
Modelado de algoritmos de descomposición con GAMS

Santiago Cerisola¹ y Andrés Ramos² y Álvaro Baíllo³

¹ Escuela Técnica Superior de Ingeniería - ICAI
Santiago.Cerisola@iit.icaei.upco.es

² Escuela Técnica Superior de Ingeniería - ICAI Andres.Ramos@iit.icaei.upco.es

³ Escuela Técnica Superior de Ingeniería - ICAI
Alvaro.Bailllo@iit.icaei.upco.es

1. Introducción

Las técnicas de descomposición tiene sentido aplicarlas a un problema de optimización cuya estructura específica permite identificar partes del mismo que son fácilmente resolubles de modo individual. Los problemas multietapa y los problemas estocásticos de gran tamaño son ejemplos de problemas cuya resolución se puede abordar mediante técnicas de descomposición. Los problemas enteros mixtos tales que la relajación de un conjunto de restricciones reduce la dificultad del mismo son también ejemplos en los que el uso de técnicas de descomposición puede ser preferible a la resolución del problema de un modo directo.

Este documento describe una implantación de la Descomposición de Benders [2, 13] y de la Relajación Lagrangiana [6, 7] en el lenguaje de modelado algebraico GAMS [4]. La descripción se centra en problemas lineales deterministas aunque su extensión para problemas estocásticos [3, 5, 9, 10] (en el caso de Benders) o para problemas mixtos (en el caso de Relajación Lagrangiana) es inmediata. A lo largo del documento se describe brevemente cada una de las dos descomposiciones y su implantación en GAMS para la resolución de problemas académicos concretos.

2. Descomposición de Benders

El método de descomposición de Benders [2, 13] se centra en la interpretación de un problema de optimización como un problema bietapa PL-2. Para este tipo de problemas, las variables de decisión aparecen claramente separadas en dos bloques: las variables de la primera etapa y las de la segunda etapa. Esta división induce de modo natural un algoritmo iterativo de resolución en el que los problemas asociados a las variables de la primera

etapa (problema maestro) y los asociados a las de la segunda etapa (subproblema) son resueltos consecutivamente. Este método de descomposición recibe también el nombre de *descomposición primal* (porque el problema maestro fija variables del primal), *descomposición en L* (porque se aplica a problemas con matriz de restricciones con dicha forma) y *descomposición por recursos* (porque el maestro asigna directamente las decisiones al subproblema sobre los recursos).

Un problema lineal bietapa PL-2 se representa matemáticamente de la forma siguiente

$$\begin{aligned} \min_{x,y} c^T x + q^T y \\ Ax = b \\ Tx + Wy = h \\ x, y \geq 0 \end{aligned} \quad (1)$$

donde x representa el conjunto de variables de la primera etapa e y representa el conjunto de variables de la segunda etapa. Supondremos que $A \in \mathfrak{R}^{m_1 \times n_1}$ y $W \in \mathfrak{R}^{m_2 \times n_2}$ y que las restantes dimensiones son conformes con éstas⁴. El tamaño del problema completo es $(m_1 + m_2) \times (n_1 + n_2)$. La estructura de la matriz de restricciones del problema (denominada triangular inferior por bloques) se presenta en la figura 1. Las restricciones $Ax = b$ afectan únicamente a las variables de la primera etapa x , mientras que las restricciones $Tx + Wy = h$ afectan a ambos conjuntos de variables x e y .

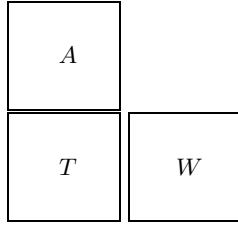


Figura 1. Estructura de la matriz de coeficientes de las restricciones en problemas lineales bietapa.

El problema lineal bietapa PL-2 (1) se puede interpretar también de esta manera:

$$\begin{aligned} \min_x c^T x + \theta(x) \\ Ax = b \\ x \geq 0 \end{aligned} \quad (2)$$

⁴Por convención en la formulación los vectores son columna, su transposición se representa por un superíndice T , las variables se ubican a la izquierda de las expresiones y los coeficientes de las variables preceden a éstas.

donde la función de recurso, $\theta(x)$, representa la función objetivo de la segunda etapa como función de las decisiones de la primera etapa y tiene la siguiente expresión:

$$\begin{aligned} \theta(x) = \min_y q^T y \\ W y = h - T x \quad : \pi \\ y \geq 0 \end{aligned} \quad (3)$$

donde π son las variables duales de las restricciones.

El problema (2) se conoce en la literatura como problema maestro y el problema (3), que evalúa la función de recurso para las decisiones de la primera etapa, como subproblema de la descomposición de Benders. El problema maestro puede ser reformulado de modo lineal utilizando la representación dual del subproblema. Con esta reformulación, el algoritmo de descomposición resuelve en cada iteración un problema maestro lineal.

2.1. Representación lineal del problema maestro

El subproblema, expresado en su forma dual es:

$$\begin{aligned} \theta(x) = \max_{\pi} (h - T x)^T \pi \\ W^T \pi \leq q \end{aligned} \quad (4)$$

Sea $\Pi = \{\pi^1, \pi^2, \dots, \pi^\nu\}$ el conjunto finito de vértices del poliedro convexo definido por la región factible $W^T \pi \leq q$. Obsérvese que la región factible del problema dual no depende del valor de x . Dado que la solución óptima de un problema lineal reside en un vértice el problema se puede resolver por enumeración de todos ellos:

$$\theta(x) = \max\{(h - T x)^T \pi^l\} \quad l = 1, \dots, \nu \quad (5)$$

El subproblema de Benders puede ser reformulado como:

$$\begin{aligned} \theta(x) = \min_{\theta} \theta \\ \theta \geq (h - T x)^T \pi^1 \\ \vdots \\ \theta \geq (h - T x)^T \pi^\nu \end{aligned} \quad (6)$$

con $\theta \in \Re$. Obsérvese que la variable θ es libre. Las restricciones de esta formulación se denominan cortes de Benders y constituyen una aproximación exterior de la función de recurso. De la ecuación anterior se deriva que la función de recurso $\theta(x)$ es una función poligonal convexa de las variables x . El problema original PL-2 se puede expresar como:

$$\begin{aligned}
& \min_{x, \theta} c^T x + \theta \\
& Ax = b \\
& \theta \geq (h - Tx)^T \pi^1 \\
& \vdots \\
& \theta \geq (h - Tx)^T \pi^\nu \\
& x \geq 0
\end{aligned} \tag{7}$$

Esta formulación se denomina *problema maestro completo*, ya que contiene todos los cortes posibles. Presenta todas las restricciones de la primera etapa más todas las condiciones necesarias derivadas de la segunda etapa. Desde el punto de vista práctico, la resolución del problema maestro completo implica disponer de forma explícita de todos los cortes de Benders, lo cual es prácticamente imposible en problemas de tamaño realista. Por esta razón, la resolución del problema original PL-2 es reemplazada por la resolución iterativa del *problema maestro relajado*⁵ definido como:

$$\begin{aligned}
& \min_{x, \theta} c^T x + \theta \\
& Ax = b \\
& \theta \geq (h - Tx)^T \pi^l \quad l = 1, \dots, j \\
& x \geq 0
\end{aligned} \tag{8}$$

donde $\theta \in \Re$ y l representa el índice de iteraciones⁶. De cara a una implantación eficiente del algoritmo de descomposición, los cortes de Benders aceptan la siguiente formulación como linealización de la función de recurso en torno a los valores de las variables de la primera etapa y de los valores de la variable θ obtenidos en cada iteración:

$$\begin{aligned}
\theta & \geq \pi^{jT}(h - Tx) = \pi^{jT}(h - Tx + Tx^j - Tx^j) = \\
& = \pi^{jT}[h - Tx^j - T(x - x^j)] = \pi^{jT}(h - Tx^j) - \pi^{jT}T(x - x^j) = \\
& = f^j - \pi^{jT}T(x - x^j)
\end{aligned} \tag{9}$$

siendo x^j y $f^j = \pi^{jT}(h - Tx^j)$ los valores de las variables de la primera etapa y el de la función de recurso para la iteración j , respectivamente. De este modo el corte para la iteración j también se expresa como

$$\theta - f^j \geq \pi^{jT}T(x^j - x) \tag{10}$$

o de modo equivalente como

$$\theta + \pi^{jT}Tx \geq f^j + \pi^{jT}Tx^j \tag{11}$$

Esta expresión indica que $\pi^{jT}T$ es un subgradiente del valor de la función de recurso $\theta(x)$ para la propuesta x^j del maestro.

⁵Por simplicidad de la exposición, se denomina como problema maestro al problema maestro relajado.

⁶Se entiende por iteración un ciclo maestro-subproblema, en este orden.

Con esta formulación el *problema maestro relajado de Benders* tiene ahora esta expresión:

$$\begin{aligned}
 & \min_{x, \theta} c^T x + \theta \\
 & Ax = b \\
 & \theta + \pi^{lT} T x \geq f^l + \pi^{lT} T x^l \quad l = 1, \dots, j \\
 & x \geq 0
 \end{aligned} \tag{12}$$

y el *subproblema de Benders* para cada iteración j se formula como:

$$\begin{aligned}
 f^j &= \min_y q^T y \\
 Wy &= h - T x^j \quad : \pi^j \\
 y &\geq 0
 \end{aligned} \tag{13}$$

2.2. Cortes de infactibilidad

La descripción anterior del algoritmo de Benders ha supuesto que el subproblema de Benders es factible y acotado para cualquier propuesta del problema maestro. Esta hipótesis, conocida en la literatura como *recurso parcialmente completo*⁷, no suele satisfacerse en la práctica y el algoritmo de descomposición es modificado cuando esto ocurre. La modificación del algoritmo consiste en la construcción de otro tipo de corte, *corte de infactibilidad*, que elimina la solución propuesta en el problema maestro. La construcción de este corte se comenta a continuación.

