2j·32

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE CIENCIAS

UN CODIGO INTERACTIVO PARA EL METODO DE DESCOMPOSICION DE DANTZIG Y WOLFE

T E S I S

QUE PARA OBTENER EL TITULO DE:

A C T U A R I O

P R E S E N T A :

JOSE EFREN MARTINEZ ROMERO

MEXICO, D. F. 1988

I N D I C E

INTRODUCCION

En la práctica, muchos problemas de programación lineal son demasiado grandes para ser resueltos por una computadora. A pesar de que teóricamente es posible resolver cualquier problema de programación lineal, surgen ciertas limitantes como el problema de dimensionalidad del problema, y casi todas las dificultades que surgen en el desarrollo de un problema de programación lineal, están relacionados con su dimensión.

Es frecuente que el modelo administrativo de una corporación produzca un programa lineal con muchos miles de renglones y un número interminable de columnas. Que comprende la coordinación de las decisiones de las divisiones "separadas" de una gran organización. Debido a que las divisiones operan con considerable autonomía el problema se puede descomponer en problemas separados, en los que cada división se interesa únicamente en optimizar su propia operación. No obstante se requiere cierta coordinación global para dividir de la mejor manera determinados recursos de la organización entre las divisiones.

En tales problemas se debe de aplicar algún método para convertir los problemas grandes en otros mas pequeños de tamaño mas manejable. Uno de los métodos a aplicar es el "Principio de Descomposición", que hace precisamente esto. Con tal método se puede separar el problema lineal en dos partes: una parte con una estructura general y otra con una estructura especial en la que se puede aplicar un método mas eficiente.

Las restricciones se dividen en dos conjuntos: restricciones generales y restricciones con estructura especial (estructura angular). Aunque despues sera claro que no necesariamente se debe de tener una estructura especial, al disponer de está, se realza la eficiencia del Principio de Descomposición.

El Principio de Descomposición consiste en operar sobre dos programas lineales separados: uno sobre el conjunto de restricciones generales y otro sobre el conjunto de restricciones especiales. La información se pasa entre uno y otro de los programas lineales hasta alcanzar un punto en el que se tiene la solución del problema original.

El programa lineal sobre las restricciones generales se llama el Programa Maestro y los programas lineales sobre las restricciones especiales se llaman subproblemas. El programa maestro le pasa un nuevo conjunto de coeficientes de costo a cada subproblema y recibe una nueva solución basada en tales coeficientes.

Conceptualmente la aplicación del "Principio de Descomposición", puede imaginarse como si cada división tuviera que resolver su subproblema y enviar está solución como su proposición al "centro de operaciones" (el problema maestro), en donde las negociaciones entonces coordinan las proposiciones de todas las divisiones con el fin de hallar una solución óptima para la organización como un todo.

Para introducir una situación tipica que sugiere la aplicación del Principio de Descomposición, consideremos el problema al que se enfrenta el gerente de una planta con dos tiendas "casi" independientes. Dentro de cada tienda, se tienen muchas restricciones las cuales no son afectadas por las actividades de la otra tienda, pero se tienen unas cuantas restricciones y una función objetivo que enlaza a las dos tiendas. El problema del gerente puede ser formulado en términos de un problema de programación lineal como sigue:

$$\text{Max } Z = P1X + P2Y \qquad (1)$$

s.a.

$$\begin{aligned} A1X + A2Y &= b \\ B1X \qquad &= b1 \\ B2Y &= b2 \end{aligned}$$

tal que $X \geq 0$, $Y \geq 0$

Donde X es el vector de actividades de la primera tienda; Y es el de la segunda tienda. La primera y segunda linea de (1) expresan la función objetivo y aquellas restricciones que enlazan conjuntamente a las dos tiendas. La tercera linea expresa las restricciones que involucran unicamente a la primera tienda y la cuarta linea representa las restricciones de la segunda tienda.

Observando el problema (1), el gerente estima que debido al tamaño del problema este está fuera de sus recursos para resolverlo ya que B1 y B2 son moderadamente grandes, y juntos hacen un problema que excede la capacidad de la computadora disponible. "Pero lo que yo realmente tengo," reflexiona el gerente "no es un gran problema pero si dos problemas de tamaño moderado, uno para cada tienda. Todo lo que necesito es una forma de separar el problema en dos partes y tener en cuenta sus interconexiones".

Para poder llevar a cabo lo anterior el gerente tendra que:

a) tener dos subproblemas que tengan las partes independientes de cada tienda, y

b) un problema maestro que enlace conjuntamente a los subproblemas.

Lo anterior es posible por medio del Algoritmo de Descomposición. Pero el precio pagado por está descomposición es que el problema maestro y los subproblemas deben de ser resueltos varias veces. Primeramente el problema maestro es resuelto, y de su solución son generadas funciones objetivo para cada uno de los subproblemas. Entonces son resueltos, y de su solución nuevas columnas son generadas para ser agregadas al problema maestro. El proceso es repetido hasta que despues de un número finito de ciclos, una condición de óptimalidad es alcanzada y términa el Principio de Descomposición.

CAPITULO I

DESARROLLO DE LA TEORIA DEL ALGORITMO DE DESCOMPOSICION

1.1 REGION ACOTADA

En está sección se supondra que la región de soluciones factibles X es acotada (la suposición de una región no acotada se tratara en la siguiente sección).

Considerese el siguiente problema de programación lineal:

$$\text{Min (max)} \quad Z = Cx$$

$$\text{sujeto a}$$

$$Ax = b \qquad \text{(p)}$$

$$x >= 0$$

en donde la región de soluciones factibles X, representa un conjunto poliedrico de estructura especial. Esa estructura especial (estructura angular) está en función de la matriz de coeficientes A, que es de la forma:

$$\begin{matrix} A0 & A1 & . & . & . & Ak \\ 0 & B1 & 0 & . & . & 0 \\ 0 & 0 & B2 & 0 & . & 0 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ 0 & 0 & . & . & . & Bk \end{matrix}$$

La cual contiene las restricciones de todos los subproblemas p.

Reescribiendo el problema original (P) se tiene;

$$\text{Min (max)} \quad Z = \sum_{p=0}^{K} CpXp \qquad \text{(a)}$$

s.a.

$$\sum_{p=0}^{K} ApXp = bo \qquad \text{(b)}$$

$$BpXp = bp \qquad \text{(c)}$$

$$p=1,2,..,k$$

Y se debe de encontrar los vectores solución $Xp >= 0$ de los subproblemas p (p=1,2,..,k) que minimizen (màximizen) al problema de programación lineal, donde Ap es una matriz de mo x np , Bp es una matriz de mp x np,Cp es un vector renglòn de np componentes, bp es un vector columna de mp componentes , y Xp es el p-esimo vector solución del vector columna X, es decir $Xp=(Xp^{1},Xp^{2},..,Xp^{Np})$.

De donde el problema tiene $m = \sum_{p=0}^{K} mp$ restricciones y $n = \sum_{p=0}^{K} np$ variables.

