $f_j / \sum_{i \in I} c_{ij}$, y que, al mismo tiempo tienen un gran salto de discontinuidad. La clase 4 está determinada por los valores del parámetro R1, relación entre la demanda total y la capacidad total, y la clase 3 está definida fundamentalmente por el ratio Número de centros/Número de plantas. Sin embargo los problemas de tipo b aparecen en todas las clases y la única conclusión parcial que se puede extraer, dada la reducida dimensión de la muestra, es que los problemas de tipo b aparecen con mayor frecuencia en la clase 4, por lo cual aquellos problemas con ratios R1 mayores que la media y ratios R2 menores que la media y un salto de discontinuidad pequeño, tienden a comportarse como los de tipo b con más frecuencia que los otros.

Como consecuencia podemos decir que se requiere más experiencia computacional para extraer conclusiones definitivas, pero en cualquier caso es evidente que la técnica de generación de restricciones proporciona buenos resultados en un significativo número de casos: 53 sobre 67 (dado que para los 13 restantes la técnica no fue aplicable al no poderse calcular planos secantes para ellos), teniendo en cuenta, como hemos comentado antes, que los problemas b. se comportan como problemas a si se incluye en el procedimiento la regla de decisión descrita. La experiencia computacional adicional debería incluir también formulaciones alternativas que conduzcan a otras familias de desigualdades válidas y caras, como las derivadas de las configuraciones -- (1-k).

