Un Algoritmo de Subgradiente y un Filtro Adicional para la Resolución del Subproblema Entero en la Partición de Benders

J. BARCELÓ, L. OLIVELLA

El método de partición de Benders es particularmente útil para resolver modelos matemáticos del tipo de "multicommodity flows" o modelos econométricos del tipo de planificación descentralizada, sin embargo, en algunos casos, el subproblema entero generado por la descomposición dual es resuelto deficientemente por los procedimientos habituales de enumeración debido a su estructura matemática, carente de función objetivo e incluyendo una variable no restringida.

En nuestro trabajo distinguimos dos casos: uno con restricciones derivadas únicamente de los puntos extremos del politopo dual y otro que incluye además restricciones procedentes de los rayos extremos. En el primer caso, proponemos un algoritmo basado en el método del subgradiente y en el segundo una variante del algoritmo del filtro de Balas con un filtro parcial calculado a partir de una restricción compuesta.

1. INTRODUCTION: EL METODO DE DESCOMPOSICION DUAL DE BENDERS

Dado el problema de programación lineal entera mixta, /1 /, que denominaremos problema (P)

$$[\text{MIN}]\ Z = c\,x + d\,y \qquad (P)$$

sometida a las condiciones

$$\begin{aligned} &A\,x + B\,y \geq b \\ &x,\ y \geq 0,\ x \text{ entero} \end{aligned} \qquad (P)$$

donde $c \in R^{n_1}$, $d \in R^{n_2}$, $A(m,n_1)$, $B(m,n_2)$, $b \in R^m$ y en particular x es una variable 0-1, es decir, $x_i \in \{0,1\}$, $\forall i$, $i=1,\ldots,n_1$, que es la situación que consideramos en el presente trabajo, dado que todo el problema entero en variables acotadas, puede formularse como un problema de programación 0-1, a partir de la descomposición binaria de las mismas. Si en el problema (P) fijamos un vector $\bar{x}$ a valores 0-1: $x=\bar{x}$, el problema queda reducido al subproblema de programación lineal:

$$Z^*(\bar{x}) = c\bar{x} + [\text{MIN}]\ dy$$

tal que:

$$\begin{aligned} &By \geq b - A\bar{x} \\ &y \geq 0 \end{aligned} \qquad (LP)$$

cuyo dual es:

$$u^*(\bar{x}) = c\,\bar{x} + [\text{MAX}]\ u\,(b - A\,\bar{x})$$

tal que:

$$\begin{aligned} &u\,b \leq d \\ &u \geq 0 \end{aligned} \qquad (D)$$

en estas condiciones el problema original se puede reformular como:

$$\begin{aligned} &Z^* = \text{MIN}\{\ c\,x + \text{MAX}\ \{u\ (b - Ax) \mid u\,B \leq d,\ u \geq 0\}\} \\ &x \geq 0 \\ &x \text{ entero } (\text{ó } x_i \in \{0,1\},\ \forall_i,\ i=1,\ldots n_i) \end{aligned}$$

o, de manera equivalente:

$$\begin{aligned} &Z^* = \text{MIN}\ \{cx + \text{MAX}\{\ u(b - Ax) \mid u \in T,\ v(b-Ax) \leq 0,\ v \in Q\}\} \\ &x \geq 0 \\ &x \text{ entero } (\text{ó } x_i \in \{0,1\},\ \forall_i,\ i=1,\ldots n_1) \end{aligned} \qquad (P')$$

siendo T el conjunto de todos los puntos extremos del politopo determinado por la región de soluciones posibles del problema dual (D) y Q el conjunto de todos los rayos extremos del mismo politopo.

Como es bien sabido, el algoritmo de Benders propone un procedimiento iterativo de resolución del problema (P') en el que en cada iteración se utilizan subconjuntos de puntos y rayos extremos del politopo dual, calculados

- J. Barceló y L. Olivella, Departamento de Investigación operativa y estadística. Facultad de Informática, Universidad Politécnica de Barcelona.
- Una primera versión de este artículo fue presentada en el congreso del EURO IV, Cambridge, Inglaterra, Julio 1980.

en las iteraciones precedentes, para cons-truir el subproblema entero:

$$
\begin{aligned}
z^* = [\text{MIN}]\ & z_0 \\
& z_0 \geq cx + u\,(b - Ax) \quad u \in T_k \\
& 0 \geq v\,(b - Ax) \quad v \in Q_k \\
& x \geq 0 \text{ y entero } (\text{ó } x_i \in \{0,1\}\ ,\forall_i, \\
& \quad i=1,\dots n_1)
\end{aligned}
$$

siendo $T_k \subseteq T$ y $Q_k \subseteq Q$ los subconjuntos de puntos y rayos extremos respectivamente, calculados en las k iteracciones anteriores.

El objetivo de nuestro trabajo consiste en la búsqueda de un algoritmo eficiente para resolver este subproblema, en aquellos contextos en que tiene que intentarse su solución directamente sin poder introducir alguna modificación simplificadora.