Para poder entender lo que es en esencia el Algoritmo de Descomposición, vamos a ver geometricamente un problema de descomposición muy simple:

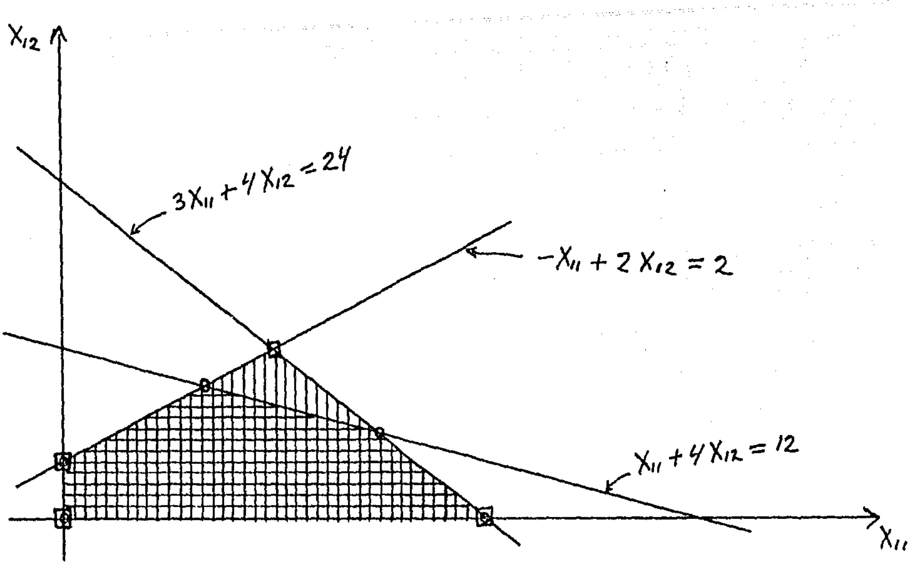
$$\text{Min} \quad Z = C_{11} X_{11} + C_{12} X_{12}$$

s.a.

$$\begin{array}{rcrcll} X_{11} & + & 4X_{12} & \leq & 12 & A1X1 <= bo \\ -X_{11} & + & 2X_{12} & \leq & 2 & \\ 3X_{11} & + & 4X_{12} & \leq & 24 & B1X1 <= b1 \end{array}$$

$$X_{11} \; , \; X_{12} >= 0$$

El área de soluciones factibles al problema están mostradas en la siguiente grafica:



El área rayada verticalmente es el espacio de soluciones a $B1X1 <= b1$ y el área cuadriculada es el espacio de soluciones al problema completo. Los puntos extremos de { X1 | B1X1 <= b1, X1>= 0} están indicados por cuadrados, mientras que los puntos extremos del problema completo están indicados por circulos.

Cualquier solución del problema completo debe tambien satisfacer $B1X1 <= b1$ aunque la implicación al reves no es cierta, es decir, cualquier solución de $B1X1 <= b1$ no necesariamente satisface al problema completo.

Puesto que el conjunto solución a las restricciones lineales del problema completo es un poliedro acotado, entonces cualquier solución $X1>= 0$ a $B1X1 <= b1$ se puede representar como una combinación convexa del número finito de puntos extremos de su conjunto solución.
Por lo que si conocemos los puntos extremos de {X1 | B1X1 <= b1, X1> 0} o tenemos los necesarios, todo lo que tenemos que hacer es

encontrar la combinación lineal convexa de estos puntos que satisfaga $A1X1 <= bo$ y optimize la función objetivo.

Lo anterior es en esencia el Algoritmo de Descomposición de Dantzig-Wolfe. En la figura se puede ver que cualquier punto extremo con circulo es tambien una combinación convexa de los puntos extremos con cuadrado y que cualquier punto en el área cuadriculada puede ser expresado como una combinación convexa de los puntos cuadrados.

Tomando en cuenta lo anterior y extendiendo está idea al problema general, de (a) a (c) y suponiendo que para cada subproblema p se tiene disponible el correspondiente conjunto de soluciones Sp a las restricciones $BpXp = bp$, $Xp >= 0$, entonces la solución al problema original puede ser pensada como la selección de una combinación convexa de los puntos extremos de cada subproblema p, de tal manera que satisfaga las restricciones comunes $\sum_{p=0}^{K} ApXp = bo$ y minimize (máximize) a $\sum_{p=0}^{K} CpXp$.

Si nuestro problema original es mirado de la manera expuesta anteriormente, entonces se podra observar que es necesario optimizar un nuevo problema con mo+k restricciones sujeto a la solución de k subproblemas individuales de mp x np en lugar de un gran problema con $\sum_{p=0}^{K}$ mp restricciones.

El nuevo problema a ser considerado es llamado el "Problema Extremo" o "Problema Maestro" y surge de la siguiente manera:

Sean Xp^{j} los puntos extremos de cada subproblema p con j=1,2,..,Np. Donde Xp^{j} son vectores solución para el subproblema p-esimo y Np es el número total de puntos extremos para el subproblema p. Cualquier solución factible Xp para el subproblema p-esimo puede ser expresada como una combinación lineal convexa de los puntos extremos de Sp esto es;

$$X_p = \sum_{j=1}^{N_p} \lambda_p^j X_p^j \qquad \text{(d)}$$

donde

$$\sum_{j=1}^{N_p} \lambda_p^j = 1 \quad y \quad \lambda_p^j \geq 0$$

Se está suponiendo temporalmente que cada Sp es un poliedro acotado.

Sustituyendo la expresión de Xp en (a) y (b) se tiene;

en (a)

$$\sum_{p=0}^{K} C_p X_p = C_0 X_0 + \sum_{p=1}^{K} C_p X_p$$

$$= C_0 X_0 + \sum_{p=1}^{K} C_p \left(\sum_{j=1}^{N_p} \lambda_p^j X_p^j \right)$$

$$= C_0 X_0 + \sum_{p=1}^{K} \sum_{j=1}^{N_p} C_p \lambda_p^j X_p^j$$

en (b)

$$\sum_{p=0}^{K} A_p X_p = b_0 \quad \text{sustituyendo} \quad X_p = \sum_{j=1}^{N_p} \lambda_p^j X_p^j$$

$$\Rightarrow A_0 X_0 + \sum_{p=1}^{K} A_p X_p = b_0$$

$$\Rightarrow A_0 X_0 + \sum_{p=1}^{K} A_p \left(\sum_{j=1}^{N_p} \lambda_p^j X_p^j \right) = b_0$$

$$\Rightarrow A_0 X_0 + \sum_{p=1}^{K} \sum_{j=1}^{N_p} A_p X_p^j \lambda_p^j = b_0$$

Como ya se considero la igualdad (c) al tomar la combinación lineal convexa de los puntos extremos del subproblema p, entonces el problema equivalente es:

encontrar $\lambda_p^j >= 0$, Xo >= 0 $\ni$

$$\text{Min } Z = C_0 X_0 + \sum_{p=1}^{K} \sum_{j=1}^{N_p} (C_p X_p^j) \lambda_p^j$$

s.a.