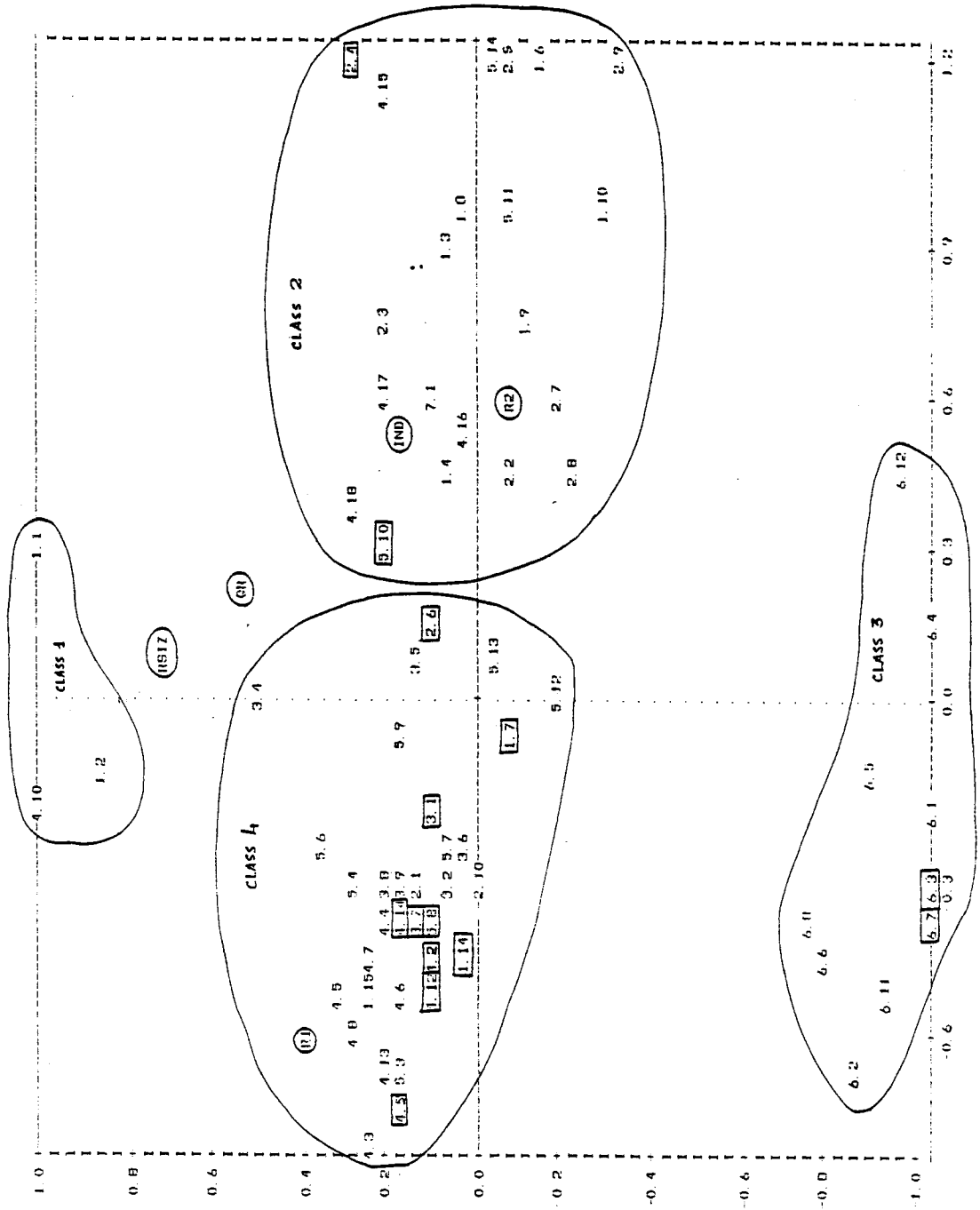


Figura 2.

Puntos multiples: punto representado / punto oculto

4.8	3.10
1.15	1.11
4.7	1.13
4.13	5.5
4.4	5.2
5.3	5.1
4.6	1.12
3.7	4.1
2.1	3.3
4.12	4.11
1.6	1.5
6.1	6.9

Z. EXPERIENCIAS CON PROBLEMAS DE PARTICIONAMIENTO.

Las experiencias computacionales de los que damos cuenta de este trabajo, tienen como punto de partida los resultados teóricos establecidos por Balas, /2/ y Balas y Padberg /4/, que resumimos a continuación.

El problema de particionamiento se puede -- plantear de la forma siguiente:

$$\text{MIN}\{cx \mid Ax=e, \quad x_j \in \{0,1\}, \quad \forall j \in N\} \quad (30)$$

donde las filas de la matriz A están asociadas a los elementos del conjunto finito $M = \{1, \dots, m\}$, y cada columna a_j de A con un subconjunto $M_j \subseteq M$, de una familia $F \subseteq \mathcal{P}(M)$ de subconjuntos de M, indexada por $N = \{1, \dots, n\}$, de manera que $a_{ij} = 1$ si el i-ésimo elemento de M pertenece al subconjunto M_j y $a_{ij} = 0$ en caso contrario, y e es un vector de componentes unidad. El problema (30) es pues el de encontrar una partición de M de peso mínimo, siendo c_j el peso asociado al subconjunto M_j .

Esta formulación corresponde a una gran variedad de problemas de scheduling que responden al planteamiento siguiente:

- Dados: (1) un conjunto finito M.
 (2) un conjunto de restricciones que define una familia F de subconjuntos de M.
 (3) un coste asociado a cada miembro de F.

Encontrar una colección de miembros de F, de coste mínimo, que constituya una partición de M.

Al problema (30) de particionamiento se le asocia el politopo de particionamiento definido por:

$$P = \text{conv}\{x \in R^n \mid Ax=e, \quad x_j \in \{0,1\}, \quad \forall j \in N\} \quad (31)$$

cuyo conjunto de vertices viene dado por:

$$\text{vert } P = P_n\{0,1\}^n \quad (32)$$

La relajación lineal del problema de particionamiento (30) es

$$LP = \{x \in R^n \mid Ax = e, \quad x \geq 0\} \quad (33)$$

Un problema estrechamente relacionado con el problema de particionamiento es el de set packing (Padberg, /18/, /19/), formulado como:

$$\text{MAX}\{cx \mid Ax \leq e, \quad x_j \in \{0,1\}, \quad \forall j \in N\} \quad (34)$$

correspondiente, en el planteamiento anterior, al caso en que en vez de una partición de peso mínimo, lo que nos interesa es una colección de subconjuntos de M, de peso máximo, tal que todo elemento de M esté contenido como máximo en un subconjunto M_j de la colección.