2. ALGUNOS EJEMPLOS DE UTILIZACION DE LA TECNICA DE DESCOMPOSICION DE BENDERS

2.1. CASO A: "MULTICOMMODITY Y DISTRIBUTION SYSTEM" (Geoffrion y Graves 1974),/2/

Se trata de uno de los modelos más completos utilizados en los últimos años para el estudio de este tipo de sistemas, el trabajo de Geoffrion y Graves constituye por otra parte, desde entonces, una referencia obligada en todo análisis de la utilización del método de Benders. En síntesis, el modelo propuesto es el siguiente:

Dados: i productos, j plantas de producción, k posibles emplazamientos de centros de distribución y l zonas de demanda.

Si : S_{ij} es la capacidad de producción del producto i en la planta j.

D_{il} es la demanda del producto i en la zona l

$\underline{V}_k$ y $\bar{V}_k$ son respectivamente el mínimo y el máximo "throughput" total anual permitido para un centro de distribución en el emplazamiento k.

f_k costes fijos de posesión y operación de un centro de distribución en el emplazamiento k.

v_k coste unitario variable del "throughput" para un centro de distribución ubicado en k.

c_{ijkl} coste medio unitario de producir y enviar el producto i en la planta j hasta la zona l a través del centro de distribución k.

x_{ijkl} cantidad del producto i enviada desde la planta j hasta la zona l a través del centro de distribución en k.

y_{kl} Variable 0-1 que valdrá 1 si el centro de distribución k sirve a los clientes de la zona l, y será 0 en caso contrario.

z_k Variable 0-1 que será 1 si un centro de distribución opera en el emplazamiento k y será 0 en caso contrario.

El modelo se puede escribir como el programa lineal entero mixto siguiente:

$$\underset{x\geq 0,y,z\in\{0,1\}}{[\text{MIN}]} \sum_{i,j,k,l} c_{ijkl}\, x_{ijkl} + \sum_k \left[f_k\, z_k + v_k \sum_{i,l} D_{il}\, y_{kl} \right] \qquad (1)$$

sometida a las restricciones:

$$\sum_{k,l} x_{ijkl} \leq S_{ij} \quad \forall\,(i,j) \qquad (2)$$

$$\sum_{j} x_{ijkl} = D_{il}\, y_{kl} \quad \forall\,(i,k,l) \qquad (3)$$

$$\sum_{k} y_{kl} = 1 \quad \forall\, l \qquad (4)$$

$$\underline{V}_k\, z_k \leq \sum_{i,l} D_{il}\, y_{kl} \leq \bar{V}_k\, z_k \ \forall\, k \qquad (5)$$

y un conjunto (6) de restricciones de configuración lineal sobre y y/o Z entre las que pueden figurar, por ejemplo:

- o Cotas superiores y/o inferiores sobre el número total de centros de distribución permitidos.
- o Especificación de subconjuntos de centros de distribución entre los cuales como máximo uno, al menos uno, exactamente dos, etc. han de estar abiertos.
- o Relaciones de precedencia entre los centros de distribución abiertos (no A a me

nos que B, etc.)

- Restricciones obligatorias de áreas de servicio; si el centro de distribución A está abierto debe servir al cliente de la zona B, etc.

- Restricciones de capacidad más detalladas sobre las dimensiones de un centro de distribución de lo que

$$\underline{V}_k\, Z_k \leq \sum_{i,l} D_{il}\, Y_{kl} \leq \bar{V}_k\, Z_k, \quad \forall_k, \quad \text{permite}$$

tales como ponderar las capacidades de consumo de cada producto o escribir restricciones por separado para cada producto o para subconjuntos de productos.

- Restricciones sobre la capacidad conjunta de varios centros de distribución si comparten recursos comunes o instalaciones.

- Restricciones de servicio al cliente, como

$$\left(\sum_{k,l} t_{ikl}\, D_{i\ell}\, Y_{kl}\right) \Big/ \left(\sum_{l} D_{il}\right) \leq T_i$$

donde t_{ikl} es el tiempo medio para hacer una entrega del producto i a la zona l del cliente, después de recibir un pedido en el centro de distribución k, y T_i es una cota deseada del plazo medio de entrega para el producto i.

La aplicación del método de descomposición de Benders permite resolver el problema mediante un algoritmo iterativo, en el que en cada iteración, una vez fijadas las variables binarias, el subproblema continuo en x puede separarse en tantos problemas clásicos de transporte como productos, de manera que el problema de transporte del i-ésimo producto es de la forma

$$\begin{aligned}
&[\text{MIN}] \sum_{j,l} C_{ij\bar{k}(l)l} \\
&\sum_{l} X_{ij\bar{k}(l)l} \leq S_{ij} \quad , \forall_j \qquad (7)\\
&\sum_{j} X_{ij\bar{k}(l)l} = D_{il} \quad \forall_\ell \\
&X_{ij\bar{k}(l)l} \geq 0 \quad \forall_{j,\ell}
\end{aligned}$$

donde $\bar{k}$ (l) se define para cada k, como el índice k para el que $Y_{kl}=1$ (por las restricciones (4), $\bar{k}(l)$ es único para cada l).