$$A_0 X_0 + \sum_{p=1}^{K} \sum_{j=1}^{N_p} (A_p X_p^j) \lambda_p^j = b_0$$

$$\sum_{j=1}^{N_p} \lambda_p^j = 1$$

p=1,2,...,k

Definiendo las transformaciones:

$$P_p^j = A_p X_p^j \quad y \quad f_p^j = C_p X_p^j$$

p=1,2,...,k

y sustituyendo se tiene:

encontrar $\lambda_p^j >= 0$, Xo >= 0 $\ni$

$$\text{Min(max) } Z = C_0 X_0 + \sum_{p=1}^{K} \sum_{j=1}^{N_p} f_p^j \lambda_p^j \qquad \text{(e)}$$

s.a.

$$A_0 X_0 + \sum_{p=1}^{K} \sum_{j=1}^{N_p} P_p^j \lambda_p^j = b_0 \qquad \text{(f)}$$

$$\sum_{j=1}^{N_p} \lambda_p^j = 1 \qquad \text{(g)}$$

p=1,2,...,k

Este es el "problema maestro" o "problema extremo". En (f) y (g) las columnas $\begin{pmatrix} A_0^j \\ 0 \end{pmatrix}$ son las columnas naturales y las columnas $\begin{pmatrix} P_p^j \\ e_p \end{pmatrix}$ son las columnas extremas, en donde e_p es la p-esima columna de una matriz unitaria de orden k.

Se debe de enfatizar que de (e) a (g) es la representación del problema de (a) a (c) en términos de todos los puntos extremos de los k subproblemas.

Las transformaciones P_p^j y f_p^j son la representación de los puntos extremos en términos de su contribución a las restricciones comúnes y la función objetivo respectivamente. Dando los λ_p^j óptimos para el problema maestro (de (e) a (g)), entonces la correspondiente solución óptima para el subproblema p en términos de sus puntos extremos X_p^j está dada por $X_p = \sum \lambda_p^j X_p^j$

Como ejemplo para ilustrar la obtención del problema maestro, consideremos el ejemplo dado anteriormente;

$$
\begin{aligned}
\text{Min} \quad & Z = C_{11} X_{11} + C_{12} X_{12} \\
\text{s.a.} \quad & \\
& X_{11} + 4X_{12} \leq 12 \\
& -X_{11} + 2X_{12} \leq 2 \\
& 3X_{11} + 4X_{12} \leq 24 \\
& X_{11} , X_{12} >= 0
\end{aligned}
$$

Agregando variables de holgura se tiene;

$$\text{Min } Z = 0\,X_{01} + C_{11}\,X_{11} + C_{12}\,X_{12} + 0\,X_{13} + 0\,X_{14}$$

s.a.

$$\begin{array}{rcrcrcrcrcl} X_{01} & + & X_{11} & + & 4X_{12} & & & & & = & 12 \\ & & -X_{11} & + & 2X_{12} & + & X_{13} & & & = & 2 \\ & & 3X_{11} & + & 4X_{12} & & & + & X_{14} & = & 24 \end{array}$$

$$\text{con } X_{ij} >= 0$$

de donde se tiene;

$$A0 = (1)\ ,\ A1 = (1,4,0,0)\ ,\ B1 = \begin{pmatrix} -1 & 2 & 1 & 0 \\ 3 & 4 & 0 & 1 \end{pmatrix}$$

$$bo = (12)\ ,\ b1 = \begin{pmatrix} 2 \\ 24 \end{pmatrix},\ Co = (0)\ ,\ C1 = (C_{11}\ ,\ C_{12}\ ,\ 0\ ,\ 0)$$

Los puntos extremos a $\{X1 \mid B1X1 <= b1\ ,\ X1 >= 0\}$ son;

$$X_1^1 = (0,0,2,24)^t \quad ; \quad X_1^3 = (4,3,0,0)^t$$

$$X_1^2 = (0,1,0,20)^t \quad ; \quad X_1^4 = (8,0,10,0)^t$$

Calculando $P_p^j = ApX_p^j$ y $f_p^j = CpX_p^j$ se tiene;

$$P_1^1 = A1X_1^1 = (1,4,0,0)(0,0,2,24)^t = 0$$

$$P_1^2 = A1X_1^2 = (1,4,0,0)(0,1,0,20)^t = 4$$

$$P_1^3 = A1X_1^3 = (1,4,0,0)(4,3,0,0)^t = 16$$

$$P_1^4 = A1X_1^4 = (1,4,0,0)(8,0,10,0)^t = 8$$

$$f_1^1 = C1X_1^1 = (C_{11}\ ,C_{12}\ ,0,0)(0,0,2,24)^t = 0$$

$$f_1^2 = C1X_1^2 = (C_{11}\ ,C_{12}\ ,0,0)(0,1,0,20)^t = C_{12}$$

$$f_1^3 = C1X_1^3 = (C_{11}\ ,C_{12}\ ,0,0)(4,3,0,0)^t = 4C_{11} + 3C_{12}$$

$$f_1^4 = C1X_1^4 = (C_{11}\ ,C_{12}\ ,0,0)(8,0,10,0)^t = 8C_{11}$$

Queremos obtener el problema maestro es decir;

$$\text{Min } Z = C_oX_o + \sum_{p=1}^{K} \sum_{j=1}^{N_p} f_p^j \lambda_p^j$$

s.a.

$$A_oX_o + \sum_{p=1}^{K} \sum_{j=1}^{N_p} P_p^j \lambda_p^j = b_o$$

$$\sum_{j=1}^{N_p} \lambda_p^j = 1$$

$$p = 1,2,\dots,k$$

lo cual nos expresara el problema original en términos de todos los puntos extremos de S1 y tendremos que encontar los λ_p^j que minimizen el problema.

En términos del problema

$$N1 = 4 \text{ (\# de puntos extremos)}$$

$$k = 1 \text{ (\# de subproblemas)}$$

calculando las partes correspondientes del problema maestro;

$$C_oX_o + \sum_{p=1}^{K} \sum_{j=1}^{N_p} f_p^j \lambda_p^j = C_oX_o + \sum_{p=1}^{1} \sum_{j=1}^{4} f_p^j \lambda_p^j$$

$$= 0 \cdot X_o' + \sum_{p=1}^{1} (f_p^1\lambda_p^1 + f_p^2\lambda_p^2 + f_p^3\lambda_p^3 + f_p^4\lambda_p^4)$$

$$= f_1^1\lambda_1^1 + f_1^2\lambda_1^2 + f_1^3\lambda_1^3 + f_1^4\lambda_1^4$$

$$A_oX_o + \sum_{p=1}^{K} \sum_{j=1}^{N_p} P_p^j \lambda_p^j = A_oX_o + \sum_{p=1}^{1} \sum_{j=1}^{4} P_p^j \lambda_p^j$$

$$= A_oX_o + \sum_{p=1}^{1} (P_p^1\lambda_p^1 + P_p^2\lambda_p^2 + P_p^3\lambda_p^3 + P_p^4\lambda_p^4)$$

$$= 1 \cdot X_o' + P_1^1\lambda_1^1 + P_1^2\lambda_1^2 + P_1^3\lambda_1^3 + P_1^4\lambda_1^4$$

$$\sum_{j=1}^{N_p} \lambda_p^j = \lambda_1^1 + \lambda_1^2 + \lambda_1^3 + \lambda_1^4$$

el problema maestro es;

$$\text{Min } Z = f_1^1 \lambda_1^1 + f_1^2 \lambda_1^2 + f_1^3 \lambda_1^3 + f_1^4 \lambda_1^4$$

s.a.