Si el subproblema es factible para un valor de x los valores duales de las restricciones forman un corte en el problema maestro denominado *corte de optimalidad* tal como se ha presentado anteriormente. El caso en el que un subproblema es no acotado carece de interés algorítmico, puesto que esta situación implica que el problema bietapa PL-2 es no acotado. La situación que debe destacarse es aquélla en la que el subproblema es infactible para la propuesta del maestro. En ese caso se puede generar un *corte de infactibilidad* derivado de la Fase I del simplex [13]. El subproblema de minimización de infactibilidades se formula como:

$$\begin{aligned}
 \theta^*(x) &= \min_{v^+, v^-} e^T v^+ + e^T v^- \\
 Wy + I v^+ - I v^- &= h - T x \quad : \pi \\
 y, v^+, v^- &\geq 0
 \end{aligned} \tag{14}$$

siendo $e^T = (1 \dots 1)$, I matriz identidad ($m_2 \times m_2$) y π las variables duales de las restricciones para la solución óptima. $\theta^*(x)$ representa la función de recurso asociada a la fase I del subproblema de Benders.

⁷En un problema bietapa por *recurso completo* se entiende que el subproblema de la segunda etapa es siempre factible para cualquier valor de las variables de la primera etapa.

El corte de optimalidad que el subproblema de minimización de infactibilidades obtiene (siguiendo la metodología descrita hasta el momento) viene dado como:

$$\theta^* \geq (h - Tx)^T \pi^j \quad (15)$$

Reemplazando la variable de recurso θ^* por 0 se obtiene una condición necesaria para aquellas soluciones del problema maestro que son factibles en el subproblema de Benders. Esta condición indica que la suma de infactibilidades para esas soluciones debe ser menor que 0. Linealizando en torno a la solución obtenida en el problema maestro se obtiene la siguiente expresión para el corte de infactibilidad, similar a la del corte de optimalidad:

$$\pi^{jT}Tx \geq f^j + \pi^{jT}Tx^j \quad (16)$$

donde f^j representa el valor del subproblema de minimización de infactibilidades para la solución propuesta.

El problema maestro relajado considerando ambos tipos de cortes se formula de la siguiente manera:

$$\begin{aligned} & \min_{x, \theta} c^T x + \theta \\ & Ax = b \\ & \delta^l \theta + \pi^{lT}Tx \geq f^l + \pi^{lT}Tx^l \quad l = 1, \dots, j \\ & x \geq 0 \end{aligned} \quad (17)$$

siendo $\delta^l = 1$ para los cortes de optimalidad y $\delta^l = 0$ para los de infactibilidad.

2.3. Algoritmo de descomposición de Benders

El algoritmo de descomposición de Benders resuelve en cada iteración el problema maestro relajado y pasa el valor de las variables de la primera etapa x^j al subproblema. La resolución del subproblema de Benders con el término de la derecha (RHS) $(h - Tx^j)$ genera unas variables duales π^j que se usan para generar un nuevo corte de Benders que se añade al problema maestro relajado. En cada iteración, el valor obtenido por la función objetivo del problema maestro relajado \underline{z} es una *cota inferior* del problema completo PL-2. La sucesión de estas cotas inferiores es monótona creciente dado que en cada iteración el problema maestro relajado contiene mayor número de restricciones. Por otra parte, una *cota superior* \bar{z} del valor óptimo de la función objetivo del problema original PL-2 viene dada en cada iteración por $(c^T x^j + q^T y^j)$ siendo x^j e y^j soluciones factibles en maestro y subproblema en esa iteración. El criterio de parada del algoritmo es la coincidencia de ambas cotas con una tolerancia relativa ϵ (por ejemplo, 10^{-4}).

$$\frac{|\bar{z} - \underline{z}|}{|\bar{z}|} = \frac{|q^T y^j - \theta^j|}{|c^T x^j + q^T y^j|} \leq \epsilon \quad (18)$$

En cada iteración del algoritmo de Benders, la variable dual generada en el subproblema es distinta del conjunto de variables duales generadas con anterioridad por el algoritmo [13]. Dado que el conjunto de posibles valores duales es finito, el algoritmo de descomposición de Benders converge en un número finito de iteraciones.

Para obtener el valor x^0 de las variables de la primera etapa en la primera iteración del algoritmo, se suele proceder a la resolución del problema maestro relajado sin cortes, lo que equivale a fijar el valor de la variable de recurso a cero, $\theta = 0$. Otra alternativa consiste en estimar razonablemente este valor en caso de que la naturaleza del problema sea conocida.

Para el caso en que las variables de la primera etapa sean continuas, en cada iteración se dispone de una base del problema maestro relajado que es infactible sólo por una variable básica, la variable de holgura del nuevo corte. Cuando se soluciona el dual del problema maestro relajado los cortes aparecen como columnas. Al añadir una nueva columna la solución previa sigue siendo factible y la nueva solución óptima puede obtenerse en pocas iteraciones del simplex. Por ello teóricamente es conveniente resolver el maestro mediante el simplex dual. Por otra parte, cada subproblema sólo cambia las cotas de las restricciones en cada iteración. Por esta razón, suponiendo que ninguna solución del maestro ocasiona infactibilidad en el subproblema, es conveniente resolverlo mediante el método simplex primal (siempre que su tamaño lo aconseje).

Esquemáticamente el algoritmo se formula a continuación:

1. Inicialización: $j = 0$, $\bar{z} = \infty$, $\underline{z} = -\infty$, $\epsilon = 10^{-4}$
2. Resolución del problema maestro

$$\begin{aligned}
 & \min_{x, \theta} c^T x + \theta \\
 & Ax = b \\
 & \delta^l \theta + \pi^{lT} T x \geq f^l + \pi^{lT} T x^l \quad l = 1, \dots, j \\
 & x \geq 0
 \end{aligned} \tag{19}$$

Obtener la solución x^{j+1} , θ^{j+1} y evaluar la cota inferior $\underline{z} = (c^T x^{j+1} + \theta^{j+1})$.

3. Resolución del subproblema de suma de infactibilidades

$$\begin{aligned}
 f^{j+1} &= \min_{v^+, v^-} e^T v^+ + e^T v^- \\
 & Wy + Iv^+ - Iv^- = h - T x^{j+1} \quad : \pi^{j+1} \\
 & y, v^+, v^- \geq 0
 \end{aligned} \tag{20}$$

Si $f^{j+1} \geq 0$, obtener π^{j+1} , formar un corte de infactibilidad y añadirlo al problema maestro, incrementar el número de iteraciones $j = j + 1$ e ir al paso 2.

Si $f^{j+1} = 0$, ir al paso 4.

4. Resolución del subproblema de Benders

$$\begin{aligned}
f^{j+1} &= \min_y q^T y \\
Wy &= h - Tx^{j+1} & : \pi^{j+1} \\
y &\geq 0
\end{aligned} \tag{21}$$

Obtener y^{j+1} y actualizar cota superior $\bar{z} = (c^T x^{j+1} + q^T y^{j+1})$.

5. Regla de parada

Si $\frac{|\bar{z} - z|}{|\bar{z}|} \leq \epsilon$ detener el algoritmo. En otro caso, obtener π^{j+1} , formar un corte de optimalidad y añadirlo al problema maestro, incrementar el número de iteraciones $j = j + 1$ e ir al paso 2.

Mientras no se haya generado ningún corte de optimalidad, se fija el valor de la variable de recurso θ a cero, pues en otro caso el problema maestro es no acotado. Una vez obtenido algún corte de infactibilidad, esta variable pasa a ser libre.

La siguiente sección presenta con un ejemplo los pasos del algoritmo de descomposición de Benders.

2.4. Ejemplo

Supongamos que se desea resolver el siguiente problema de programación lineal:

$$\begin{aligned}
\min_{x_1, x_2, y_1, y_2} \quad & -x_1 - 2x_2 - 2y_1 - 3y_2 \\
& x_1 + x_2 \leq 600 \\
& x_1 - 2x_2 \leq 0 \\
& x_1 + x_2 + y_1 + y_2 \leq 1000 \\
& x_1 + y_1 \leq 500 \\
& -2y_1 + y_2 \leq 0 \\
& x_1, x_2, y_1, y_2 \geq 0
\end{aligned}$$

entonces,

$$\begin{aligned}
c &= \begin{pmatrix} -1 \\ -2 \\ -2 \\ -3 \end{pmatrix}, \quad q = \begin{pmatrix} -2 \\ -3 \end{pmatrix}, \quad b = \begin{pmatrix} 600 \\ 0 \end{pmatrix}, \quad h = \begin{pmatrix} 1000 \\ 500 \\ 0 \end{pmatrix}, \\
A &= \begin{pmatrix} 1 & 1 \\ 1 & -2 \end{pmatrix}, \quad T = \begin{pmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 0 \end{pmatrix} \quad \text{y} \quad W = \begin{pmatrix} 1 & 1 \\ 1 & 0 \\ -2 & 1 \end{pmatrix}
\end{aligned}$$