El politopo de set packing correspondiente está definido por:

$$\bar{P} = \text{conv}\{x \in R^n \mid Ax \leq e, \quad x_j \in \{0,1\}, \quad \forall j \in N\} \quad (35)$$

A la matriz A de incidencia de los problemas de particionamiento y de set packing, - se le puede asociar un grafo G_A , denominado grafo de intersección de A, de la manera siguiente: se trata de un grafo que tiene un nodo j por cada columna a_j de A y una arista (i,j) por cada par de nodos a_i, a_j , tales que $a_i \cdot a_j = 0$.

Un primer resultado teórico es el establecido por Padberg en /18/, donde demuestra que una desigualdad de la forma

$$\sum_{j \in V} x_j \leq 1 \quad (36)$$

es una desigualdad válida para el politopo de set packing definido por A, es decir, es satisfecha por todo vector $x \in \bar{P}$, si y solo si, V es el conjunto de nodos de un subgrafo completo de G_A , y es una cara de \bar{P} , si y sólo si V es el conjunto de nodos de un clique (subgrafo completo maximal) de G_A .

A partir de la matriz A podemos definir los siguientes subconjuntos de filas (subconjuntos de M) y de columnas de A (subconjuntos de N):

$$M_k = \{i \in M \mid a_{ik} = 1 \text{ (filas en las que la columna k tiene un 1)} \quad \forall k \in N \quad (37)$$

$$\bar{M}_k = M \setminus M_k \quad (\text{filas en las que la columna k tiene un 0})$$

$$N_i = \{k \in N \mid a_{ik} = 1\} \text{ (columnas en las que la fila i tiene un 1)} \quad \forall i \in M \quad (38)$$

$$N_i = N \setminus N_i \quad (\text{columnas en las que la fila i tiene un 0})$$

$$N_{ik} = \{j \in N_i \mid a_j \cdot a_k = 0\}, \quad i \in \bar{M}_k, \quad k \in N \quad (39)$$

conjunto de índices de las columnas a_j ortogonales a a_k tales que $a_{ij} = 1$. Puesto que $a_{ik} = 0$, dado que $i \in \bar{M}_k$, $x_k = 1$ implica que al menos una de las variables $x_j, j \in N_{ik}$ debe ser igual a 1.

Dos clases de desigualdades no homogéneas - que conducen a interesantes interpretaciones teóricas en los grafos asociados, son las definidas en la proposición siguiente:

Proposición 7.1:

Para algún $k \in N$ e $i \in \bar{M}_k$, sea $Q \subseteq N_{ik}$. Entonces $x \in LP$ satisface cualquiera de las desigualdades siguientes, si y solo si, las satisface todas:

$$x_k - \sum_{j \in Q} x_j \leq 0 \quad (40)$$

$$x_k + \sum_{j \in N_i \setminus Q} x_j \leq 1 \quad (41)$$

$$\sum_{j \in N_i \setminus \{k\}} x_j + \sum_{j \in Q} x_j \geq 1, \quad h \in M_k \quad (42)$$

(la demostración se puede encontrar en la referencia, Balas /2/). Se puede demostrar que la proposición continua siendo cierta si la condición " $x \in LP$ " se sustituye por la " x tal que $Ax=e$ ", y que la desigualdad (41) tipo set packing y la (42) de recubrimiento -- son equivalentes a (40) únicamente con respecto a los puntos $x \in R^n$ tales que $Ax = e$, - sin embargo, las desigualdades (41) de set packing no son, en general, válidas para el politopo de set packing \bar{P} asociado al politopo de particionamiento P, tal como establece la proposición:

Proposición 7.2:

La desigualdad

$$x_k + \sum_{j \in N_i \setminus Q} x_j \leq 1$$

es violada por algún $x \in \bar{P} \setminus P$ si y solo si $Q \neq N_{ik}$

Desde el punto de vista práctico de los algoritmos enumerativos, toda solución del problema de set packing proporciona una solución parcial del problema de particionamiento asociado, por lo cual el resultado que establece la proposición anterior tiene interés en lo que respecta a la eliminación de soluciones parciales.