$$X_0^1 + P_1^1 \lambda_1^1 + P_1^2 \lambda_1^2 + P_1^3 \lambda_1^3 + P_1^4 \lambda_1^4 = 12$$

$$\lambda_1^1 + \lambda_1^2 + \lambda_1^3 + \lambda_1^4 = 1$$

Utilizando las equivalencias numericas calculadas anteriormente se tiene;

$$\text{Min } Z = C_{12} \lambda_1^2 + (4C_{11} + 3C_{12}) \lambda_1^3 + 8C_{11} \lambda_1^4$$

s.a.

$$X_0^1 + 0\lambda_1^1 + 4\lambda_1^2 + 16\lambda_1^3 + 8\lambda_1^4 = 12$$

$$0\lambda_1^1 + \lambda_1^2 + \lambda_1^3 + \lambda_1^4 = 1$$

el cual es el "Problema maestro" del problema original.

Como se menciono anteriormente, las transformaciones nos llevan a un problema con mo+k restricciones; las mo de las restricciones comunes (f) y las k restricciones convexas de (g).

Sin embargo el número de variables se ha incrementado al total de puntos extremos del poliedro convexo Sp. Por lo que el número de variables se puede incrementar grandemente y aún mas, si se tuvieran que calcular todos los puntos extremos, la cantidad de calculos necesarios harian prohibitiva la resolución de cualquier problema; aun siendo pequeño.

La ventaja de Algoritmo de Descomposición, es que necesitamos considerar unicamente un pequeño número de ese gran total de puntos extremos, y solo necesitamos la representación explicita de aquellos a ser considerados.

Por lo que se vera el Algoritmo de Descomposición toma en cuenta los elementos basicos del vector de precios (π) y las transformaciones basicas del método simplex revisado.

Se tratará el problema maestro de (e) a (g) como cualquier otro problema de programación lineal, excepto que se debe de tomar en cuenta el hecho que se es incapaz de escribir en forma explicita la matriz del sistema correspondiente a (f). Sin embargo, se asumira que se conoce suficiente acerca del sistema de tal forma que se tiene una solución basica factible al problema maestro.

Está primera solución puede ser obtenida usando una base artificial o basada en la estructura de los subproblemas. Por lo que se es capaz de encontrar una solución de puntos extremos para los subproblemas, los cuales combinados nos dan una solución factible inicial para el problema maestro.

Está base es de orden mo+k y debe de contener al menos una columna extrema de cada espacio de soluciones Sp, esto es contener en total al menos k columnas extremas, y mo columnas adicionales (naturales, extremas, o ambas).

Denotando el vector asociado de multiplicadores simplex por $\pi = (\hat{\pi}, \hat{\hat{\pi}})$ donde $\hat{\pi}$ es asociado con las primeras mo restricciones de (f) y los k componentes del vector renglón $\hat{\hat{\pi}}$ es asociado con las restricciones de (g).

Los componentes de $\hat{\pi}$ son llamados los coeficientes de costo y seran denotados por $\hat{\pi}_i$ y aquellos pertenecientes a

$\tilde{\hat{\pi}}$ seran denotados por $\tilde{\hat{\pi}}_p$.

Para los vectores en la base se debe de tener $f_p^j = \pi P_p^j + \tilde{\hat{\pi}}_p$;
que es el correspondiente $C_j - Z_j = 0$.

Como en el método simplex regular, se necesita determinar si la solución factible inicial puede o no ser incrementada por la introducción de alguna columna natural o extrema.

Obteniendo una columna extrema de algún subproblema p, se tienen los costos relativos (que son $C_j - Z_j$);

$$
\begin{aligned}
\therefore f_p^j - \pi \begin{pmatrix} P_p^j \\ e_p \end{pmatrix} &= f_p^j - (\pi, \hat{\pi}) \begin{pmatrix} P_p^j \\ e_p \end{pmatrix} \\
&= f_p^j - \pi P_p^j - \hat{\pi} e_p \\
&= C_p X_p^j - \pi A_p X_p^j - \tilde{\hat{\pi}}_p \\
&= (C_p - \pi A_p) X_p^j - \tilde{\hat{\pi}}_p \qquad \text{(h)}
\end{aligned}
$$

p=1,2,..,k

Para las columnas naturales, se tiene que para cada j el costo relativo;

$$C_o^j - \pi A_o^j \qquad \text{(i)}$$

donde Co^j son los elementos de Co. Como el problema maestro es un problema de minimización, entonces si todos los costos relativos (h) e (i) son no negativos para todo p y j, y por el criterio de finalización del método simplex, entonces se tiene un óptimo. Como (h) e (i) corresponden al conjunto de los Cj - Zj del algoritmo simplex, y para un problema de minimización se tiene el óptimo si todos los Zj - Cj <= 0 o si todos los Cj - Zj >= 0.

El costo (i) puede ser calculado dado que se conoce Ao^j . Sin embargo para determinar si cualquier costo relativo (h) son negativos, se necesita resolver p problemas de optimización que surgen de la siguiente manera:

Para cualquier subproblema p se necesita conocer si para cualquier conjunto de puntos extremos Xp^{j} se tiene:

$$\text{minimo} \quad [(Cp - \hat{\pi} Ap)Xp^{j} - \hat{\pi} p] \leq 0 \quad p=1,2,..,k \qquad (j)$$

s.a.

$$BpXp = bp$$

$$Xp >= 0$$

Pero (j) establece que para cada subproblema p la cantidad en parentesis es minimizada sobre aquellos puntos extremos Xp^{j} que satisfacen las restricciones del p-esimo subproblema. Si el minimo es no negativo, entonces todos los puntos extremos del p-esimo subproblema son óptimos.

Como la condición (j) es independiente del escalar $\hat{\pi} p$, (j) se reduce a resolver para cada subproblema p, el problema de programación lineal, que es:

$$\text{Min } Z = (Cp - \hat{\pi} Ap)Xp \qquad (k)$$

s.a.

$$BpXp = bp \qquad (l)$$

$$Xp >= 0$$

Dado que la solución a (l) por el mètodo simplex genera únicamente soluciones de puntos extremos, entonces la solución al subproblema (k) y (l) tambien satisfara a (j). Tambien se debe de notar que cada subproblema (k) y (l) es equivalente al subproblema dado, excepto que el correspondiente coeficiente de costo ha sido "reducido" por $\pi \cdot A_p$; los tèrminos $(C_p - \pi A_p)$ son los llamados "costos ajustados". Se puede ver que πA_p es calculado por medio del producto punto de mo elementos y puede ser facilmente calculado.