El algoritmo comienza con un problema maestro que no contiene ningún corte de Benders, por lo que el valor de la variable de recurso θ está fijado a cero. La resolución del primer problema maestro obtiene la siguiente solución para las variables de la primera etapa:

$$x = \begin{pmatrix} 0 \\ 600 \end{pmatrix}$$

Para estos valores de la primera etapa, el subproblema es factible y la cota inferior es $\underline{z} = -1200,0$. Como solución se obtiene

$$y = \begin{pmatrix} 133,3 \\ 266,7 \end{pmatrix} \quad y \quad \pi = \begin{pmatrix} -2,667 \\ 0,000 \\ -0,333 \end{pmatrix}$$

y un valor para la cota superior $\bar{z} = -2266,7$. Como la diferencia entre ambas cotas es elevada (la cota superior es este momento es ∞) se continúa iterando. Con los valores duales se forma un corte para el problema maestro dado por:

$$\theta - 2,667x_1 - 2,667x_2 \geq -2666,7$$

La resolución del problema maestro obtiene una nueva solución

$$x = \begin{pmatrix} 0,000 \\ 0,000 \end{pmatrix}$$

y un valor para la cota inferior $\underline{z} = -2666,7$. Al introducir en el subproblema estos valores se obtiene como solución

$$y = \begin{pmatrix} 333,3 \\ 666,7 \end{pmatrix} \quad y \quad \pi = \begin{pmatrix} -2,667 \\ 0,000 \\ -0,333 \end{pmatrix}$$

Se calcula la cota superior en esta iteración, siendo $\bar{z} = -2666,7$. La diferencia entre ambas cotas es ahora nula y, por lo tanto, acaba el algoritmo. La función objetivo es $z^* = -2666,7$ y la solución óptima es:

$$x^* = \begin{pmatrix} 0,000 \\ 0,000 \end{pmatrix} \quad e \quad y^* = \begin{pmatrix} 333,3 \\ 666,7 \end{pmatrix}$$

2.5. Descomposición de Benders en GAMS

Dentro del lenguaje GAMS la descomposición de Benders puede implantarse creando un modelo para el problema maestro y un modelo para el subproblema. La adición de cortes para el problema maestro dentro del proceso iterativo se consigue declarando la ecuación de cortes de Benders sobre un índice estático sobre el que se construye un subconjunto dinámico que representa el conjunto de cortes activos en cada iteración. Posteriormente, la ecuación de cortes se construye sobre este conjunto dinámico, de modo que durante el proceso iterativo la activación incremental del conjunto dinámico implica la inclusión incremental de estas restricciones adicionales en el problema maestro. Estas ideas se presentan a continuación en el siguiente código que resuelve el problema académico de la anterior sección. Se ha seguido en la medida de lo posible la notación y la formulación presentada anteriormente.

```

$title Descomposición de Benders (Bd)

SETS
  I  indice de variables      de la primera etapa / i1 * i2 /
  L  indice de variables      de la segunda etapa / l1 * l2 /
  M  indice de restricciones  de la primera etapa / m1 * m2 /
  N  indice de restricciones  de la segunda etapa / n1 * n3 /

  J  indice de iteraciones                    / j1 * j20 /
  JJ(j) subconjunto de iteraciones

* Datos del problema

PARAMETERS
  C(i)  coeficientes función objetivo primera etapa
        / i1  -1
          i2  -2 /
  Q(l)  coeficientes función objetivo segunda etapa
        / l1  -2
          l2  -3 /
  B(m)  cotas restricciones primera etapa
        / m1  600
          m2   0 /
  H(n)  cotas restricciones segunda etapa
        / n1 1000
          n2  500
          n3   0 /

TABLE A(m,i) matriz de restricciones primera etapa
           i1  i2
  m1      1   1
  m2      1  -2

TABLE T(n,i) matriz de restricciones segunda etapa
           i1  i2
  n1      1   1
  n2      1   0
  n3      0   0

TABLE W(n,l) matriz de restricciones segunda etapa
           l1  l2
  n1      1   1
  n2      1   0
  n3     -2   1

* Fin datos del problema

POSITIVE VARIABLES
  X(i)  variables primera etapa
  Y(l)  variables segunda etapa

VARIABLES
  Z1    función objetivo primera etapa
  Z2    función objetivo segunda etapa
  theta función de recurso

SCALARS
  TOL    tolerancia relativa / 1e-6 /
  Z_INF  cota inferior      / -INF /
  Z_SUP  cota superior      /  INF /

PARAMETERS
  PI(n,j) variables duales restricciones segunda etapa en la iteración j
  DELTA(j) tipo de corte (infactible 0 óptimo 1) de la iteración j
  X_J(i,j) valores de las variables de la primera etapa en la iteración j
  Z2_J(j)  valor de la función objetivo del subproblema en la iteración j

```

```

* Declaración de las ecuaciones

EQUATIONS
  EQ_OBJ      función objetivo problema completo
  EQ_Z1      función objetivo primera etapa
  EQ_Z2      función objetivo segunda etapa
  EQ_R1(m)   restricciones primera etapa
  EQ_R2(n)   restricciones segunda etapa
  CORTES(j)  cortes de Benders ;

* Construcción de las ecuaciones

EQ_OBJ      .. Z1 =E= SUM(i, C(i)*X(i)) + SUM(l, Q(l)*Y(l)) ;

EQ_Z1      .. Z1 =E= SUM(i, C(i)*X(i)) + theta ;

EQ_Z2      .. Z2 =E= SUM(l, Q(l)*Y(l)) ;

EQ_R1(m)   .. SUM(i, A(m,i)*X(i)) =L= B(m) ;

EQ_R2(n)   .. SUM(i, T(n,i)*X(i)) + SUM(l, W(n,l)*Y(l)) =L= H(n) ;

CORTES(jj) .. DELTA(jj) * theta =G= Z2_J(jj) +
              SUM(n, PI(n,jj)*SUM(i, T(n,i)*(X_J(i,jj) - X(i)))) ;

MODEL MAESTRO / EQ_R1, CORTES, EQ_Z1 / ;
MODEL SUBPROBLEMA / EQ_R2, EQ_Z2 / ;
MODEL COMPLETO / EQ_R1, EQ_R2, EQ_OBJ / ;

FILE COPT / cplex.opt /

PUT COPT PUT 'scaind -1' / 'lpmethod 1' / 'preind 0' / 'epopt 1.1e-9' / 'eprhs 1.1e-9'
PUTCLOSE COPT ;

SUBPROBLEMA.OPTFILE = 1 ;

* Algoritmo de Descomposición de Benders

* Inicialización de parámetros del problema
JJ(j) = NO ;
theta.FX = 0 ;
DELTA(j) = 0 ;
Z2_J(j) = 0 ;
X_J(i,j) = 0 ;
PI(n,j) = 0 ;

* Iteraciones del algoritmo

LOOP(j $(ABS(1-Z_INF/Z_SUP) > TOL),

* Resolución del problema maestro
  SOLVE MAESTRO USING LP MINIMIZING Z1 ;
* Adquisición de la solución
  X_J(i,j) = X.L(i) ;

* Fijación de la variable de la primera etapa y
* Resolución del subproblema
  X.FX(i) = X.L(i) ;
  SOLVE SUBPROBLEMA USING LP MINIMIZING Z2 ;

* Adquisición de los parámetros para formar un nuevo corte
  IF(SUBPROBLEMA.MODELSTAT = 4,
* Subproblema infactible
  DELTA(j) = 0 ;
  Z2_J(j) = SUBPROBLEMA.SUMINFES ;
  ELSE

```

```

*   Actualización de la cota inferior
    Z_INF = Z1.L ;
    DELTA(j) = 1 ;
    theta.LO = -INF ; theta.UP = INF ;
*   Actualización de la cota superior
    Z_SUP = SUM(i, C(i)*X.L(i)) + SUM(1, Q(1)*Y.L(1)) ;
) ;

Z2_J(j) = Z2.L ;
PI(n,j) = EQ_R2.M(n) ;
X.LO(i) = 0 ; X.UP(i) = INF ;

*   Incremento del conjunto de cortes
    JJ(j) = YES ;
) ;

```

Los lenguajes de modelado algebraico como GAMS permiten el uso de diferentes optimizadores para resolver los problemas. En el caso anterior, el optimizador escogido ha sido CPLEX [8]. Para que las variables duales devueltas por el subproblema de Benders sean correctas para generar un corte de Benders tanto en el caso de optimalidad como en el de infactibilidad, para este optimizador deben ser desactivadas las opciones de *preproceso* (`preind 0`) y de *escalado* (`scaind -1`) y el subproblema debe ser resuelto mediante el algoritmo simplex primal (`lpmethod 1`). Con el uso de otros optimizadores se debe comprobar la corrección de las variables duales tanto para optimalidad como para infactibilidad. Si las variables duales no son las adecuadas para generar un corte de infactibilidad, el subproblema de minimización de infactibilidades debe de ser planteado explícitamente y resuelto en el código GAMS.

2.6. Problema de transporte con coste fijo mediante descomposición de Benders

Veamos a continuación un caso ejemplo característico de la aplicación del método de descomposición de Benders. Se trata del problema de transporte donde algunos o todos los arcos tienen un coste fijo asociado a la decisión de su instalación o a su uso. El problema consiste en la minimización de los costes fijos y variables sujeto a las restricciones de respetar las ofertas máximas de los orígenes y las demandas en los destinos. El problema se formula de la siguiente manera

$$\begin{aligned}
& \min_{x_{ij}, y_{ij}} \sum_{ij} (c_{ij}x_{ij} + f_{ij}y_{ij}) \\
& \sum_j x_{ij} \leq a_i \\
& \sum_i x_{ij} \geq b_j \\
& x_{ij} \leq M_{ij}y_{ij} \\
& x_{ij} \geq 0, y_{ij} \in \{0, 1\}
\end{aligned} \tag{22}$$

siendo c_{ij} el coste variable unitario de transporte, f_{ij} el coste fijo asociado a la decisión de inversión en el arco ij , a_i la oferta máxima de producto en el origen i , b_j la demanda del destino j , x_{ij} la variable que indica el flujo que

recorre el arco ij , y_{ij} la variable que representa la decisión de inversión en el arco ij y M_{ij} un cota superior de cualquier flujo en dicho arco ij (por ejemplo, $M_{ij} = \min\{a_i, b_j\}$).