La fijación de las variables binarias se realiza, en cada iteración del método de Benders, resolviendo el subproblema entero:

$$\underset{Y,Z\in\{0,1\},\, Y_o}{[\text{MIN}]} \quad Y_o + \sum_{k} \left[f_k\, Z_k + V_k \sum_{i,l} D_{il}\, Y_{kl} \right] \qquad (8)$$

sometido a las restricciones (4), (5), (6) y

$$Y_o + \sum_{i,k,l} \Pi^h_{ikl}\, D_{il}\, Y_{kl} \geq -\sum_{i,j} u^h_{ij} S_{ij}, \quad h=1,\ldots,H \qquad (8a)$$

donde Π^h y u^h son los vectores de variables duales de las soluciones del problema global continuo en la iteración h, correspondiendo Π a las restricciones (2) y u a las restricciones (3) del problema primal.

2.2. CASO B: "UN PROCEDIMIENTO ITERATIVO DE PLANIFICACION". (Vegara 1976) /3/

Se trata del modelo: hallar $x_1, x_2, \ldots, x_n$ e y tales que

$$\begin{aligned}
\text{MIN } Z \;=\; & c_1 x_1 + \ldots + c_n x_n + c_o y \\
& A_1 x_1 + \ldots + A_n x_n + A_o y = b \\
& B_1 x_1 \qquad\qquad\qquad = b_1 \qquad (9)\\
& \quad B_2 x_2 \qquad\qquad = b_2 \\
& \qquad \vdots \qquad\qquad\quad \vdots \\
& \qquad\quad B_n x_n \qquad = b_n \\
& x_1, x_2, \ldots, x_n \geq 0, \; y \in \{0,1\}
\end{aligned}$$

donde A_j (m,n_j), B_j (m_j,n_j), $b\in R^m$, $b_j\in R^{m_j}$, $c_j\in R^{n_j}$, $c_o\in R^{n_o}$,
el modelo representa un sistema productivo consistente en un centro de planificación y n unidades productivas que deben satisfacer una demanda global disponiendo de una serie de recursos comunes, cada unidad productiva dispone de una capacidad de producción ya instalada y se halla sujeta a un cierto número de restricciones específicas; el centro de planificación por su parte dispone asimismo de un cierto número de decisiones relativas a la creación de nuevas capacidades de producción, las variables continuas x_j representan los niveles de las actividades controladas por las unidades productivas. El vector y representa las decisiones controladas por el --

centro de planificación.

Utilizando un método de descomposición basado en una directiva de precios, /4/, /6/, el problema (8), tiene el siguiente "problema extremo" asociado; hallar λ_{jk} e y tales que:

$$[\text{MIN}]\ z = \sum_j \sum_k c_{jk} \lambda_{jk} + c_o y$$

$$\sum_j \sum_k P_{jk} \lambda_{jk} + A_o y = b \quad (10)$$

$$\sum_k \lambda_{jk} = 1 \quad \forall_j$$

$$\lambda_{jk} \geq 0 \ , \ \forall j,k$$

Resolviendo este problema por medio del procedimiento de Benders aparece en cada iteración el siguiente subproblema entero: Hallar Z e y tales que

$$[\text{MIN}]\ z$$
$$z \geq c_o y + \bar{\Pi}(b - A_o y) + \sum_{j=1}^{n} \bar{\Pi}_j \qquad (11)$$
$$y \in \{0,1\}$$

Donde es la solución óptima del subproblema continuo

$$[\text{MAX}]\ w = \Pi (b - A_o \bar{y}) + \sum_{j=i}^{n} \Pi_j$$
$$\Pi_j \leq [\text{MIN}]\ (c_j - \Pi_j A_j)\ x_j \qquad (12)$$
$$\left\{ \begin{array}{l} B_j \ x_j = b_j \\ x_j \geq 0 \end{array} \right\} \ \forall j$$

(e $\bar{y}$ la solución óptima de (10) en la iteración anterior).

3. RESOLUCION DE LOS SUBPROBLEMAS ENTEROS

3.1. CASO A.

La estrategia de Geoffrion y Graves consiste en no intentar resolver el problema (8) hasta obtener el óptimo, sino detenerse tan pronto como se haya obtenido una solución - posible con un valor menor que una cota UB menos una tolerancia ε. El problema se convierte entonces en : buscar Y, $Z \in \{0,1\}$ e Y_0 que satisfagan las restricciones (4), (5), (6), (8a) y

$$\sum_k \left[f_k z_k + V_k \sum_{i,l} D_{il} \quad Y_{kl} \right] + Y_0 \leq UB - \epsilon$$

y eliminando Y_0: encontrar, Y, $Z \in \{0,1\}$ que satisfagan (4), (5), (6) y

$$\sum_k f_k z_k \quad V_k \sum_{i,l} D_{il} Y_{kl} - \sum_{i,j} u^h_{ij} S_{ij} - \sum_{i,k,l} \Pi_{jikl} D_{il} Y_{kl} \leq UB - \varepsilon, h=1,\ldots,H \qquad (13)$$

Esta formulación del problema (8) permite introducir cualquier función lineal objetivo Φ (Y,Z) apta para producir buenas soluciones - posibles, con lo que esta etapa de la iteración se reduce a encontrar una solución posible del problema:

$$[\text{MIN}]\ \Phi(Y,Z)$$
$$Y,Z \in \{0,1\}$$

sometida a; (4), (5), (6), y (13).

OBSERVACION: La transformación ha convertido el subproblema (8) en un problema entero puro, que puede resolverse mediante cualquiera de los algoritmos estándar.