Si permitimos que X_p^{ℓ} sea una soluciòn òptima a (k) y (l) para un subproblema p dado (X_p^{ℓ} sera punto extremo si Sp es acotado) y sea el correspondiente valor òptimo de la funciòn objetivo denotado por;

$$Z_p^{\ell} = (C_p - \pi A_p) X_p^{\ell}$$

si $Z_p^{\ell} - \tilde{\pi}_p < 0$, entonces la nueva columna extrema formada por la transformaciòn $P_p^{\ell} = A_p X_p^{\ell}$ con su costo asociado $f_p^{\ell} = C_p X_p^{\ell}$ es un candidato a entrar a la base.

El nuevo punto extremo X_p^{ℓ} es llamado un vector propuesto, con el vector transformaciòn P_p^{ℓ} y f_p^{ℓ} su costo asociado.

Dado el conjunto de costos minimos (h) y los costos (i), el mètodo simplex nos indica tomar como la siguiente columna para entar a la base del problema maestro, aquella correspondiente al minimo de (Cj - Zj) que es;

$$\min[\min_{p}(Z_p^{\ell} - \tilde{\pi}_p), \min_{j} (C_o^{j} - \pi A_o^{j})] \qquad (m)$$

donde p recorre sobre el conjunto de subproblemas y j recorre sobre las columnas naturales no basicas.

Si (m) es no negativo, entonces la actual soluciòn basica factible del problema maestro nos da un òptimo para el problema original.

Si (m) es negativo, se selecciona la correspondiente columna natural o extrema para entrar a la base. Para este último caso, se determina el renglón pivote de la manera usual, se determina la nueva π , y se repite el procedimiento hasta que un óptimo es alcanzado.

La solución óptima al problema original está dada por;

$$[\ X_o,\ X_p = \sum_{j} \lambda_p^{j} \cdot X_p^{j} \quad (p=1,2,..,k)]$$

donde la sumatoria es tomada sobre aquellos puntos extremos del p-esimo subproblema, sobre el cual está la solución óptima del problema maestro.

1.2 REGION NO ACOTADA

Para un poliedro Sp' no acotado, el Algoritmo de Descomposición debe modificarse ligeramente. En este caso no es posible representar los puntos en Sp' como una combinación lineal convexa de los puntos extremos, pero si como una combinación convexa de los puntos extremos más una combinación no negativa de las direcciones extremas.

En otras palabras Xp ∈ Sp' si y solo si;

$$X_p = \sum_{j=1}^{t} \lambda_p^j X_p^j + \sum_{j=1}^{l} \mu_p^j d_p^j$$

y

$$\sum_{j=1}^{t} \lambda_p^j = 1$$

$$\lambda_p^j \geq 0 ; p = 1, 2, .., t$$

$$\mu_p^j \geq 0 ; p = 1, 2, .., l$$

de donde X_p^j son los puntos extremos y d_p^j son las direcciones extremas de los subproblemas no acotados.

Para poder obtener el problema maestro se sustituirá la nueva expresión de Xp en (a) y (b), se tiene;

$$\text{en (a)} \quad \sum_{p=0}^{k} C_p X_p = C_o X_o + \sum_{p=1}^{k} C_p X_p$$

$$= C_o X_o + \sum_{p=1}^{k} C_p \left(\sum_{j=1}^{t} \lambda_p^j X_p^j + \sum_{j=1}^{\ell} \mu_p^j d_p^j \right)$$

$$= C_o X_o + \sum_{p=1}^{k} \sum_{j=1}^{t} C_p \lambda_p^j X_p^j + \sum_{p=1}^{k} \sum_{j=1}^{\ell} C_p \mu_p^j d_p^j$$

$$\text{en (b)} \quad \sum_{p=0}^{K} A_p X_p = b_o$$

$$\Rightarrow A_o X_o + \sum_{p=1}^{K} A_p X_p = b_o$$

$$\Rightarrow A_o X_o + \sum_{p=1}^{K} A_p \left(\sum_{j=1}^{t} \lambda_p^j X_p^j + \sum_{j=1}^{\ell} \mu_p^j d_p^j \right)$$

$$\Rightarrow A_o X_o + \sum_{p=1}^{k} \sum_{j=1}^{t} A_p \lambda_p^j X_p^j + \sum_{p=1}^{K} \sum_{j=1}^{\ell} A_p \mu_p^j d_p^j$$

Como ya se considero a los subproblemas p p=1,2,..,k en la expresión de Xp, entonces de (a) y (b) sustituyendo obtenemos:

$$\text{Min } Z = C_0 X_0 + \sum_{p=1}^{k} \sum_{j=1}^{t} C_p \lambda_p^j X_p^j + \sum_{p=1}^{k} \sum_{j=1}^{l} C_p \mu_p^j d_p^j$$

s.a.

$$A_0 X_0 + \sum_{p=1}^{k} \sum_{j=1}^{t} A_p \lambda_p^j X_p^j + \sum_{p=1}^{k} \sum_{j=1}^{l} A_p \mu_p^j d_p^j$$

$$\sum_{j=1}^{t} \lambda_p^j = 1 \qquad p=1,2,..,k$$

definiendo las transformaciones:

$$P_p^j = A_p X_p^j \;,\; f_p^j = C_p X_p^j \;,\; \bar{P}_p^j = A_p d_p^j \;,\; \bar{f}_p^j = C_p d_p^j$$

p=1,2,..,k

y sustituyendo se tiene:

encontar $\lambda_p^j >= 0$, $\mu_p^j >= 0$.

$$\text{Min } Z = C_0 X_0 + \sum_{p=1}^{k} \sum_{j=1}^{t} f_p^j \lambda_p^j + \sum_{p=1}^{k} \sum_{j=1}^{l} \bar{f}_p^j \mu_p^j$$

s.a.

$$A_0 X_0 + \sum_{p=1}^{k} \sum_{j=1}^{t} P_p^j \lambda_p^j + \sum_{p=1}^{k} \sum_{j=1}^{l} \bar{P}_p^j \mu_p^j$$

$$\sum_{j=1}^{t} \lambda_p^j = 1 \qquad p=1,2,..,k$$

El cual es el "Problema maestro" o "Problema extremo" para un problema no acotado. Se debe de notar que en el caso de una región acotada se metía como columna extrema a la base del problema maestro a $\begin{pmatrix} P_p^j \\ e_p \end{pmatrix}$, pero en el caso de una región no acotada se mete como columna extrema a $\begin{pmatrix} \bar{P}_p^j \\ 0 \end{pmatrix}$.