Las variables y_{ij} son binarias. Una vez conocidas el problema anterior es un problema clásico de transporte. Las variables y_{ij} son las variables que complican la resolución y, por consiguiente, son asignadas al problema maestro en un entorno de descomposición de Benders. El subproblema se formula de la siguiente manera

$$\begin{aligned}
 & \min_{x_{ij}} \sum_{ij} c_{ij} x_{ij} \\
 & \sum_j x_{ij} \leq a_i \\
 & \sum_i x_{ij} \geq b_j \\
 & x_{ij} \leq M_{ij} y_{ij}^k \quad : \pi_{ij}^k \\
 & x_{ij} \geq 0
 \end{aligned} \tag{23}$$

y el problema maestro como

$$\begin{aligned}
 & \min_{y_{ij}, \theta} \sum_{ij} (\theta + f_{ij} y_{ij}) \\
 & \theta + \sum_{ij} (\pi_{ij}^l M_{ij} y_{ij}) \geq f^l + \sum_{ij} (\pi_{ij}^l M_{ij} y_{ij}^l) \quad l = 1, \dots, k \\
 & y_{ij} \in \{0, 1\}
 \end{aligned} \tag{24}$$

A continuación se expresa en GAMS este problema para un caso ejemplo en el que se suponen cuatro orígenes del producto y tres puntos de demanda. El problema debe decidir la combinación óptima de arcos de entre todos los posibles, dados en la figura 2.

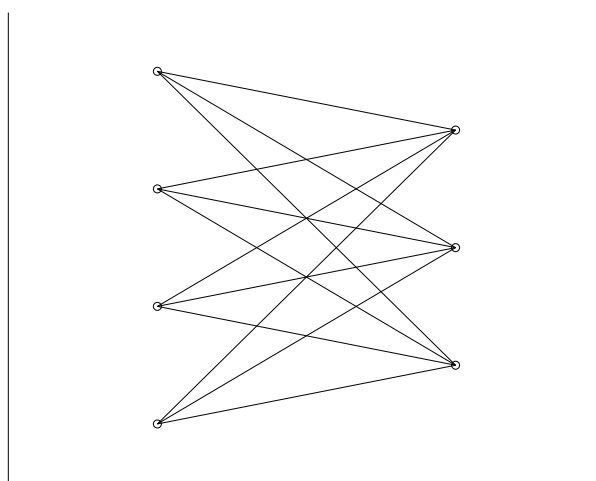


Figura 2. Arcos posibles

```

$TITLE Problema de transporte con coste fijo (Bd)

* Tolerancia relativa en la resolución de problemas MIP
OPTION OPTCR = 0

SETS
  J      índice de iteraciones / j1 * j20 /
  JJ(j) subconjunto de iteraciones
  I      orígenes / i1 * i4 /
  L      destinos / l1 * l3 /

* Datos del problema

PARAMETERS
  A(i)  ofertas de producto
        / i1 10, i2 30, i3 40, i4 20 /
  B(l)  demandas de producto
        / l1 20, l2 50, l3 30 /

TABLE C(i,l) coste variable unitario de transporte
      i1  i2  i3
i1  1   2   3
i2  3   2   1
i3  2   3   4
i4  4   3   2

TABLE F(i,l) coste fijo de transporte
      i1  i2  i3
i1  10  20  30
i2  20  30  40
i3  30  40  50
i4  40  50  60

* Fin datos del problema

ABORT $(SUM[i, A(i)] < SUM[l, B(l)]) 'Problema infactible'

POSITIVE VARIABLES
  X(i,l)  flujo por los arcos

BINARY VARIABLE
  Y(i,l)  decisiones de inversión en los arcos

VARIABLES
  Z1      función objetivo primera etapa
  Z2      función objetivo segunda etapa
  theta   función de recurso

SCALARS
  TOL     tolerancia relativa / 1e-6 /
  Z_INF   cota inferior / -INF /
  Z_SUP   cota superior / INF /

PARAMETERS
  Y_J(i,l,j) valores de las variables de la primera etapa en la iteración j
  PI(i,l,j)  variables duales restricciones segunda etapa en la iteración j
  DELTA(j)  tipo de corte (infactible 0 óptimo 1) de la iteración j
  Z2_J(j)   valor de la función objetivo del subproblema en la iteración j

* Declaración de las ecuaciones

EQUATIONS
  EQ_Z1     función objetivo primera etapa
  EQ_Z2     función objetivo segunda etapa
  EQ_OBJ    función objetivo problema completo
  OFERTA(i) ofertas de los orígenes

```

```

DEMANDA(1) demanda de los destinos
LIMITE(i,1) límite de uso del arco
CORTES(j) cortes de Benders ;

EQ_Z1      .. Z1 =E= SUM[(i,1), F(i,1)*Y(i,1)] + theta ;

EQ_Z2      .. Z2 =E= SUM[(i,1), C(i,1)*X(i,1)] ;

EQ_OBJ     .. Z1 =E= SUM[(i,1), F(i,1)*Y(i,1)] + SUM[(i,1), C(i,1)*X(i,1)] ;

OFERTA(i)  .. SUM[1, X(i,1)] =L= A(i) ;

DEMANDA(1) .. SUM[i, X(i,1)] =G= B(1) ;

LIMITE(i,1) .. X(i,1) =L= MIN[A(i),B(1)] * Y(i,1) ;

CORTES(jj) .. DELTA(jj) * theta =G= Z2_J(jj) -
              SUM[(i,1), PI(i,1,jj) * MIN[A(i),B(1)] * (Y_J(i,1,jj) - Y(i,1))] ;

MODEL MAESTRO      / EQ_Z1, CORTES / ;
MODEL SUBPROBLEMA / EQ_Z2, OFERTA, DEMANDA, LIMITE / ;
MODEL COMPLETO    / EQ_OBJ, OFERTA, DEMANDA, LIMITE / ;

X.UP(i,1) = MIN[A(i),B(1)]

FILE COPT / cplex.opt /

* para que los resultados de un problema infactible sean los correctos con CPLEX sólo
* se puede utilizar el método simplex y sin preproceso ni escalado

PUT COPT PUT 'scaind -1' / 'lpmethod 1' / 'preind 0' / 'epopt 1.1e-9' / 'eprhs 1.1e-9'
PUTCLOSE COPT ;

SUBPROBLEMA.OPTFILE = 1 ;

* inicialización de parámetros del problema

JJ(j)      = NO ;
theta.FX   = 0 ;
DELTA(j)   = 0 ;
Y_J(i,1,j) = 0 ;
Z2_J(j)    = 0 ;
PI(i,1,j)  = 0 ;

* Iteraciones del algoritmo
LOOP(j $(ABS(1-Z_INF/Z_SUP) > TOL),

* Resolución del problema maestro
SOLVE MAESTRO USING MIP MINIMIZING Z1 ;
* Adquisición de la solución
Y_J(i,1,j) = Y.L(i,1) ;

* Fijación de la variable de la primera etapa y
* Resolución del subproblema
Y.FX(i,1) = Y.L(i,1) ;
SOLVE SUBPROBLEMA USING RMIP MINIMIZING Z2 ;

* Adquisición de los parámetros para formar un nuevo corte
IF(SUBPROBLEMA.MODELSTAT = 4,
  DELTA(j) = 0 ;
  Z2_J(j) = SUBPROBLEMA.SUMINFES ;
ELSE
* Actualización de la cota inferior
Z_INF = Z1.L ;
DELTA(j) = 1 ;
theta.LO = -INF; theta.UP = INF;
* Actualización de la cota superior

```

```

      Z_SUP = SUM[(i,1), F(i,1)*Y.L(i,1)] + SUM[(i,1), C(i,1)*X.L(i,1)] ;
      Z2_J(j) = Z2.L;
    ) ;

    PI(i,1,j) = LIMITE.M(i,1) ;
    Y.LO(i,1) = 0 ; Y.UP(i,1) = 1 ;

*   Incremento del conjunto de cortes
    JJ(j) = YES ;
  ) ;

```

La solución óptima es $y_{11} = y_{23} = y_{31} = y_{32} = y_{42} = 1$ y se alcanza en 15 iteraciones⁸ con un coste total, fijo más variable, de 380. La evolución de las combinaciones de arcos propuestas por el algoritmo de descomposición se presenta en la figura 3. Debe destacarse que solamente las propuestas de las iteraciones 7, 10 y 15 son factibles para el subproblema, de modo que la mayoría de cortes que el algoritmo genera son de infactibilidad y sólo dos cortes son de optimalidad. Por último, el cuadro 1 presenta la evolución de la cota superior y la cota inferior del problema durante el algoritmo. En este ejemplo, el algoritmo converge cuando la cota inferior es exactamente la misma que la cota superior. La convergencia se alcanza también cuando aparece una propuesta repetida. En este ejemplo, la propuesta de arcos de la última iteración es, de hecho, la propuesta de la iteración 10.

Iteración	Cota Inferior	Cota Superior
1	$-\infty$	∞
2	$-\infty$	∞
3	$-\infty$	∞
4	$-\infty$	∞
5	$-\infty$	∞
6	$-\infty$	∞
7	140	390
8	140	390
9	120	390
10	360	380
11	360	380
12	360	380
13	360	380
14	360	380
15	380	380

Cuadro 1. Evolución de cotas en el algoritmo de Benders

⁸En este caso particular el subproblema tiene soluciones alternativas por lo que el algoritmo puede converger en diferente número de iteraciones con otro optimizador u otra versión del mismo optimizador.