3.2. CASO B.

El subproblema (11) es el típico subproblema entero generado por la partición de Benders, se trata de un problema entero no puro, ya - que está presente la variable Z, que es una variable continua no restringida en signo.
La obtención de soluciones para este subproblema puede entrañar en algunos casos serias dificultades ya que:

- Los procedimientos habituales de enumeración son muy ineficaces en este caso, pudiendo llegarse en algunas situaciones a una enumeración casi total. (Véase, comentarios de Salkin, 1975, /6/ y de Geoffrion y Marsten, 1973, /7/, en el caso - del FMPS, nivel 5).

- Es difícil de manejar con los códigos estándar para programación matemática.

4. ALGORITMOS USUALES PARA LA RESOLUCION DEL SUBPROBLEMA ENTERO

Las dificultades señaladas en el apartado 3. 2. nos llevaron a centrar nuestro interés en los métodos para resolver el problema (11), en un intento de encontrar un camino más eficiente. El análisis de la bibliografía a nuestro alcance nos llevó a la conclusión de que los informes contenidos en /6/ y /7/ eran bastante representativos y, en general el subproblema (11) siempre se había intentado resolver mediante variantes de procedimientos enumerativos, apareciendo entre ellos el algoritmo del filtro de Balas, /8/, como uno de los más eficaces, por lo cual, en una primera etapa nos dedicamos a profundizar en el estudio de dicho algoritmo.

4.1. Breve resumen del algoritmo del filtro de Balas.

Formulado el problema en la iteración k-esima de la forma estándar siguiente:

$$[\text{MIN}]\ Z \qquad \begin{aligned} &Z \geq ub + (c-uA)\,x\ , && u \in T_k \\ &0 \geq v(b-Ax) &&, \ v \in Q_k \\ &x \in \{0,1\} \end{aligned} \qquad (PK)$$

el algoritmo de Balas empieza por relajar las restricciones de integridad del problema (PK), convirtiéndolo en el programa lineal ordinario (con la variable Z no restringida en signo):

$$[\text{MIN}]\ Z \qquad \begin{aligned} &Z \geq ub + (c-uA)\,x && , \ u \in T_k \\ &0 \geq v(b-Ax) && , \ v \in Q_k \\ &0 \leq x \leq 1 \end{aligned} \qquad (PKR)$$

la solución del dual del problema relajado (PKR) proporciona el vector de multiplicadores de las restricciones del problema (PK), que permiten construir la restricción de agregación, o restricción filtro, que nos conduce al problema filtro (FP):

$$[\text{MIN}]\ Z \qquad \begin{aligned} &a_o Z + ax \geq b_o \\ &x \in \{0,1\} \end{aligned} \qquad (FP)$$

construído de manera que una oportuna reordenación de variables $a_i \leq o, \forall i$, defina un indexado óptimo tal que $a_{i_1} > a_{i_2} \Rightarrow i_1 < i_2$.
El algoritmo se basa en un procedimiento de ramificación que genera soluciones posibles para el problema filtro (FP), tales que:

a) La sucesión de valores de Z sea no decreciente.

b) $\forall x^s$ asociado a una pseudosolución, se define una solución posible mediante la relación

$$Z^s = b_o - ax^s$$

y una vez alcanzado un Z^s quedan enumeradas explícita e implícitamente todas las soluciones tales que $Z < Z^s$.

A cada solución posible para (FP) se le aplican una serie de tests para comprobar si es también solución posible de (PK), en cuyo caso se compara con la mejor solución obtenida hasta el momento y otra batería de test complementarios para comprobar si se continúa el proceso de enumeración desde dicha pseudo solución y en tal caso qué variable se elige como variable de ramificación.

4.2. Experiencias con el filtro de Balas.

Un estudio empírico de la aplicación del algoritmo de filtro a una serie de problemas test nos llevó a las siguientes conclusiones:

1) en el caso de un problema que únicamente tenga restricciones del primer tipo, es decir, restricciones de la forma

$$Z \geq ub + (c-uA)\,x\ , \ u \in T_k$$

debido a la relación entre los conjuntos de soluciones posibles del problema original F(PK y del problema filtro derivado F(FP): $\quad F(PK) \subset F(FP)$
podemos llegar, en el caso de conjuntos muy diferentes, a una enumeración completa.

2) Si tenemos ambos tipos de restricciones, el filtro nos obliga a generar a priori muchas soluciones no posibles ya que, únicamente las soluciones de $0 \geq v(b-Ax)$, $v \in Q_k$, pueden ser soluciones posibles de (P).

3) Incluso en los casos en que tenemos ambos tipos de restricciones pueden darse situaciones en las que el filtro se comporta como una restricción de sustitución ("surrogate constraint") muy relajada con la consiguiente pérdida de información sobre las soluciones posibles. Así, por ejemplo, en el caso del problema

$$[\text{MIN}]\ z$$
$$z \geq 15-7x_1 - 14x_2 + 6x_3 + 12x_4$$
$$z \geq 2x_1 + 4x_2 + 3x_3 + 6x_4$$
$$0 \geq 23-13x_1 - 26x_2 + x_3 - 2x_4$$

el filtro sería:

$$13z - 37x_3 - 74x_4 \geq 46$$

que no nos da información sobre x_1 y x_2, deja indeterminadas sus posibilidades, con lo cual cualquier combinación de soluciones 0,1 para x_1 y x_2 es válida para el filtro, siendo (0,1,0,0) $z = 4$ la solución buscada.