Diremos que un conjunto $Q \subseteq N_{ik}$ es minimal si no se puede suprimir ningún elemento de Q sin invalidar la correspondiente desigualdad válida (40).

Corolario 7.1: Sea $k \in N$ si los conjuntos

$Q_i \subseteq N_{ik}, i \in \bar{M}_k$ son minimales, entonces la desigualdad (41) elimina todas las solu-

ciones parciales de la forma $x_k = x_n = 1$, con $a_k \cdot a_n = 0$ que no tienen una completación factible.

lo cual sugiere que las desigualdades (41) - pueden utilizarse como refuerzo de los tests de ortogonalidad en la enumeración implícita (Barceló, /6/). Las desigualdades (41) tienen una interesante interpretación en términos - del grafo de intersección de la matriz A.

Todas las desigualdades (41) para las que $\{k\} \cup \{n\} \in Q$ es el conjunto de nodos de un clique de G_A , son satisfechas por todo $x \in P$ y por la proposición 7.2 son precisamente aquellas para las que $Q = N_{ik}$. Las restantes desigualdades de este tipo, para las que $Q \neq N_{ik}$, no tienen interpretación en G_A , lo cual sugiere su interpretación en un supergrafo de G_A , el denominado grafo de intersección fuerte de la matriz A, denotado por $G(A)$, definido como un grafo que tiene un nodo j por cada columna a_j de A, $j \in N$ y una arista por cada par $i, j \in N$ y una arista por cada par $i, j \in N$ tal que no existe un vector $x \in \{0,1\}^n$ que satisfaga $Ax=e$, con $x_i = x_j = 1$. Obviamente G_A es un subgrafo de $G(A)$.

Las desigualdades (40), (41) pueden interpretarse en el grafo de intersección fuerte $G(A)$ de la manera siguiente:

Proposición 7.3:

(1) La desigualdad

$$\sum_{j \in V} x_j \leq 1 \quad (43)$$

es satisfecha por todo $x \in P$ si y solo si V es el conjunto de nodos de un subgrafo completo de $G(A)$.

(2) Supongamos que para cada $j \in V$ existe un $x \in P$ tal que $x_j = 1$, y que (43) es satisfecha por todo $x \in P$. Entonces la desigualdad (43) es maximal si y solo si V es el conjunto de nodos de un clique de $G(A)$.

(La demostración de esta proporción se encuentra, como las anteriores, en la referencia: Balas, /2/).

A partir del conjunto de proposiciones teóricas que acabamos de resumir, nuestro tra-

bajo se centró en el estudio de la posibilidad de utilizarlas computacionalmente, posibilidad acrecentada por el hecho de que, según las estructuras de datos que se utilizan para almacenar en forma compacta los elementos de la matriz A, (Greenberg, /12/), habitualmente de poca densidad para los problemas reales de particionamiento, se puede disponer directamente de los conjuntos M_k , \bar{M}_k , N_i y \bar{N}_i , sin ningún esfuerzo computacional extra y, de una forma directa se pueden obtener los conjuntos N_{ik} cuando se necesitan. Consecuentemente nuestro esfuerzo se dirigió, según las ideas expuestas al principio del artículo, en la dirección de, dada una solución \bar{x} de la relajación lineal (33) del problema de particionamiento (31), encontrar una desigualdad válida para P de la forma (30) o (31) violada por \bar{x} , para de acuerdo con el método de identificación de restricciones, añadirse a la formulación de (30) y reoptimizar. Es importante resaltar que las restricciones de la forma (31) mantienen la estructura del problema y pueden ser utilizadas también en combinación con algoritmos enumerativos.