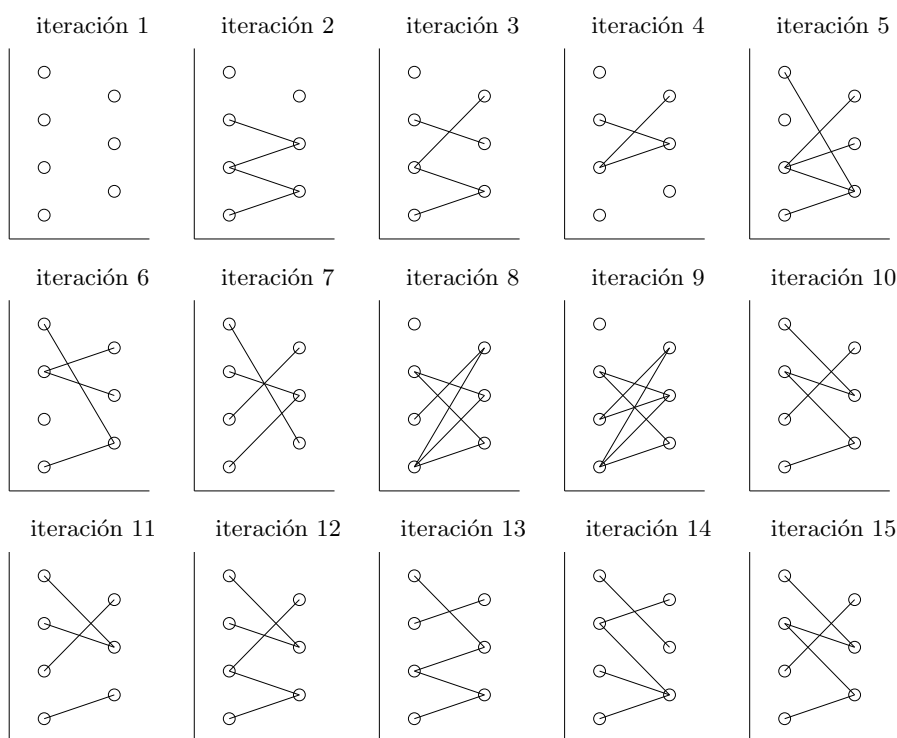


Figura 3. Evolución de arcos propuestos por el algoritmo de Benders

3. Relajación Lagrangiana

La Relajación Lagrangiana [6, 7] es una de las técnicas más extendidas en optimización discreta. Se emplea principalmente cuando en un problema de programación matemática aparece un conjunto de ecuaciones que *complica* la resolución del problema. Esto es, la resolución del problema sin esas ecuaciones tiene una estructura cuya resolución es más sencilla. Esta técnica se basa en la *dualización* de esas ecuaciones y en la formulación de un *problema dual* cuya resolución proporciona una aproximación del valor óptimo del problema original mejor que la resolución de su relajación lineal⁹. En caso de que el problema sea lineal y que el conjunto de ecuaciones de complicación sean el conjunto de restricciones del problema, este problema dual es el problema dual del problema original.

Se considera el siguiente problema de optimización lineal:

⁹Por relajación lineal de un problema P se entiende el problema P en el que las variables enteras son sustituidas por variables continuas.

$$\begin{aligned}
& \underset{x}{\text{mín}} c^T x \\
& Ax = b \\
& Dx \leq d \\
& x \geq 0
\end{aligned} \tag{25}$$

y supongamos que $Ax = b$ son el conjunto de restricciones de complicación. Supondremos que $A \in \Re^{m_1 \times n}$, $D \in \Re^{m_2 \times n}$ y que el resto de dimensiones son conformes con éstas.

Dado $\lambda \in \Re^{m_1}$ se formula la función de Lagrange $L(x, \lambda)$ como:

$$L(x, \lambda) = c^T x + \lambda^T (Ax - b) \tag{26}$$

y la función dual $\omega(\lambda)$ como evaluación del siguiente problema, *subproblema de Lagrange*:

$$\begin{aligned}
\omega(\lambda) = \underset{x}{\text{mín}} c^T x + \lambda^T (Ax - b) \\
Dx \leq d \\
x \geq 0
\end{aligned} \tag{27}$$

Para todo $\lambda \in \Re^{m_1}$ es inmediato observar que $\omega(\lambda) \leq c^T x^*$, siendo x^* la solución óptima del problema (25). El problema dual consiste en buscar el vector de multiplicadores λ para el cual esta cota inferior dada por la función dual es máxima:

$$\begin{aligned}
& \underset{\lambda}{\text{máx}} \omega(\lambda) \\
& \lambda \in \Re^{m_1}
\end{aligned} \tag{28}$$

La función dual es cóncava independientemente de que el problema original sea lineal o lineal entero mixto. Por esta razón, las técnicas basadas en subgradientes aparecen de modo natural a la hora de maximizar la función dual [11, 12]. Estas técnicas actualizan los valores de los multiplicadores siguiendo la dirección del subgradiente, utilizando diferentes longitudes de paso en cada iteración [1]. Nótese que si x^j es la solución óptima del subproblema de Lagrange para un valor λ^j del multiplicador, $Ax^j - b$ es un subgradiente de la función dual. Una alternativa a las técnicas basadas en el subgradiente es la representación lineal del problema dual, lo que induce de modo natural un algoritmo de aproximación exterior similar al algoritmo de descomposición de Benders. La próxima sección presenta este algoritmo que posteriormente es formulado en GAMS.

La Relajación Lagrangiana recibe también el nombre de *descomposición dual* o *descomposición por precios* porque se basa en la evaluación del subproblema para distintos precios o variables duales.

Se describe a continuación la formulación del algoritmo de aproximación exterior de la función dual, que denominaremos algoritmo de descomposición basado en Relajación Lagrangiana.

3.1. Representación lineal del problema dual

Consideremos de nuevo el subproblema de Lagrange que evalúa la función dual $\omega(\lambda)$:

$$\begin{aligned} \omega(\lambda) = \min_x & c^T x + \lambda^T (Ax - b) \\ & Dx \leq d \\ & x \geq 0 \end{aligned} \quad (29)$$

En caso de disponer de forma explícita del conjunto de soluciones factibles de la región $\{Dx \leq d, x \geq 0\}$, el anterior problema podría ser resuelto mediante enumeración de todas ellas como:

$$\omega(\lambda) = \min c^T x^l + \lambda^T (Ax^l - b) \quad l = 1, \dots, \nu \quad (30)$$

La expresión anterior indica que la función dual es cóncava y que el problema dual puede reformularse como el siguiente problema lineal, denominado *problema maestro de la Relajación Lagrangiana*:

$$\begin{aligned} & \text{máx } \omega \\ & \omega \leq c^T x^1 + \lambda^T (Ax^1 - b) \\ & \vdots \\ & \omega \leq c^T x^\nu + \lambda^T (Ax^\nu - b) \end{aligned} \quad (31)$$

Cada restricción del problema anterior se denomina *corte de Lagrange*. Dado que para problemas realistas disponer del conjunto de soluciones es prácticamente imposible, la optimización del problema dual se reemplaza por la resolución iterativa de problemas maestros relajados¹⁰, cuyo número de cortes de Lagrange aumenta con cada iteración. Cada resolución de un problema maestro relajado propone un nuevo valor del multiplicador λ que, una vez evaluado en la función dual a través del subproblema de Lagrange, propone un nuevo corte de Lagrange que aumenta el problema maestro relajado¹¹.

De modo similar a la posibilidad de subproblemas infactibles en el algoritmo de descomposición de Benders, el algoritmo de descomposición debe afrontar la posibilidad de encontrar subproblemas de Lagrange no acotados. Cuando esto ocurre, el multiplicador propuesto no es válido y se debe introducir una restricción en el problema maestro que lo elimine del conjunto de multiplicadores posibles. Este tipo de corte se denomina *corte de acotamiento* y su construcción se comenta en la siguiente sección.

¹⁰Relajado hace referencia a que no se dispone de la totalidad de los cortes, sino solamente de un conjunto de ellos

¹¹Por simplicidad de la exposición, se denomina como problema maestro al problema maestro relajado.

3.2. Cortes de acotamiento

Consideremos un valor del multiplicador λ y consideremos de nuevo el subproblema de Lagrange

$$\begin{aligned} \omega(\lambda) = \min_x & c^T x + \lambda^T (Ax - b) \\ & Dx \leq d \\ & x \geq 0 \end{aligned} \quad (32)$$

Supongamos que para este valor λ del multiplicador existe $v \geq 0$ y $Dv \leq 0$ ¹² tal que $c^T + \lambda^T A < 0$. Entonces, si x_0 es una solución del subproblema de Lagrange, observando que $tv \geq 0$ y $Dtv \leq 0$ para cualquier valor positivo de t se tiene que

$$c^T(x_0 + tv) + \lambda^T(A(x_0 + tv) - b) = c^T x_0 + \lambda^T A(x_0 - b) + t(c^T + \lambda^T A)v \quad (33)$$

con lo que el subproblema no está acotado dado que toma valores muy pequeños cuando t tiende a ∞ .