5. PROPUESTA DE ALGORITMOS ALTERNATIVOS PARA RESOLVER (PK)

5.1. Un algoritmo de subgradiente.

Los resultados del apartado 4.2. nos llevan a analizar diferentes posibilidades de solución del problema (PK). La primera nos condujo a la siguiente reformulación del problema (PK):

$$[\text{MIN}]\ z$$
$$z \geq ub + (c-uA)\,x\ ,\ u\in T_k \qquad (14)$$
$$x \in R\ :\ R = \{x\ |\ vAx \geq vb\ ,\ v\in Q_k\quad x\in\{0,1\}\}$$

para el cual caben los dos casos siguientes:

CASO 1 $Q_k = \emptyset$

Si el conjunto de rayos extremos es vacío sólo tenemos restricciones del primer tipo y el problema (14) puede relajarse de la forma siguiente:

$$[\text{MIN}]\ z$$
$$z \geq ub + (c-uA)\,x\ ,\ u\in T_k \qquad (15)$$
$$x \in R\ :\ R = \{x\ |\ 0 \leq x \leq 1\}$$

que se puede replantear como:

$$[\text{MAX}]\ \{ub + (c-uA)x\}$$
$$u\in T_k \qquad (16)$$
$$x\in R$$

problema equivalente a su vez a; dado $\bar{z}$, encontrar un $\bar{x}$ tal que $z(\bar{x}) \geq \bar{z}$ siendo:

$$\bar{z} \geq ub + (c-uA)x$$
$$u\in T_k \qquad (17)$$
$$x\in R$$

e introduciendo el cambio de notación siguiente:

$$c_i = u^i b$$
$$u^i\in T_k$$
$$v_i = c-u^i A$$

de donde

$$\bar{z} \geq c_i + v_i\,x\ ,\ i = 1,2,\ldots,|T_k|$$
$$x \in R \qquad (18)$$

que es un problema resoluble mediante un algoritmo de subgradiente, complementado con una proyección sobre la región posible R. Las condiciones de finitud y convergencia del algoritmo de subgradiente para el problema (18) son las generales definidas en /11/, adaptadas a la proyección sobre la región R, /5/, /9/, lo cual significa que la sucesión $\{x^m\}$ con:

$$x^{m+1} = x^m + \theta m\, v_m \qquad (19)$$

siendo:

$$\theta m = -\lambda m\ \frac{\bar{z}-z(x^m)}{||v_m||},\quad 0< \lambda m\leq 2 \qquad (20)$$

$$v_m = c - u^m A \qquad (21\text{-a})$$
$$z(x^m) = u^m b + (c-u^m A)\,x^m \qquad (21\text{-b})$$
$$u^m = \{u^i|\ \underset{i}{\text{MAX}}\ [c_i + v_i\, x^m]\ ,\ i=1,2,\ldots,|T_k|\} \qquad (21\text{-c})$$

y la proyección:

$$x^{m+1} = \text{MAX}\ \{0,x_i^m + \theta_m\, v_{m_i}\}$$
$$x^{m+1} = \text{MIN}\ \{1,x_i^m + \theta_m\, v_{m_i}\}\quad i=1,\ldots,n \qquad (22)$$

será una sucesión convergente, puesto que la región R sobre la que se proyecta es convexa y cerrada (Shapiro, /5/).

El ALGORITMO 1 propuesto para resolver el subproblema (PK) es un algoritmo de enumeración implícita que en cada nodo a explorar utiliza el algoritmo de subgradiente como procedimiento de resolución del subproblema (15), subproblema relajado, para calcular cotas y en su caso fijar a 0 ó 1 alguna de las variables enteras.

ALGORITMO 1

FASE 1 (INICIALIZACION)

1.1. Definición del subproblema (PK) en la iteracción k-ésima del procedimiento de Banders.

1.2. Definición de: $\bar{Z}$ mejor cota superior
F = {conjunto de variables libres}
S = {conjunto de variables asignadas a 1}

FASE 2 (EXPLORACION DEL NODO l-ésimo)

2.1. Definición del subproblema relajado en el nodo l-ésimo

$$[\text{MIN}]\ W(x)$$
$$W(x)=\underset{j}{\text{MAX}}\left[c_j+\sum_{i\in F} x_i v_{ji}\right] \quad j=1,\ldots,|T_k| \qquad (23)$$
$$0\leq x_i\leq 1$$

Donde:

$$c_j = u_j\, b + \sum_i (c_i - u_j\, a_{ji}),\ i\in S$$

$$v_{ji} = c_i - u_j\, a_{ji}\ ,\ i\in S$$

2.2. Resolución del problema relajado (23) mediante el algoritmo de subgradiente (19) a (22).

2.3. Sea X^m la solución de (23) después de m iteraciones subgradientes.
Si $X^m \in \{0,1\}$ hemos obtenido una solución posible de (PK), entonces:

2.3.1. Si $W(X^m) < \bar{Z}$, entonces X^m es la nueva solución incumbente, hacer $\bar{Z}=W(X^m)$, redefinir F y S e ir al procedimiento de retroceso (2.5.).