El procedimiento desarrollado es el siguiente:

- a) Sea $S_1 \subset N$ el conjunto de variables no enteras de la solución \bar{x} a la relajación lineal (33), y S_2 el de las variables cuyo valor es 1, es decir:

Dada $\text{MIN}\{cx \mid Ax = e, x \geq 0\}$

Sea $x_i = y_{i0} - \sum_{j \in R} y_{ij} x_j$

la tabla óptima del simplex, entonces

$$S_1 = \{i \in N \mid 0 < y_{i0} < 1\} \quad (44)$$

y

$$S_2 = \{i \in N \mid y_{i0} = 1\} \quad (45)$$

- b) Elegir una variable x_k , $k \in S_1$.

$$\text{criterio: } x_k = \text{MAX}_{i \in S_1} \{y_{i0}\} \quad (46)$$

en caso de empate elegir aquella con mayor $|M_k|$.

- c) Definir $\bar{M}_k = M \setminus M_k$ (filas en las que la columna k tiene elementos nulos).

- d) Elegir un elemento $i \in \bar{M}_k$ (fila).
- d1) Si $S_2 = \emptyset$ entonces elegir $i \in \bar{M}_k$ tal que $|N_i|$ sea máximo.
- d2) Si $S_2 = \emptyset$ sea $l \in S_2$, entonces $x_k + x_l > 1$. Elegir i tal que
- $$i \in \bar{M}_k \cap \bar{M}_l \quad (47)$$
- e) Definir $N_i = \{j \in N \mid a_{ij} = 1\}$, columnas en las que la fila i tiene elementos no nulos.
- f) Definir $N_{ik} = \{j \in N_i \mid a_{jk} = 0\}$, columnas de N ortogonales a k .
- g) Elegir $Q \subseteq N_{ik}$
Si $S_2 \neq \emptyset$
- g1) $N_{ik} = \emptyset \Rightarrow Q = \emptyset$
Añadir la restricción $x_k + \sum_{j \in N_i} x_j \leq 1$
- g2) $N_{ik} \neq \emptyset$, $N_{ik} = \{j\}$, $|N_{ik}| = 1$, $|a_k| + |a_j| \leq m$, $j \in S_2$
hacer $Q = \{j\}$ $j \in S_2$
añadir $x_k + \sum_{j \in N_i \setminus Q} x_j \leq 1$
- g3) $N_{ik} \neq \emptyset$, $|N_{ik}| > 1$
hacer $Q = N_{ik}$
añadir $x_k + \sum_{j \in N_i \setminus Q} x_j \leq 1$

Si $S_2 = \emptyset$ hacer $Q = N_{ik}$ y proceder como en -- (g3), si la restricción no es violada por \bar{x} elegir otra i u otra k .

El procedimiento propuesto establece una -- heurística que genera de una manera sistemática desigualdades válidas para P , violadas por la solución continua en curso \bar{x} , que eventualmente son caras de P .

La justificación del procedimiento se puede establecer de la forma siguiente:

En general el conjunto $S_1 \subseteq N$ de variables no enteras de la solución \bar{x} a la relajación lineal (33) permite determinar al menos un -- subgrafo completo de $G(A)$, ya que si $x_i \in S_1$ y x_j tiene un 1 en la fila i , de $a_i \cdot \bar{x} = 1$

se deduce que $\exists k \in N$: $x_k \in S_1$ que también tiene un 1 en la fila i por lo que $a_i a_k \neq 0$.

El criterio (46) de selección de la variable x_k , identifica una variable x_k en un mayor $|M_k|$, por lo que, si $S \neq \emptyset$, la fila i que satisface el criterio (47) tendrá en general, el mayor número de columnas no ortogonales a k y posiblemente, ortogonales a l , puesto que $a_l \cdot a_i = 0$, $\forall i \in S_1$ y $\forall l \in S_2$, por lo que pertenecerán a S_1 y entonces

$$x_n + \sum_{j \in N_i \setminus Q} x_j \leq 1$$

quedará violada por \bar{x} , puesto que, en general, en estas condiciones N_{ik} será de dimensión mínima, al haber muchas columnas de N_i no ortogonales a a_k .

8. EXPERIENCIA COMPUTACIONAL.

Hasta el momento la heurística propuesta se ha probado en un reducido número de casos, cuyos resultados han sido lo suficientemente prometedores como para continuar. Citamos a continuación, a falta de un mayor número de experiencias que permitan una completa sistematización de los resultados, algunos de los ejemplos más relevantes.