Dando los λ_p^j y μ_p^j óptimos para el Problema maestro, entonces la correspondiente solución óptima para el problema original de cada Xp está dada por:

$$X_p = \sum \lambda_p^j X_p^j + \sum \mu_p^j d_p^j$$

CAPITULO II

INTERPRETACION ECONOMICA DEL ALGORITMO DE DESCOMPOSICION

Considerando la estructura que deben de tener los subproblemas de programación lineal, para poder aplicar el método de Descomposición se puede hacer la siguiente interpretación económica.

Sea la estructura siguiente:

función objetivo para todos los subsistemas

control maestro de todos los subsistemas	recursos que utilizan todos los subsistemas simultáneamente.
subsistema 1	recursos disponibles únicamente para el subsistema 1.
subsistema 2	recursos disponibles únicamente para el subsistema 2.
. . .	. . . . .
subsistema p	recursos disponibles únicamente para el subsistema p.

Se consideran ahora los efectos causados por cualquier acción del subsistema p (p=1,2,..,k) en la función objetivo de todo el sistema. Si el subsistema p decide en el valor de los niveles de las actividades asociadas con el vector Xp, se genera un costo

CpXp en la función objetivo del sistema y se consumen recursos ApXp que otros subsistemas no podran usar.

Sin embargo, los efectos resultantes de las acciones asociadas con la decisión del subsistema p en todo el sistema, queda reflejada en el vector de los precios duales π. Si las decisiones del subsistema p causan efectos negativos en la función objetivo de todo el sistema, el vector dual actuará como un mecanismo de corrección y no permitira que el nivel de decisión Xp sea muy elevado, ni que se consuman muchos recursos ApXp. Por el otro lado, si los efectos son beneficos en la función objetivo, el vector de precios duales π, hara que el nivel de Xp sea lo suficientemente elevado y que ademas se consuman los recursos necesarios ApXp a ese nivel.

Este tipo de coordinación es la que el Algoritmo de Descomposición logra al optimizar los p subproblemas de la forma

$$\text{Opt } (Cp - \pi \, Ap)Xp$$

s.a.

$$BpXp = bp \qquad p=1,2,..,k$$

$$Xp >= 0$$

Las variables λ_i^j que se van determinando en el proceso iterativo, son los pesos ponderados de las diferentes decisiones j (j=1,2,..,Np) asociadas a un subsistema i (i=1,2,..,p). El criterio de òptimalidad:

$$(Cp - \pi \, Ap)Xp - \tilde{\pi} \, p <= 0$$

$$p=1,2,..,k$$

indica que en el punto óptimo los efectos negativos resultantes de la decisión del subsistema p (p=1,2,..,k) dados por $C_pX_p - \pi A_p X_p$, deben ser menor o igual a los efectos negativos resultantes de la decisión integrada de todo el sistema dada por π. En efecto, en el punto óptimo se tiene:

$$(C_p - \pi A_p)X_p - \tilde{\pi}_p <= 0 \qquad p=1,2,..,k$$

o

$$\tilde{\pi}_p >= (C_p - \pi A_p)X_p \qquad p=1,2,..,k$$

En resumen se puede decir que las decisiones individuales de cada subsistema p (p=1,2,..,k) dadas por Xp son trasmitidas al control maestro, el cual genera pesos ponderados λ_i^j asociadas con cada decisión, y ademas mecanismos correctores $\pi = (\pi , \tilde{\pi})$ que influirán en las nuevas decisiones que hacen individualmente los subsistemas.

CAPITULO III

ETAPAS EN EL ALGORITMO DE DESCOMPOSICION

1) Si en el problema de programación lineal se tienen restricciones de = y/o >= con términos independientes > 0 , entonces determinar el coeficiente de penalización. El cual va a ser el número mas grande de la función objetivo multiplicado por 10000. Donde el coeficiente de penalización se sumará si se está minimizando y se restará si se está máximizando, a las variables artificiales que asi lo requieran.

2) Poner el problema de programación lineal en forma estandar, agregando las variables de holgura.

3) Obtener información del problema de programación lineal en forma estandar de las siguientes matrices;

Ap : p=0,1,2,..,k Ap son los coeficientes de las restricciones generales, limitadas por cada subproblema p.

Bp : p=1,2,..,k Bp son los coeficientes del subproblema p

bp : p=0,1,2,..,k b0 = términos independientes de las restricciones generales

bp = términos indeps del subproblema p.

Cp : p=0,1,2,..,k Cp son los coeficientes de costo de cada subproblema p.

4) Determinar una base inicial factible para el problema maestro, agregando variables artificiales si fuera necesario. Si son agregadas variables artificiales, seran penalizadas en la función objetivo por el coeficiente calculado en 1).

5) Calcular la inversa de la base obtenida denotandola por B^{-1}

6) Denotar por V a las variables basicas asociadas a B.

7) Denotar por C_B a los coeficientes de costo de las variables basicas.

8) Denotar por X_B a los tèrminos independientes del problema maestro.

9) Calcular el vector de precios π ;

$$\text{donde } \pi = C_B B^{-1}$$

10) Separar el vector de precios π en;

$\hat{\pi}$ = vector que contendra tantos elementos como número de restricciones generales del P.P.L.

$\hat{\hat{\pi}}$ = vector que contendra tantos elementos como número de subproblemas (k).

$$\pi = (\hat{\pi} , \hat{\hat{\pi}})$$

11) Calcular las nuevas funciones objetivo.

$$Zp = (Cp - \hat{\pi} Ap)Xp \qquad p=1,2,...,k$$

12) Calcular los puntos extremos solución de los siguientes problemas de programación lineal:

$$\text{Min (max)} \quad Zp = (Cp - \hat{\pi} Ap)Xp$$

s.a.

$$BpXp = b_p \qquad p=1,2,...,k$$

$$X_p \geq 0$$

13) Evaluar;

$$\min[\min_{p}(Z_p - \tilde{\hat{\pi}}\, p), \min(C_o^j - \pi A_o)] \quad .. \quad (i) \qquad p=1,2,\ldots,k$$

o

$$\max[\max_{p}(Z_p - \tilde{\hat{\pi}}\, p), \max(C_o - \tilde{\pi} A_o)] \quad .. \quad (ii) \qquad p=1,2,\ldots,k$$

si se está minimizando o máximizando se evaluara (i) o (ii) según corresponda. Si (i) >= 0 o (ii) <= 0 entonces se alcanzo la condición de óptimalidad para el problema maestro y la solución para el problema de programación lineal original está dada por:

$$X_p = \sum \lambda_p^j X_p^j + \sum \mu_p^j d_p^j \qquad p=1,2,\ldots,k$$

el valor de Zmin o Zmax se calculará como; $C_B X_B$.

Fin del Algoritmo de Descomposición

14) Si no se cumple 13); evaluar (i) o (ii) segun corresponda y determinar que variable es propuesta a entrar a la base del problema maestro, considerando que;

- si el subproblema es acotado "crear" la variable λ_p^j
- si el subproblema no es acotado "crear" la variable μ_p^j
- si se selecciono a la variables Xo , la variable será X_o^j donde j es un número que distingue una variable de otra.