2.3.2. Si $W(X^m) \geq \bar{Z}$, no es posible encontrar una solución posible mejor que la incumbente. Ir al procedimiento de retroceso (2.5).

Si $X^m \notin \{0.1\}$, como $0\leq X^m\leq 1$. hemos de decidir si continuamos o no la exploración del nodo:

2.3.3. Si $W(X^m) < \bar{Z}$, hay posibilidades de encontrar una solución mejor que la incumbente, por lo tanto se continúa explorando. Ir a (2.4).

2.3.4. Si $W(X^m) \geq \bar{Z}$, no es posible encontrar una solución mejor que la incumbente. Ir al procedimiento de retroceso (2.5)

2.4. Definido el subproblema del nodo l-ésimo como en (23) hay dos posibles estrategias para continuar la exploración, la primera basada en un análisis de la restricción de (23) que según el subgradiente nos ha proporcionado el mejor valor y la segunda basada en un análisis de posibilidades del conjunto de restricciones.

2.4.1. Sea u^m definido según (21-c) y m el vector correspondiente. Definamos:

$$LS = \{i \mid i\in F,\ v_{mi} < 0\}$$

Si $LS = \emptyset$ no hay posibilidad de mejorar. Ir al procedimiento de retroceso (2.5).

Si $LS \neq \emptyset$ definir el nuevo nodo $S = S\cup\{r\}$

$$r = \underset{i\in LS}{\text{MIN}}\ \{v_{mi}\}$$

CONTINUAR LA EXPLORACION

2.4.2. (Estrategia alternativa).

Definir $L = \{i \mid i\in F\ ,\ \exists j = 1,\ldots,|T k|\ ,\ v_{ji}<0\}$
Si $L = \emptyset$ no hay posibilidad de mejora.
Ir al procedimiento de retroceso (2.5).

Si $L \neq \emptyset$
Definir: $ST(i) = \sum_j v_{ji}\quad i\in L,\ j\in\{1,2,\ldots,|T k|\}$.
Definir el nuevo nodo $S = S\cup\{r\}$
$$r = \underset{i\in L}{\text{MIN}}\ \{ST(i)\}$$

CONTINUAR LA EXPLORACION

2.5. (Procedimiento de retroceso). Se utiliza el procedimiento ordinario de retroceso de los algoritmos de enumeración implícita, se define un nuevo nodo y se vuelve a empezar la fase 2, cuando el procedimiento de retroceso indique que no quedan más nodos por explorar se ha llegado a fin del algoritmo. La solución incumbente es la óptima.

CASO 2 $Q_k \neq \emptyset$

Es este caso el subproblema (PK) planteado en la iteracción k-ésima del procedimiento de Benders es el formulado según (14). El camino seguido para intentar resolver el problema fue el de adaptar el algoritmo descrito en el caso 1. La fase 2 se iniciaba de la misma manera y en el punto 2.2 se incluía la siguiente modificación:

o Si al resolver el problema (23) la solución encontrada X^m es tal que $vA\, X^m \geq vb$, continuar con los pasos siguientes, en caso contrario proyectar sobre la región posible

$$R = \{x \mid 0 \leq x \leq 1,\ vAx \geq vb,\ v \in Q_k\}$$

mediante el procedimiento de proyección siguiente;

a) Definir: $I = \{i \mid v^i\, A\, x^m < v^i\, b,\ v^i \in Q_k\}$ conjunto de índices de las restricciones violadas.

b) Calcular

$$\mu^m = \underset{i \in I}{\text{MIN}} \left[\frac{v^i b - v^i A x^m}{(v^i A)\ [c - u^m A]} \right]$$

c) Definir

$$x_i^{m+1} = \text{MAX}\ \{0,\ x_i^m + \mu^m\, v_{mi}\}$$
$$x_i^{m+1} = \text{MIN}\ \{1,\ x_i^m + \mu^m\, v_{mi}\}$$

El nuevo algoritmo satisface también las condiciones de convergencia puesto que la nueva región posible R es, a su vez, una región convexa y cerrada. Sin embargo, en la práctica el algoritmo tiene una tendencia a un efecto de zig-zag, difícilmente resoluble en este caso debido a la heurística del cálculo del parámetro en el subgradiente. Este inconveniente se puede subsanar modificando el procedimiento de proyección sobre la región posible, una alternativa sería utilizar un procedimiento de proyección sobre las restricciones activas, de manera similar a la del algoritmo de Rosen sin embargo ello nos lleva a un algoritmo casi tan complejo como el de la utilización del simplex en el cálculo de los filtros, en consecuencia decidimos explorar una vía más sencilla.