Del razonamiento anterior se deduce que el conjunto de multiplicadores aceptables debe verificar que

$$(c^T + \lambda^T A)x \geq 0 \quad \forall x \in \{x \geq 0, Dx \leq 0\} \quad (34)$$

La condición anterior, una vez propuesto un valor para el multiplicador, puede ser comprobada resolviendo el siguiente problema, que denominamos *subproblema de acotamiento*:

$$\begin{aligned} \omega^*(\lambda) = \min_x & c^T x + \lambda^T Ax \\ & Dx \leq 0 \\ & 0 \leq x \leq 1 \end{aligned} \quad (35)$$

y en caso de que tenga un valor negativo, se debe ser introducir un corte de acotamiento en el problema maestro de la Relajación Lagrangiana de la forma:

$$0 \leq c^T x^j + \lambda^T Ax^j \quad (36)$$

De este modo, el algoritmo de la Relajación Lagrangiana itera entre un problema maestro, formado por cortes de Lagrange y cortes de acotamiento, y un subproblema de Lagrange que evalúa los multiplicadores propuestos por el maestro. El problema maestro, considerando ambos tipos de cortes, puede ser formulado de la siguiente forma

$$\begin{aligned} & \text{máx } \omega \\ & \delta^l \omega \leq c^T x^l + \lambda^T (Ax^l - \delta^l b) \quad l = 1, \dots, j \end{aligned} \quad (37)$$

¹²Es decir, $v \in \{x \geq 0, Dx \leq 0\}$ que es el sistema homogéneo asociado a $\{x \geq 0, Dx \leq d\}$

considerando $\delta^l = 1$ para los cortes de Lagrange y $\delta^l = 0$ para los cortes de acotamiento. Con esta formulación, el algoritmo de Relajación Lagrangiana se describe esquemáticamente en la siguiente sección.

3.3. Algoritmo de Relajación Lagrangiana

El algoritmo de la Relajación Lagrangiana, considerando la interpretación lineal del problema dual anterior, es descrito en los siguientes pasos.

1. Inicialización: $j = 0$, $\epsilon = 10^{-4}$
2. Resolución del problema maestro de la Relajación Lagrangiana

$$\begin{aligned} & \underset{\lambda, \omega}{\text{máx}} \omega \\ & \delta^l \omega \leq c^T x^l + \lambda^T (Ax^l - \delta^l b) \quad l = 1, \dots, j \end{aligned} \quad (38)$$

Obtener el valor de λ e ir al paso 3.

3. Resolución del problema de acotamiento

$$\begin{aligned} \omega^*(\lambda) = \underset{x}{\text{mín}} \quad & c^T x + \lambda^T Ax \\ & Dx \leq 0 \\ & 0 \leq x \leq 1 \end{aligned} \quad (39)$$

Si $\omega^*(\lambda) \geq 0$ ir al paso 4. En otro caso obtener la solución x^j y formar el corte de acotamiento

$$0 \leq c^T x^j + \lambda^T Ax^j \quad (40)$$

Ir al paso 2.

4. Resolución del subproblema de Lagrange

$$\begin{aligned} \omega(\lambda) = \underset{x}{\text{mín}} \quad & c^T x + \lambda^T (Ax - b) \\ & Dx \leq d \\ & x \geq 0 \end{aligned} \quad (41)$$

Obtener la solución x^j y formar el corte de Lagrange

$$\omega \leq c^T x^j + \lambda^T (Ax^j - b) \quad (42)$$

Ir al paso 5.

5. Regla de parada
 Cálculo de la diferencia entre los multiplicadores propuestos en la iteración j y la anterior $j - 1$.
 Si $d(\lambda^j - \lambda^{j-1}) < \epsilon$ detener¹³. En otro caso, ir al paso 2.

¹³ $d(\lambda^j - \lambda^{j-1})$ representa distancia entre λ^j y λ^{j-1}

En el algoritmo de la Relajación Lagrangiana, los multiplicadores λ en el problema maestro están acotados por valores suficientemente amplios que impiden que éste sea no acotado. El problema lineal resuelto por Benders es ahora resuelto mediante el algoritmo de Relajación Lagrangiana para ver en detalle los pasos del algoritmo.

Cuando la Relajación Lagrangiana se aplica a problemas lineales, la sucesión de los valores óptimos del problema maestro (que siempre es decreciente) converge al valor óptimo del problema lineal. Por el contrario, esto no se satisface cuando el problema es entero mixto. Este valor al que la sucesión decreciente converge es simplemente una cota inferior del valor óptimo del problema. Esta diferencia entre el valor óptimo del problema y el valor obtenido por la Relajación Lagrangiana es lo que se conoce en la literatura como *intervalo de dualidad* (*duality gap*). La aparición del intervalo de dualidad origina que la solución primal propuesta por la Relajación Lagrangiana pueda ser infactible. Una alternativa para evitar esta situación consiste normalmente en realizar un postprocesado de las soluciones obtenidas para encontrar la solución factible. Este postprocesado depende del problema concreto que se resuelva y suele estar basado en el conocimiento específico de dicho problema. Otra alternativa para obtener una solución factible consiste en aumentar la función objetivo del subproblema de Lagrange con un término cuadrático que penalize las infactibilidades asociadas a las ecuaciones relajadas. Esta técnica es conocida como *Relajación Lagrangiana Aumentada* y existen numerosas aplicaciones en la literatura.

3.4. Ejemplo

Se desea resolver el siguiente problema lineal:

$$\begin{aligned} \min_{x_1, x_2, y_1, y_2} \quad & -x_1 - 2x_2 - 2y_1 - 3y_2 \\ & x_1 + x_2 \leq 600 \\ & x_1 - 2x_2 \leq 0 \\ & x_1 + x_2 + y_1 + y_2 \leq 1000 \\ & x_1 + y_1 \leq 500 \\ & -2y_1 + y_2 \leq 0 \\ & x_1, x_2, y_1, y_2 \geq 0 \end{aligned}$$

entonces,

$$c = \begin{pmatrix} -1 \\ -2 \\ -2 \\ -3 \end{pmatrix}, \quad A = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & -2 & 0 & 0 \\ 0 & 0 & -2 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 600 \\ 0 \\ 0 \end{pmatrix},$$

$$D = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \text{ y } d = \begin{pmatrix} 1000 \\ 500 \end{pmatrix}$$

esto es, se está considerando que las ecuaciones $x_1 + x_2 + y_1 + y_2 \leq 1000$ y $x_1 + y_1 \leq 500$ son las de complicación.

Para este ejemplo académico, codificado en GAMS tal como se presenta en la próxima sección, las variables λ que representan los multiplicadores han sido acotadas entre 0 y 10. El algoritmo comienza resolviendo un problema maestro de Relajación Lagrangiana sin ningún corte, proponiendo un primer valor para los multiplicadores dado por

$$\lambda = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Este valor de los multiplicadores no es adecuado y el subproblema de acotamiento lo rechaza introduciendo un corte en el problema maestro dado por

$$0 \leq -5 + 2\lambda_1 + 2\lambda_2$$

Una nueva resolución del problema maestro propone

$$\lambda = \begin{pmatrix} 2,5 \\ 0 \end{pmatrix}$$

y el subproblema de acotamiento

$$0 \leq -4 + 1,5\lambda_1 + 0,5\lambda_2$$

con lo que una nueva resolución del maestro obtiene

$$\lambda = \begin{pmatrix} 2,667 \\ 0 \end{pmatrix}$$

Este valor de multiplicadores es aceptado por el problema de acotamiento y el algoritmo pasa a resolver el subproblema de Lagrange. Una vez resuelto, éste propone un corte de Lagrange para el problema maestro que es

$$\omega \leq -1000\lambda_1 - 500\lambda_2$$

La nueva resolución del maestro propone el mismo multiplicador de Lagrange, por lo que el algoritmo termina.

Para problemas lineales, el algoritmo de Relajación Lagrangiana termina con una solución que es óptima para el dual del problema lineal. Esta solución viene dada como el opuesto de este multiplicador óptimo. Este hecho puede apreciarse en este ejemplo. Los valores de las variables duales de las ecuaciones relajadas son (cuando se resuelve el problema completo):

$$\pi = \begin{pmatrix} -2,667 \\ 0 \end{pmatrix}$$

Una desventaja de este método de descomposición es la pérdida de factibilidad de la solución primal durante las sucesivas iteraciones del método. En este ejemplo, la última solución primal, aunque factible $(x_1 \ x_2 \ y_1 \ y_2) = (0 \ 0 \ 0 \ 0)$ no es óptima. Sin embargo, puede comprobarse que la solución óptima, $(x_1 \ x_2 \ y_1 \ y_2 \ x_4) = (0 \ 0 \ 333,333 \ 666,667)$ es una solución alternativa del último subproblema de Lagrange resuelto.

Las próximas secciones presentan la implantación en GAMS del algoritmo de descomposición basado en Relajación Lagrangiana para el ejemplo académico anterior y para el problema de coste fijo que anteriormente ha sido resuelto mediante la descomposición de Benders.