15) Una vez determinado el subproblema al que corresponde el min(max) o seleccionada la X_o^j correspondiente evaluar;

ρ_p^j y f_p^j donde p = # de subproblema seleccionado en 14)

considerando que si se selecciono X_o^j entonces;

$P_o^j = A_o^j$ donde A_o^j indica la columna j de la matriz Ao

y

$f_\rho^j = 0$

16) La columna propuesta a ser introducida a la base del problema maestro, debe de ser modificada de la siguiente forma;

- si el problema es acotado se transformará por:

$$B^{-1}\begin{pmatrix} P_p^j \\ \mathcal{C}_p \end{pmatrix} \text{ donde } \mathcal{C}_p = \text{p-esima columna de una matriz unitaria de pxp.}$$

- si el problema es no acotado o si se selecciono X_o^j, se transformará por;

$$B^{-1}\begin{pmatrix} P_p^j \\ 0 \end{pmatrix}$$

17) Determinar que variable sale del problema maestro, calculando;

r = min(max) [del vector X entre la columna transformada de 16)] con lo cual se conoce en que renglón se encuentra el min(max).

18) Generar la nueva base del problema maestro que sera;

- para un problema acotado asociarle $\begin{pmatrix} P_p^j \\ \mathcal{C}_p \end{pmatrix}$ en la columna r.
- para un problema no acotado o si se selecciono la variable X_o^j, asociarle $\begin{pmatrix} P_p^j \\ 0 \end{pmatrix}$ en la columna r.

19) Calcular la inversa de la nueva base generada del problema maestro.

20) Actualizar el vector V en la columna r con la variable seleccionada.

21) Actualizar el vector C_B en la columna r con el coeficiente de costo (f_ρ^j) asociado a la variable seleccionada.

22) Determinar el nuevo vector de términos independientes como:

$X_B = B^{-1} X_B^*$ donde X_B^* es el último vector de términos independientes calculado.

23) Ir a 9).

CAPITULO IV

EJEMPLO PRACTICO DEL ALGORITMO DE DESCOMPOSICION

Ejemplo de un problema de programacion lineal, resuelto utilizando el Algoritmo de Descomposicion.

$$\text{Min } Z = -2X_1 + 5X_2 - 5X_3 + 10X_4$$
$$\text{s.a}$$
$$\begin{aligned}
5X_1 - 2X_2 - 12X_3 + 7X_4 &<= 25 \\
X_1 + X_2 + X_3 + X_4 &>= 0.001 \\
X_1 + 2X_2 &>= 4 \\
3X_1 + X_2 &>= 6 \\
2X_3 + 5X_4 &<= 10 \\
X_3 + 0.2X_4 &<= 4 \\
X_1,\ X_2,\ X_3, X_4 &>= 0
\end{aligned}$$

Agregando las variables de holgura correspondientes;

$$\text{Min } Z = -2X_1 + 5X_2 - 5X_3 + 10X_4$$
$$\text{s.a}$$
$$\begin{aligned}
X_0^1 + 5X_1 - 2X_2 - 12X_3 + 7X_4 &= 25 \\
-X_0^2 + X_1 + X_2 + X_3 + X_4 &= 0.001 \\
X_1 + 2X_2 - X_7 &= 4 \\
3X_1 + X_2 - X_8 &= 6 \\
2X_3 + 5X_4 + X_9 &= 10 \\
X_3 + 0.2X_4 + X_{10} &= 4 \\
X_0^1, X_0^2, X_1, X_2, X_3, X_4, X_7, X_8, X_9, X_{10} &\geq 0
\end{aligned}$$

de donde se tiene;

$$A_0 = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \quad A_1 = \begin{pmatrix} 5 & -2 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \quad A_2 = \begin{pmatrix} -12 & 7 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$

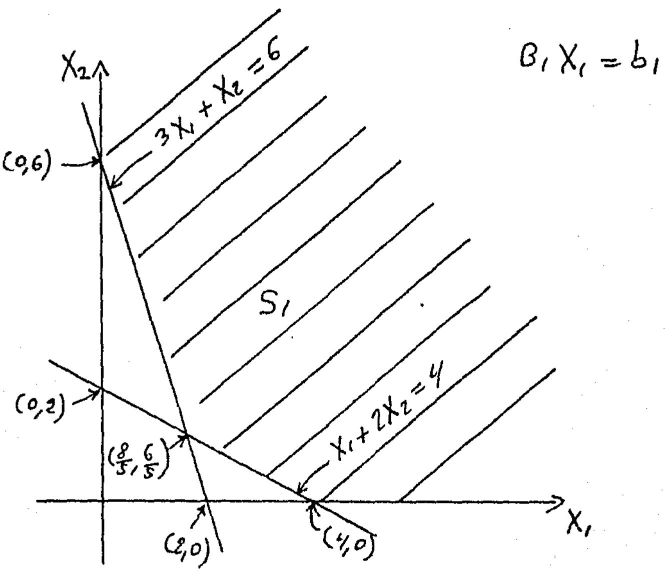
$$B_1 = \begin{pmatrix} 1 & 2 & -1 & 0 \\ 3 & 1 & 0 & -1 \end{pmatrix} \quad B_2 = \begin{pmatrix} 2 & 5 & 1 & 0 \\ 1 & 0.2 & 0 & 1 \end{pmatrix}$$

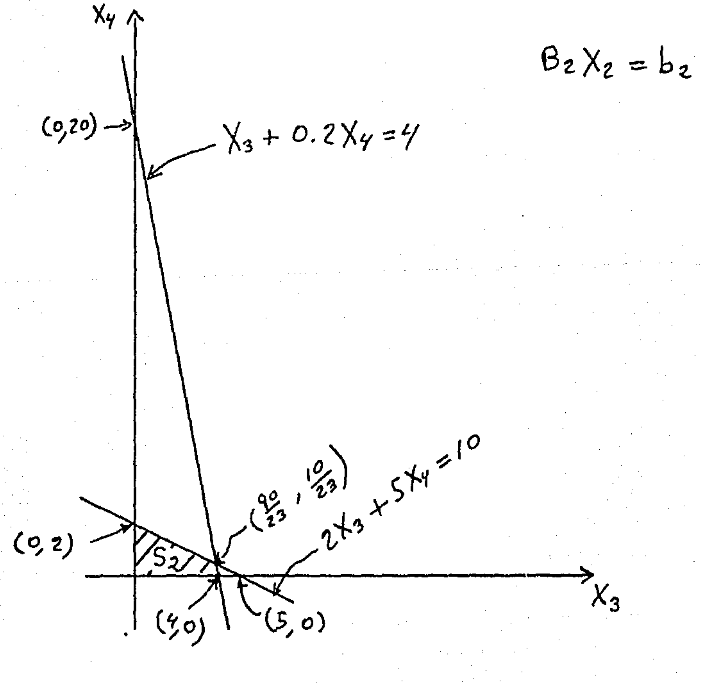
$$b_0 = (25, 0.001)^t \ ; \ b_1 = (4,6)^t \ ; \ b_2 = (10,4)^t$$

$$C_0 = (0,0) \ ; \ C_1 = (-2,5,0,0) \ ; \ C_2 = (-5,10,0,0)$$

$$X_0 = (X_0^1, X_0^2) \ ; \ X_1 = (X_1, X_2, X_7, X_8) \ ; \ X_2 = (X_3, X_4, X_9, X_{10})$$

Dibujando los espacios solucion para los subproblemas $B_1X_1 = b_1$ y $B_2X_2 = b_2$