5.2. Un algoritmo de filtro para el subproblema (PK) cuando $Q_k \neq \emptyset$.

Señalábamos en 4.2 que cuando $Q_k \neq \emptyset$ el filtro de Balas obliga a generar a priori muchas soluciones no posibles, puesto que únicamente las soluciones de $0 \geq v(b - Ax)$ cuentan en este caso, una manera de obviar esta dificultad y la indicada en el apartado 3 de 4.2 sería construir un filtro de la forma:

$$[\text{MIN}]\ Z$$
$$a_1 x + Z \geq b_1 \qquad (24\text{-a})$$
$$a_2 x \geq b_2 \qquad (24\text{-b})$$
$$x \in \{0,1\}$$

donde (24-a) y (24-b) representan los filtros correspondientes a las restricciones derivadas de los puntos extremos $u \in T_k$ y los rayos extremos $v \in Q_k$ respectivamente y utilizar un algoritmo similar al del filtro de Balas pero generando soluciones únicamente a partir del filtro (24-b). La experimentación con tal tipo de filtro sugirió la conveniencia de construirlo de manera que el conjunto de soluciones posibles para la restricción $a_2 x \geq b_2$ fuese tan próximo como fuera posible, al conjunto de soluciones posibles de las restricciones $vA \geq vb$, puesto que en este caso se reduciría al máximo el número de soluciones no válidas a enumerar, minimizando así el inconveniente expuesto en 4.2, situación que sería óptima en el caso de que ambos conjuntos de soluciones posibles coincidiesen.

Una realización factible de este planteamiento se puede conseguir si en vez de calcular el filtro como una restricción de sustitución, calculando los coeficientes de la combinación lineal no negativa mediante el simplex dual, calculamos el filtro como una restricción compuesta /10/, que nos garantiza la igualdad entre el conjunto de soluciones posibles de la restricción compuesta y el conjunto de soluciones posibles de las restricciones originales. La restricción compuesta puede calcularse mediante una adaptación del algoritmo de Kendall-Zionts, /10/.

ALGORITMO 2.

FASE 1 (CALCULO DE LA RESTRICCION COMPUESTA)

1.1. Dado el problema (PK) en la iteración - k-ésima del procedimiento de Benders, - formular las desigualdades $vAx \geq vb$ de la forma:

$$\sum_{j=1}^{n} \alpha_{ij} x_j \leq \beta_i \qquad i = 1,2,\ldots,|Q_k|$$

1.2. Añadir variables de holgura para convertirlas en igualdades

$$\sum_{j=1}^{n=1} \alpha_{ij} x_j = \beta_i \qquad i = 1,2,\ldots, |Q_k|$$

$$n_1 = n + |Q_k|$$

siendo las variables de holgura x_{n+i}

Hacer $i = 1$.

1.3. Calcular los coeficientes λ_1 y λ_2 tales que satisfagan las condiciones siguientes:

a) $\lambda_1 = \beta_{i+1} - \sum_{j=1}^{n_1} \alpha^-_{i+1,j} - \text{MIN}\{|\alpha_{i+1,j}|\}$

$\alpha_{i+1,j} \neq 0$

$\lambda_2 \geq \beta_i - \sum_{j=1}^{n_1} \alpha^-_{ij} - \text{MIN}\{|\alpha_{ij}|\}$

$\alpha_{ij} \neq 0$

(siendo α^-_{ij} los $\alpha_{ij} < 0$)

b) $$\frac{\beta i+1 - \sum_{j=1}^{n_1} \alpha^-_{i+1,j}}{\lambda_1} \text{ y } \frac{\beta i - \sum_{j=1}^{n_1} \alpha^-_{ij}}{\lambda_2} \text{ no enteros}$$

c) λ_1 y λ_2 primos entre sí.

1.4. Calcular la combinación lineal con coeficientes λ_1 y λ_2 de las restricciones i-ésima de i+1-ésima:

$$a_j = \lambda_1 \alpha_{ij} + \lambda_2 \alpha_{i+1,j} \quad , j=1,\ldots,n_1$$

$$d = \lambda_1 \beta i + \lambda_2 \beta_{i+1}$$

1.5. Si $i+1 = |Q_k|$ fin del procedimiento, la - última combinación lineal calculada es la restricción compuesta. Ir a 1.6

Si $i+1 < |Q_k|$

Hacer $i = +1$

$\alpha_{ij} = a_j$, $j = 1,\ldots, n_1$, $\beta_i = d$

$i = i+1$

Volver a 1.3 y repetir el proceso combinando la última restricción compuesta - con la siguiente restricción.

1.6. Formular la restricción compuesta como

$ax \leq d$

suprimiendo las variables de holgura (a_j, $j=1,\ldots,n$)

FASE 2 (EXPLORACION DEL NODO 1-ésimo).

Sean: F = {conjunto de variables libres}
S = {conjunto de variables asignadas a 1}
X^S solución parcial.

2.1. Calcular $Y^S = a\,x^S - d$
Si $Y^S \leq 0$ entonces X^S es una solución - posible.
Ir a 2.2.

En caso contrario continuar la exploración definiendo una variable de ramificación.

Definir: LS =

$\{1 \{1 \in F \text{ y } \exists i \in M_1 : c_i - \sum_j u_i a_{ji} < 0,\ u_i \in T_k\}$

donde $M_1 = \{1,2,\ldots, |T_k|\}$

Si LS = $\emptyset$ no se puede continuar la exploración, ir a 2.3 (procedimiento de - retroceso).