3.5. Relajación Lagrangiana en GAMS

La implantación en GAMS de un código de Relajación Lagrangiana es similar a la implantación de la descomposición de Benders. Se formula un modelo para el problema maestro y otro para el subproblema y éstos son resueltos sucesivamente en un bucle de iteraciones. Las ecuaciones que representan los cortes de Lagrange (una única ecuación que engloba a los dos tipos de cortes) es declarada sobre un conjunto estático de índices y construida sobre un subconjunto dinámico. Durante el proceso algorítmico, este conjunto dinámico se va actualizando, aumentando por tanto el número de cortes de Lagrange que contiene el problema maestro.

```

$TITLE Relajación Lagrangiana en GAMS (RL)

OPTION OPTCR = 0

SETS
  J           índice de iteraciones /j1 * j100 /
  JJ(j)      subconjunto de índices
  I           índice de variables /i1 * i4 /
  M           índice de restricciones /m1 * m2/
  N           índice de restricciones /n1 * n3/

* Datos del problema

TABLE A(m,i) matriz A
  i1 i2 i3 i4
m1  1  1  1  1
m2  1  0  1  0

TABLE D(n,i) matriz D
  i1 i2 i3 i4
n1  1  1
n2  1 -2
n3          -2  1

PARAMETERS
  C(i)      coeficientes objetivo
            / i1 -1, i2 -2, i3 -2, i4 -3 /
  B(m)      término B
            / m1 1000, m2 500 /
  Dd(n)     matriz d (notación matemática)
            / n1 600, n2 0, n3 0 /

```



```

* Fin datos del problema

SCALARS
  TOL      tolerancia relativa / 1e-6 /
  Z_INF    cota inferior      / -INF /
  Z_SUP    cota superior      /  INF /
  DELTA    0 para subproblema de acotamiento 1 para Lagrange
  DIF      diferencia en iteraciones

PARAMETERS
  W_J(j)   función objetivo del problema dual   en la iteración j
  X_J(i,j) valores de las variables de flujo   en la iteración j
  DELTAJ(j) tipo de corte (acotación 0 óptimo 1) de la iteración j
  LAMBDA_J(m,j) multiplicadores                en la iteración j

POSITIVE VARIABLES
  X(i)     variables
  LAMBDA(m) multiplicador

VARIABLES
  Z        variable objetivo primal (subproblema)
  W        variable dual

* Declaración de las ecuaciones
EQUATIONS
  EQ_OBJ    función objetivo problema
  EQ_R1(m)  restricciones que se relajan
  EQ_R2(n)  resto de restricciones
  CORTES_LR(j) cortes de Relajación Lagrangiana
  CORTE_RE  corte redundante;

EQ_OBJ .. Z =E= SUM(i, C(i)*X(i))
          + SUM(m, LAMBDA.L(m)*(SUM(i,A(m,i)*X(i))-DELTA*B(m))) ;

EQ_R1(m) .. SUM(i,A(m,i)*X(i)) =L= B(m) ;

EQ_R2(n) .. SUM(i,D(n,i)*X(i)) =L= DELTA*Dd(n) ;

CORTES_LR(jj) .. DELTAJ(jj)*W =L= SUM(i, C(i)*X_J(i,jj)) +
                SUM(m, LAMBDA(m)*(SUM(i,A(m,i)*X_J(i,jj))-DELTAJ(jj)*B(m))) ;

CORTE_RE .. W =L= 10000 ;

MODEL MAESTRO_LR / CORTES_LR , CORTE_RE / ;
MODEL SUB_LR    / EQ_OBJ,      EQ_R2 / ;
MODEL COMPLETO  / EQ_OBJ, EQ_R1, EQ_R2 / ;

FILE COPT / cplex.opt /

PUT COPT PUT 'scaind -1' / 'lpmethod 1' / 'preind 0' / 'epopt 1.1e-9' / 'eprhs 1.1e-9'
PUTCLOSE COPT ;

SUB_LR.OPTFILE = 1 ; COMPLETO.OPTFILE = 1 ;

* inicialización de parámetros del problema

JJ(j)      = NO ;
DELTAJ(j)  = 0 ;
DELTA      = 0 ;
X_J(i,j)   = 0 ;
LAMBDA.LO(m) = 0 ;
LAMBDA.UP(m) = 10 ;
DIF = INF;

* Iteraciones del algoritmo
LOOP(j $(DIF>TOL),

```

```

* Resolución del problema maestro de Relajación Lagrangiana
IF(ORD(j) > 1,
  SOLVE MAESTRO_LR USING LP MAXIMIZING W;
  W_J(j) = W.L;
ELSE
  LAMBDA.L(m) = 0 ;
) ;
LAMBDA_J(m,j) = LAMBDA.L(m) ;

* Resolución del subproblema de acotamiento
* Normalización de las cotas no nulas de las variables
X.UP(i) = 1 ;
DELTA = 0 ;
SOLVE SUB_LR USING RMIP MINIMIZING Z;

* Si la solución es negativa, generar un corte de acotamiento
IF(Z.L < -TOL,
  X_J(i,j) = X.L(i) ;
  DELTAJ(j) = 0 ;

* En caso contrario resolver el subproblema de Relajación Lagrangiana
ELSE
  X.UP(i) = INF ;
  DELTA = 1 ;
  SOLVE SUB_LR USING RMIP MINIMIZING Z;
  X_J(i,j) = X.L(i) ;
  DELTAJ(j) = 1 ;
) ;

* Actualización del conjunto de cortes de Lagrange
JJ(j) = YES;

* Actualización de la diferencia de multiplicadores
IF(ORD(j)>1,
  DIF = SUM(m, MAX(LAMBDA_J(m,j)-LAMBDA_J(m,j-1),
                  LAMBDA_J(m,j-1)-LAMBDA_J(m,j))) ;
) ;
) ;

X.UP(i) = INF ;
LAMBDA.FX(m) = 0 ;
DELTA = 1 ;
SOLVE COMPLETO USING RMIP MINIMIZING Z ;

```

Debe destacarse la necesidad de resolver independientemente el subproblema de acotamiento y el subproblema de Lagrange. En el código presentado esto se ha simplificado introduciendo un scalar (`DELTA`) que, declarado como 0 o bien como 1, genera el subproblema de acotamiento o el subproblema de Lagrange respectivamente. Otra característica de este código es la introducción de una ecuación redundante para el problema maestro $\omega \leq 10000$. Esta ecuación es introducida en el maestro para que la variable objetivo que es maximizada, ω , no desaparezca del problema. La desaparición de la variable objetivo de un problema es interpretado por este lenguaje como un error.

En la siguiente sección se presenta un código en GAMS orientado a resolver el problema de coste fijo anteriormente descrito mediante Relajación Lagrangiana.

3.6. Problema de coste fijo mediante Relajación Lagrangiana

Consideremos el problema de transporte con coste fijo de la sección 2 y supongamos que son las restricciones de capacidad límite las que son relajadas e introducidas en la función objetivo a través del lagrangiano. Para un valor λ_{ij} el subproblema de Lagrange presenta la forma:

$$\begin{aligned} & \min_{x_{ij}} \sum_{ij} (c_{ij}x_{ij} + f_{ij}y_{ij}) + \lambda_{ij}(x_{ij} - M_{ij}y_{ij}) \\ & \sum_j x_{ij} \leq a_i \\ & \sum_i x_{ij} \geq b_j \\ & x_{ij} \geq 0, y_{ij} \in \{0, 1\} \end{aligned} \tag{43}$$

que reformulado queda como

$$\begin{aligned} & \min_{x_{ij}} \sum_{ij} (c_{ij} + \lambda_{ij})x_{ij} + (f_{ij} - \lambda_{ij}M_{ij})y_{ij} \\ & \sum_j x_{ij} \leq a_i \\ & \sum_i x_{ij} \geq b_j \\ & x_{ij} \geq 0, y_{ij} \in \{0, 1\} \end{aligned} \tag{44}$$

Esta reformulación permite observar que el subproblema es separable en dos problemas. Un problema de transporte en el que el coste variable es ligeramente modificado por el multiplicador y un segundo problema que es entero puro y cuya solución puede obtenerse de modo inmediato.

Problema de transporte

$$\begin{aligned} & \min_{x_{ij}} \sum_{ij} (c_{ij} + \lambda_{ij})x_{ij} \\ & \sum_j x_{ij} \leq a_i \\ & \sum_i x_{ij} \geq b_j \\ & x_{ij} \geq 0 \end{aligned} \tag{45}$$

Problema entero puro

$$\begin{aligned} & \min_{y_{ij}} \sum_{ij} (f_{ij} - \lambda_{ij}M_{ij})y_{ij} \\ & y_{ij} \in \{0, 1\} \end{aligned} \tag{46}$$

Esta formulación de la Relajación Lagrangiana para el problema del transporte con coste fijo puede implantarse en GAMS con el siguiente código.

```

$title Problema de transporte con coste fijo (RL)

option optcr = 0

sets
  J      índice de iteraciones / j1 * j100 /
  JJ(j) subconjunto de índices
  I      orígenes           / i1 * i4 /
  L      destinos           / l1 * l3 /

* Datos del problema
    
```

```

PARAMETERS
  A(i)      ofertas de producto
            / i1 10, i2 30, i3 40, i4 20 /
  B(1)      demandas de producto
            / 11 20, 12 50, 13 30 /

TABLE C(i,1) coste variable unitario de transporte
      11 12 13
i1    1  2  3
i2    3  2  1
i3    2  3  4
i4    4  3  2

TABLE F(i,1) coste fijo de transporte
      11 12 13
i1   10 20 30
i2   20 30 40
i3   30 40 50
i4   40 50 60

* Fin datos del problema

ABORT $(SUM[i, A(i)] < SUM[1, B(1)]) 'Problema infactible'

SCALARS
  TOL      tolerancia relativa / 1e-6 /
  Z_INF    cota inferior      / -INF /
  Z_SUP    cota superior      /  INF /
  DIF      diferencia en iteraciones

PARAMETERS
  X_J(i,1,j)  valores de las variables de flujo      en la iteración j
  Y_J(i,1,j)  valores de las variables de inversión en la iteración j
  DELTA(j)    tipo de corte (acotación 0 óptimo 1) de la iteración j
  LAMBDA_J(i,1,j) multiplicadores                    en la iteración j

POSITIVE VARIABLES
  X(i,1)      flujo por los arcos
  LAMBDA(i,1) multiplicador

BINARY VARIABLE
  Y(i,1)      decisiones de inversion en los arcos

VARIABLES
  Z           variable objetivo primal (subproblema)
  W           variable dual

* Declaración de las ecuaciones
EQUATIONS
  EQ_OBJ      función objetivo problema completo
  OFERTA(i)   ofertas de los origenes
  DEMANDA(1)  demanda de los destinos
  LIMITE(i,1) límite de uso del arco
  CORTES_LR(j) cortes de Relajación Lagrangiana ;

EQ_OBJ      .. Z =E= SUM[(i,1), F(i,1)*Y(i,1)] + SUM[(i,1), C(i,1)*X(i,1)]
              + SUM[ (i,1), LAMBDA.L(i,1)*(X(i,1)-MIN[A(i),B(1)] * Y(i,1))];

OFERTA(i)   .. SUM[1, X(i,1)] =L= A(i) ;

DEMANDA(1)  .. SUM[i, X(i,1)] =G= B(1) ;

LIMITE(i,1) .. X(i,1) =L= MIN[A(i),B(1)] * Y(i,1) ;

CORTES_LR(jj) .. DELTA(jj)*W =L=
              SUM[(i,1), F(i,1)*Y_J(i,1,jj)] + SUM[(i,1), C(i,1)*X_J(i,1,jj)] +
              SUM[(i,1), LAMBDA(i,1)*(X_J(i,1,jj)-MIN[A(i),B(1)] * Y_J(i,1,jj))];