Problema 1

Columna	1	2	3	4	5	6	7	8	9	10
C_j	5	4	3	2	2	3	1	1	1	0
Fila 1	1	0	0	0	0	1	1	0	1	0
2	1	1	0	0	0	0	0	0	0	1
3	0	1	1	0	0	1	1	1	0	0
4	0	1	1	1	0	0	1	0	0	0
5	0	0	1	1	1	1	0	1	1	0

La solución de la relajación lineal (33) es:

$$x_7 + x_3 + x_4 + x_6 + x_8 + x_9 \leq 1 \tag{48}$$

$$x_4 = 1/3, \quad x_7 = 2/3, \quad x_8 = 1/3, \quad x_9 = 1/3, \quad x_{10} = 1$$

$$x_1 = x_2 = x_3 = x_5 = x_6 = 0$$

$$S_1 = \{4, 7, 8, 9\}, \quad S_2 = \{10\}$$

por lo que, según (46), $k=7$, la variable elegida es x_7 , luego

$$\bar{M}_7 = \{2, 5\}, \quad \bar{L} = 10, \quad \bar{M}_{10} = \{1, 3, 4, 5\} \text{ y } \bar{M}_7 \cap \bar{M}_{10} = \{5\}$$

y por lo tanto, por (47), $i=5$,

$$N_S = \{3, 4, 5, 6, 8, 9\} \text{ y } N_{S7} = \{5\} \text{ y haciendo}$$

$Q = N_{S7}$ queda:

desigualdad válida del tipo (41), que en este caso satisface las condiciones de la proposición 7.1 y además $\{3, 4, 6, 7, 8, 9\}$ forma un clique de $G(A)$, y es violada por \bar{x} .

Añadiendo (48) al conjunto de restricciones del problema y reoptimizando se obtiene

$$x_5 = x_7 = x_{10} = 1$$

$$x_1 = x_2 = x_3 = x_4 = x_6 = x_8 = x_9 = 0$$

que es la solución definitiva.

Problema 2

Columna	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
C_j	5	4	3	2	2	2	3	1	2	2	1	1	1	0	0
Fila 1	1	0	0	0	0	1	1	1	1	0	0	0	1	1	0
2	1	1	0	0	0	0	0	0	0	1	1	0	0	0	1
3	0	1	1	0	0	1	1	1	0	0	0	1	0	0	0
4	0	1	1	1	0	0	0	1	1	0	1	0	0	0	0
5	0	0	1	1	1	0	1	0	0	1	0	1	1	0	0

cuya solución para la relajación lineal (33) es

$$x_8 = x_{11} = x_{12} = x_{13} = x_{15} = 1/2$$

$$x_1 = x_2 = x_3 = x_4 = x_5 = x_6 = x_7 = x_9 = x_{10} = x_{14} = 0$$

es decir:

$$S_1 = \{8, 11, 12, 13, 15\}$$

$$S_2 = \emptyset$$

de donde $k=8$ y $\bar{M}_8 = \{2, 5\}$ y como $|N_2| = 5$ y $|N_5| = 7$ elegimos $i=5$ con $N_5 = \{3, 4, 5, 7, 10, 12, 13\}$ y $N_{58} = \{5, 10\}$. Haciendo $Q = \{5, 10\}$ resulta

$$x_8 + x_3 + x_4 + x_7 + x_{12} + x_{13} \leq 1$$

que es violada por \bar{x} , y nuevamente $(8, 3, 4, 7, 12, 13)$ es un clique de $G(A)$.