Si LS $\neq \emptyset$, definir

$$ST\ (1) = \sum_{i \in M_1} (c_i - \sum_j u_i \ a_{ji})$$

Asignar $x_k = 1$, siendo $k = \underset{1 \in LS}{\text{MAX}} \{ST\ (1)\}$

Hacer $F = F - \{k\}$
$S = S \cup k$

Volver al principio de la fase 2

Calcular

$$t_i^s = u_i b - (c - u_i A)\ x^s, u_i \in T_k, \forall i \in M_1$$

Definir

$$t_{i_o} = \max_{i \in M_1} \{t_i^s\}$$

Si $t_{i_o} \geq \bar{Z}$ siendo $\bar{Z}$ la solución incumbente - de (PR), la nueva solución posible no es mejor que la incumbente.

Ir a 2.3 (Procedimiento de retroceso).

Si $t_{i_o} < \bar{Z}$ la solución posible obtenida es mejor que la incumbente. Almacenarla como nueva incumbente y hacer $\bar{Z} = t_{i_o}$.

Definir:

$$T = \{l \mid l\ F,\ a_{i_o l} < 0 \text{ y } Y^s + a_l \leq 0\}$$

Si $T = \emptyset$ no se puede mejorar la solución obtenida siguiendo la exploración.

Ir a 2.3 (Procedimiento de retroceso).

Si $T \neq \emptyset$

Asignar $x_k = 1$, $x = \min_{l \in T} \{a_{i_o l}\}$

Definir: $F = F - \{k\}$

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

Volver a 2.1

2.3. (Procedimiento de retroceso)

El mismo que el del Algoritmo 1.

El procedimiento propuesto consta pues de -- dos algoritmos y se puede esquematizar de la manera siguiente:

Dado el problema (PK) en la K-ésima itera--- ción del procedimiento de Benders si:

$Q = \emptyset$, se resuelve el problema mediante el Algoritmo 1 de subgradiente, mientras -- que si

$Q \neq \emptyset$, se resuelve mediante el Algoritmo 2 - de filtraje a partir de la restricción - compuesta".

La experiencia de cálculo ha sido buena en - una serie de problemas test de dimensiones - reducidas (hasta 20 variables binarias y 7 - restricciones), reduciéndose en algunos ca-- sos en un 50% el número de iteraciones necesario. En estos momentos estamos preparando la experiencia con problemas de mayores di-- mensiones.

Hay que señalar sin embargo los problemas de convergencia del algoritmo de subgradiente - debido a la elección del parámetro λ_m en (20). En este punto es necesario desarrollar una - heurística adecuada que vaya reduciendo la - longitud del paso a medida que el algoritmo progresa. Una heurística ensayada con éxito en nuestro caso ha consistido en reducir λm a $\lambda m/2$ la primera vez que $X^{m+1} = X^m$.

Por lo que respecta a las restricciones compuestas su cálculo resulta más rápido que el de la restricción de sustitución debido a -- que el cálculo de los coeficientes requiere un esfuerzo menor que el del simplex, y en - cada iteración del procedimiento de Benders no es necesario calcular una nueva restric-- ción compuesta sino que, si no se ha añadido ningún rayo extremo se puede utilizar la anterior, y si se ha añadido un nuevo rayo extremo, la nueva restricción de compuesta es el resultado de combinar con la nueva restricción precedente del rayo extremo. Si el problema mixto (P), que se está resolviendo por Benders, es un caso de buena convergencia -- del procedimiento de Benders, éste efectuará pocas iteraciones, por lo que habrá pocas -- restricciones en el subproblema (PK) y entonces las restricciones compuestas presentan - una clara ventaja frente a las de sustitución, queda por experimentar el comportamiento en casos de mala convergencia del procedimiento de Benders y, por lo tanto, de generación de un gran número de restricciones para (PK).

6. BIBLIOGRAFIA

/1/ BENDERS, J.F. : "Partitioning procedures for solving mixed variables programming problems", Numerische Mathematik, 4(1962) pp. 238-252.

/2/ GEOFFRION, A.M. and GRAVES, G.W. : "Multicommodity distribution system design by Benders decomposition", Management ---- Science, v. 20, Num. 5, January 1974, pp. 822-844.

/3/ VEGARA, J.M. : "Programación matemática y cálculo económico", Ed. Vicens Vives, Barcelona, 1976.

/4/ LASDON, L. : "Optimization theory for - large systems", Ed. Mac Millan, 1970.

/5/ SHAPIRO, J.E. : "Mathematical program-- ming: structures and algorithms", Ed.-- John Willey, 1979.

/6/ SALKING : "Integer Programming", Ed. John Willey, 1975.

/7/ GEOFFRION, A.M. and MARSTEN, R.E. : --- "Integer programming: a framework and - state of the art survey", Management -- Science, 18, 1972, pp. 465-491.

/8/ BALAS, E. : "Discrete programming by -- the filter method", Operations Research 15, 1967, pp. 915-957.

/9/ SANDI, C. : "Subgradient optimization" in Combinatorial Optimization ed. by -- N. Christofides, et al., John Willey, - 1979.

/10/ KENDALL, K., and ZIONTS, S. : "Solving integer programming problems by aggregating constraints", Paper presented at - the Joint Meeting of the Operations Research Society of América, The Institute of Management Sciences, American Institute of Industrial Engineers, Atlan-- tic City, New Jersey, November 8-10, -- 1972.

/11/ HELD, M., Wolf, P. and CROWDER, H. : "Validation of subgradient optimization" Math. Program., 6, 1974, pp. 62-88.