```

```

MODEL MAESTRO_LR / CORTES_LR / ;

MODEL SUB_LR      / EQ_OBJ, OFERTA, DEMANDA / ;

MODEL COMPLETO    / EQ_OBJ, OFERTA, DEMANDA, LIMITE / ;

X.UP(i,l) = MIN[A(i),B(1)] ;

FILE COPT / cplex.opt / ;

* para que los resultados de un problema infactible sean los correctos con CPLEX sólo
* se puede utilizar el método simplex y sin preproceso

PUT COPT PUT 'scaind -1' / 'lpmethod 1' / 'preind 0' / 'epopt 1.1e-9' / 'eprhs 1.1e-9'
PUTCLOSE COPT ;

SUB_LR.OPTFILE = 1 ; COMPLETO.OPTFILE = 1 ;

* inicialización de parámetros del problema

JJ(j)          = NO ;
DELTA(j)       = 0 ;
Y_J(i,l,j)    = 0 ;
X_J(i,l,j)    = 0 ;
LAMBDA.L0(i,l) = 0 ;
LAMBDA.UP(i,l) = 10 ;
DIF = INF;

* Iteraciones del algoritmo
LOOP(j $(DIF>TOL),

* Resolución del problema maestro de Relajación Lagrangiana
IF(ORD(j) > 1,
  SOLVE MAESTRO_LR USING LP MAXIMIZING W;
ELSE
  LAMBDA.L(i,l) = 0 ;
) ;
LAMBDA_J(i,l,j) = LAMBDA.L(i,l) ;

* Resolución del subproblema de acotamiento
* Normalización de las cotas no nulas de las variables
X.UP(i,l) = 1 ;
SOLVE SUB_LR USING RMIP MINIMIZING Z ;

* Si la solución es negativa, generar un corte de acotamiento
IF(Z.L < 0,
  X_J(i,l,j) = X.L(i,l) ;
  Y_J(i,l,j) = Y.L(i,l) ;
  DELTA(j)   = 0 ;

* En caso contrario resolver el subproblema de Relajación Lagrangiana
ELSE
  X.UP(i,l) = MIN[A(i),B(1)] ;
  SOLVE SUB_LR USING MIP MINIMIZING Z;
  X_J(i,l,j) = X.L(i,l) ;
  Y_J(i,l,j) = Y.L(i,l) ;
  DELTA(j)   = 1 ;
) ;

* Actualización del conjunto de cortes de Lagrange
JJ(j) = YES;

* Actualización de la diferencia de multiplicadores
IF(ORD(j)>1,
  DIF = SUM((i,l), MAX(LAMBDA_J(i,l,j)-LAMBDA_J(i,l,j-1),
                     LAMBDA_J(i,l,j-1)-LAMBDA_J(i,l,j))) ;

```

```

) ;
) ;
X.UP(i,1) = MIN[A(i),B(1)] ;
LAMBDA.FX(i,1) = 0 ;
SOLVE COMPLETO USING MIP MINIMIZING Z ;

```

En este ejemplo, la sucesión de valores del problema maestro y la sucesión de óptimos de los subproblemas convergen a la solución dada por la relajación lineal del problema, véase la figura 2. En este caso, la cota inferior del valor óptimo del problema dada por la relajación lineal del mismo no puede ser superada por la aplicación de la Relajación Lagrangiana¹⁴. Por otra parte, los valores obtenidos por las variables de decisión no son factibles en ninguna de las iteraciones del método. Se necesita un método de postprocesado de estas soluciones para obtener una solución. En este problema, una posibilidad consiste en considerar la solución dada para las variables continuas x_{ij} por el problema de transporte (lineal) y adecuar las variables binarias a esta solución, es decir

$$u_{ij} = 1 \text{ si } x_{ij} > 0$$

Con este procesado de las soluciones se puede comprobar que en la iteración 51 (el algoritmo converge en la iteración 55) la solución obtenida es, en este caso, la solución óptima del problema entero mixto:

$$y_{11} = y_{23} = y_{31} = y_{32} = y_{42} = 1$$

4. Implantación en grandes modelos

GAMS fue desarrollado para permitir la implantación rápida de modelos de optimización concentrando la labor del modelador en su formulación. Además, dada la potencia del lenguaje permite la creación de modelos complejos con un número reducido de instrucciones. Esto hace que el mantenimiento o la modificación de un modelo sea muy sencilla. De la misma manera estas ventajas se extienden a la escritura de métodos de descomposición. Sin embargo, el tiempo de resolución de un problema en GAMS por métodos de descomposición es muy elevado. Las razones son las siguientes:

- el paso de un problema definido en GAMS al optimizador encargado de su resolución se hace escribiéndolo en disco en lugar de a través de memoria principal. Esto hace que se consuma mucho tiempo cuando esta tarea se debe hacer repetidas veces para cada maestro o subproblema. Este inconveniente se puede soslayar parcialmente recurriendo a la creación de un disco virtual en memoria principal mediante un controlador RAMDISK.

¹⁴Cuando esto ocurre se dice que el problema satisface la *propiedad de integralidad* (*integrality property*) y tiene poco sentido el uso de la Relajación Lagrangiana como técnica de aproximación del valor óptimo del problema.

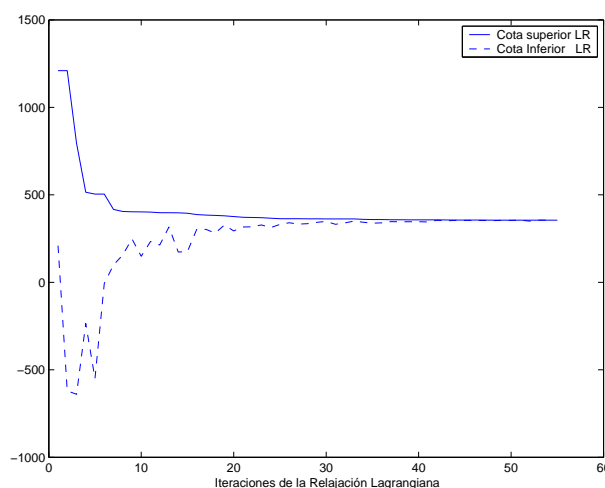


Figura 4. Evolución de cotas en Relajación Lagrangiana

- cada maestro o subproblema se crea de nuevo cada vez aunque las modificaciones entre una iteración y la siguiente para cada problema son menores. Precisamente esta idea de resolución iterativa de problemas muy similares es el núcleo central de la descomposición y es la que se debería poder aprovechar computacionalmente tanto en la fase de creación del problema como en el algoritmo de resolución (mediante el simplex dual). GAMS genera desde cero cada problema aunque sí permite guardar la base previa.

En conclusión, GAMS puede ser utilizado para el desarrollo de prototipos de modelos de optimización y de métodos de descomposición. Sin embargo, una vez validados estos prototipos se deben utilizar lenguajes alternativos de menores tiempos de computación.

Referencias

1. Bazaraa, M.S. "On the choice of step sizes in subgradient optimization" *Journal of Operational Research*. Vol 7, pp 380-388. 1981.
2. Benders, J.F. "Partitioning Procedures for Solving Mixed-Variable Programming Problems" *Numerische Mathematik*. Vol 4, pp 238-252. 1962.
3. Birge, J.R. and Louveaux, F. (1997) *Introduction to Stochastic Programming*. Springer-Verlag.
4. Brooke, A., Kendrick, D., Meeraus, A. and Raman, R. *GAMS A User's Guide*. GAMS Development Corporation. December, 1998.
5. Ermoliev, Y. and Wets, R.J-B. (eds.) *Numerical Techniques for Stochastic Optimization*. Springer-Verlag. Berlin, Germany. 1988.
6. Geoffrion, A.M. "Elements of Large-Scale Mathematical Programming" *Management Science*. Vol 16, No 11, pp 652-691. July 1970.

7. Geoffrion, A.M. "Lagrangean relaxation for integer programming" *Mathematical Programming Study*. Vol 2, pp 82-114. 1974.
8. ILOG "<http://www.ilog.com>"
9. G. Infanger (1994) *Planning Under Uncertainty—Solving Large-Scale Stochastic Linear Programs*, The Scientific Press Series, Boyd and Fraser.
10. Kall, P. and Wallace, S.W. (1995) *Stochastic Programming*. John Wiley and Sons.
11. Kiwiel, K. C. "Proximity control in bundle methods for convex nondifferentiable minimization" *Mathematical Programming*. Vol 46, pp 105-122. 1990.
12. Jiménez, N. "Coordinación Hidrotérmica en el Corto Plazo Mediante Técnicas de Relajación Lagrangiana" *Tesis doctoral, Universidad de Málaga* Málaga, ESpaña. 1999.
13. Van Slyke, R.M. and Wets, R.J.-B. "L-Shaped Linear Programs with Application to Optimal Control and Stochastic Programming" *SIAM Journal on Applied Mathematics*. Vol 17, No 4, pp 638-663. July 1969.