La adición de la restricción proporciona, después de reoptimizar

$$x_{11} = x_{12} = x_{14} = 1$$

$$x_1 = x_2 = x_3 = x_4 = x_5 = x_6 = x_7 = x_8 = x_9 = x_{10} = x_{13} = x_{15} = 0$$

que es óptima

$$S_1 = \{7, 9, 14\}$$

$$S_2 = \{4\}$$

y por tanto

$$K = 7, \bar{M}_7 = \{1, 3, 5\}$$

$$l = 4, \bar{M}_4 = \{1, 4, 5, 6\}, \text{ y } \bar{M}_7 \cap \bar{M}_4 = \{5\} \Rightarrow i = 5$$

$$N_5 = \{2, 3, 6, 9, 13, 14, 15\}$$

$$N_{57} = \emptyset \Rightarrow Q = \emptyset$$

y la desigualdad es

$$x_7 + x_2 + x_3 + x_6 + x_9 + x_{13} + x_{14} + x_{15} \leq 1$$

violada por \bar{x} , pero en este caso $(2, 3, 6, 7, 9, 13, 14, 15)$ no es un clique de $G(A)$, el clique correspondiente está definido por $(2, 3, 6, 7, 9, 12, 13, 14, 15)$, sin embargo añadiéndola la reoptimización da como resultado

$$x_3 = x_{11} = 1$$

$$x_1 = x_2 = x_4 = x_5 = x_6 = x_7 = x_8 = x_9 = x_{10} = x_{12} = x_{13} = x_{14} = x_{15} = 0$$

que es la solución óptima.

De forma similar se ha aplicado este proce-

Problema 3

Columna	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	15	20	15	8	9	18	5	10	16	10	8	4	10	6	10
Fila	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
	2	1	1	0	0	0	1	1	1	0	0	0	0	0	0
	3	0	0	0	1	1	0	0	0	1	1	0	0	0	0
	4	0	1	0	0	1	1	1	0	0	1	1	1	1	0
	5	0	1	1	0	0	1	0	0	1	0	0	1	1	1
	6	0	0	0	0	0	1	0	1	1	1	1	1	0	1

que proporciona para (33) la solución

$$x_7 = x_9 = x_{14} = 1/2, \quad x_4 = 1$$

$$x_1 = x_2 = x_3 = x_5 = x_6 = x_8 = x_{10} = x_{11} = x_{12} = x_{13} = x_{15} = 0$$

de donde

dimiento a la resolución de tres series de 10 problemas cada una, de dimensiones, --- 10x30, 15x40 y 15x50 respectivamente, con los resultados siguientes:

Serie	Número medio de planos añadidos	Máximo número de planos añadidos
10x30	3	5
15x40	4	6
15x50	5	7

Para problemas de mayores dimensiones, este planteamiento tropieza con los inconvenientes típicos de la degeneración del simplex para problemas de particionamiento, la continuación de estas experiencias será posible cuando dispongamos de un código estandar de programación lineal más evolucionado, capaz de tratar algunos de los típicos inconvenientes numéricos que aparecen en estos casos.

9. REFERENCIAS BIBLIOGRAFICAS.

- /1/ E. BALAS: "Facets of the Knapsack Polytope". *Math. Prog.* 8, pp.146-164 (1975).
- /2/ E. BALAS: "Some valid inequalities for the set partitioning problem". *Annals of Discrete Mathematics*, 1, 13-47 (1977).
- /3/ E. BALAS AND C.H. MARTIN: "Pivot and Complement: A Heuristic for 0-1 Programming". *Management Science*, 26, pp. 86-96 (1980).
- /4/ E. BALAS AND M. PARDBERG: "Set Partitioning: A Survey in: N. Christofides, A. Mingozzi, P. Toth, and C. Sandi (ed.): *Combinatorial Optimization*". John Wiley, (1979).
- /5/ E. BALAS AND E. ZEMEL: "Facets of the knapsack polytope from minimal covers". *Siam Journal on Applied Mathematics*, 34, pp. 119-148 (1978).
- /6/ J. BARCELÓ: "An integer programming approach to Bus Crew Scheduling". 26 th. - International Meeting of TIMS, Copenhagen, (1984).
- /7/ H. CROWDER, E.L. JOHNSON AND M.W. PADBERG: "Solving Large-Scale Zero-One linear Programming problems". *Operations Research*, 31, pp. 803-834 (1983).
- /8/ A.M. GEOFFRION AND R. McBRIDE: "Lagrangian Relaxation Applied to Capacitated Facility Location Problems". *AIIE Transactions*, 10, pp. 40-47 (1978).
- /9/ A.M. GEOFRION AND R. MARSTEN: "Integer Programming Algorithms: A Framework and State-of-the Art-Survey". *Management Science* 18, pp. 465-491 (1972).
- /10/ R.E. GOMORY: "Outline of an Algorithm for Integer Solutions to Linear Problems" *Bull. Amer. Math.Soc.*, 64 pp. 275-278. (1958).
- /11/ R.E. GOMORY: "Solving Linear Programs in Integers in: R. Bellmann and M. Hall (eds): *Combinatorial Analysis*". American Mathematical Society, Providence, pp. 211-216. (1960).
- /12/ H.J. GREENBERG: "A tutorial on Matricial Packing in: H.J. Greenberg (ed.): *Design and Implementation of Optimization Software*". Sijthoff and Noordhoff, (1978).
- /13/ M. GRÖTSCHEL: "Approachs to Hard Combinatorial Optimization Problems": in B. Korte (ed.): *Modern Applied Mathematics: Optimization and Operations Research*, North-Holland, Amsterdam, pp. 437-515. (1982).
- /14/ M. GRÖTSCHEL: "Developments in Combinatorial Optimization". Preprint No. 26, Universität Augsburg, Mathematisches Institut, (1984).
- /15/ M. GRÖTSCHEL, L. LOVASZ AND A. SCHRIJVER "The Ellipsoid Method and its consequences in Combinatorial Optimization". *Combinatorica*, 1, 169-197 (1981).
- /16/ M. GRÖTSCHEL AND M. PADBERG: "Polyhedral Aspects of the Traveling Salesman Problem: I Theory, II Computation" Reports noms. 82212-OR and 82214-OR, Institut für Ökonometrie und Operations Research", Rheinische Friedrich-Wilhelms Universität Bonn.
- /17/ M. GUIGNARD AND K. SPIELBERG: "A Direct Dual Method for the Mixed Plant Location Problem with Some Side Constraints" *Math. Prog.*, 17, pp. 198-228 (1979).

- /18/ M. PADBERG: "On the Facial Structure of the Packing Polyhedra". *Mat. Prog.*, 5, pp. 199-215. (1973).
- /19/ M. PADBERG: "Covering, Packing and Knapsack Problems". *Annals of Discrete Mathematics*, 4, pp. 265-287 (1979).
- /20/ M. PADBERG: "A Note on Zero-One Programming" *Operations Research*, 23, 833-837, (1975).
- /21/ M. PADBERG AND M.R. RAO: "Odd Minimum Cut -Sets and b-Matchings". *Mathematics of Operations Research*, 1, 67-80 (1982).
- /22/ M. PADBERG, T.J. VAN ROY AND L.A. WOLSEY: "Valid Inequalities for Fixed Charge Problems". *Core Discussion Paper*, No. 8232, Université Catholique de Louvain, (1982).
- /23/ W. PULLEYBLANK: "Polyhedral Combinatorics" in: A. Bachem, M. Grötschel and B. Korte (eds.): *Mathematical Programming: The State of the Art*, Springer-Verlag, Berlin, 1983.
- /24/ L. SCHRAGE: "Implicit Representation of Variable Upper Bounds in Linear Programming". *Math. Prog. Study*, 4, pp. 118-132 (1975).
- /25/ L. SCHRAGE: "Implicit Representation of Generalised Variable Upper Bounds in Linear Programming". *Math. Prog.*, 14, pp. 11-20 (1978).
- /26/ K. SPIELBERG: "Plant Location with Generalised Search Origin". *Management Science*, 16, pp. 165-178. (1969).
- /27/ T.J. VAN ROY: "A Cross-Descomposition Algorithm for Capacitated Facility Location Working Paper", Afdeling Industrieel Beleid, Katholieke Universiteit Leuven, (1983).
- /28/ H. WEYL: "Elementare Theorie der Convexen Polyeder". *Comentarii Mathematici Helvetici*, 7, pp. 290-306 (1935).