$B_1 X_1 = b_1$
X_2
$(0,6)$
$3X_1 + X_2 = 6$
S_1
$(0,2)$
$(\frac{8}{5}, \frac{6}{5})$
$X_1 + 2X_2 = 4$
$(2,0)$
$(4,0)$
X_1


X_4
$B_2 X_2 = b_2$
$(0,20)$
$X_3 + 0.2X_4 = 4$
$(\frac{90}{23}, \frac{10}{23})$
$2X_3 + 5X_4 = 10$
$(0,2)$
S_2
$(4,0)$
$(5,0)$
X_3

Calculando el problema maestro;

$$\text{Min } Z = C_0 X_0 + \sum_{p=1}^{k} \sum_{j=1}^{N_p} f_p^j \lambda_p^j$$

$$\text{s.a} \quad A_0 X_0 + \sum_{p=1}^{k} \sum_{j=1}^{N_p} P_p^j \lambda_p^j = b_0$$

$$\sum_{j=1}^{N_p} \lambda_p^j = 1 \qquad p=1,2,\ldots,k$$

Se añaden las variables artificiales a, a_1, a_2 al problema maestro para tener una base inicial factible.

		λ_1^1	λ_1^2		λ_1^5	λ_2^1	λ_2^2		λ_2^4	a	a_1	a_2	
0	0	$C_1X_1^1$	$C_1X_1^2$	..	$C_1X_1^5$	$C_2X_2^1$	$C_2X_2^2$	..	$C_2X_2^4$	M	M	M	0
X_0^1	0	$A_1X_1^1$	$A_1X_1^2$	..	$A_1X_1^5$	$A_2X_2^1$	$A_2X_2^2$	..	$A_2X_2^4$	0	0	0	25
	$-X_0^2$									1	0	0	0.001
0	0	1	1	..	1	0	0	..	0	0	1	0	1
0	0	0	0	..	0	1	1	..	1	0	0	1	1

La M es el coeficiente de penalización de las variables artificiales; sea M = 100000.

Con la anterior tabla que contiene al problema maestro se observa que la base factible inicial es;

$$B = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad B^{-1} = I$$

y las variables asociadas a B son ; $V = (X_0^1, a, a_1, a_2)$.

Sea X_B los terminos independientes del problema maestro y C_B los coeficientes de costo de las variables basicas;

$$X_B = (25, 0.001, 1, 1) \quad ; \quad C_B = (0, 1E5, 1E5, 1E5)$$

Calculando $\Pi = C_B B^{-1}$

$$= (0, 1E5, 1E5, 1E5)$$

y como $\Pi = (\hat{\pi}, \hat{\hat{\pi}})$

$\therefore\ \hat{\pi} = (0,1E5)\ ;\ \tilde{\hat{\pi}} = (1E5,1E5)$

Para el vector de costos π, necesitamos ver si cualquier otro punto extremo de los subproblemas (B_1X_1, B_2X_2) pueden incrementar el valor de la función objetivo. Por lo que se calculará las nuevas funciones objetivo a optimizar para los subproblemas;

$$\text{i.e } Z_p = (C_p - \pi A_p)X_p$$

$Z_1 = (C_1 - \pi A_1)X_1 = (-100002,-99995,0,0)X_1$

$Z_2 = (C_2 - \pi A_2)X_2 = (-100005,-99990,0,0)X_2$

Entonces tenemos que encontrar los puntos extremos solucion de los siguientes problemas de programación lineal;

$$\text{Min } Z_1 = -100002X_1 - 99995X_2 + 0X_7 + 0X_8$$

s.a

$$B_1X_1 = b_1$$

$$X_1 \geq 0$$

y

$$\text{Min } Z_2 = -100005X_3 - 99990X_4 + 0X_9 + 0X_{10}$$

s.a

$$B_2X_2 = b_2$$

$$X_2 \geq 0$$

la solucion para $Z_1 = -10,599,470$ con el siguiente rayo extremo;

$d_1' = (0,6,8,0) + \alpha(0,1,2,1)$ sea $\alpha = 100$

$d_1' = (0,106,208,100)$

la solucion para $Z_2 = -434,797.826$ con el vector solucion;

$X_2' = (3.91,0.434,0,0)$

como $\hat{\pi} = (1E5,1E5)$

$$Z_1 - \tilde{\hat{\pi}}_1 = -10,599,470 - 1E5 = -10,699,470$$

$$Z_2 - \tilde{\hat{\pi}}_2 = -434,797.826 - 1E5 = -534,797.826$$

determinando si entra la variable X_0^2, calculando su coeficiente de costo asociado;

$$C_0^2 - Z_0^2 = 0 - C_B B^{-1}(0,-1,0,0)$$
$$= 1E5$$

el minimo corresponde al subproblema 1

y seleccionamos μ_1^1 para entrar a la base del problema maestro.

$$P_1^1 = A_1 d_1^1 = (-212, 106)$$
$$f_1^1 = C_1 d_1^1 = 530$$

Como se está usando el metodo simplex revisado el vector a ser introducido debe de ser transformado por B^{-1}

$$\therefore \quad B^{-1}(P_1^1 0)^t = (-212, 106, 0, 0)^t$$

determinando la variable que sale de la base B, se tiene;

$$\text{Min } [\, X_B/(B^{-1}(P_1^1 0)^t \,]$$

el minimo esta en el segundo renglon, y sale la segunda variable;

a

la nueva base es;

$$B = \begin{pmatrix} 1 & -212 & 0 & 0 \\ 0 & 106 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad ; \quad B^{-1} = \begin{pmatrix} 1 & 2 & 0 & 0 \\ 0 & .0094 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

El nuevo vector de variables basicas es; $V=(X_0^1, \mu_1^1, a_1, a_2)$
con $C_B = (0, 530, 1E5, 1E5)$

El nuevo vector de coeficientes es;

$$X_B = B^{-1} X_B = (25.002, 9.4339E-6, 1, 1)^t$$

El vector de precios es;

$$\pi = C_B B^{-1} = (0, 5, 1E5, 1E5)$$
$$\pi = (0,5) \quad ; \quad \hat{\pi} = (1E5, 1E5)$$

Utilizando el nuevo vector de precios $\pi = (0,5,1E5,1E5)$, repetimos el proceso para ver si un punto extremo de los subproblemas puede incrementar la solucion.

Calculando las nuevas funciones objetivo;

$$\text{i.e } Z_p = (C_p - \pi A_p) X_p$$

$Z_1 = (C_1 - \pi A_1) X_1 = (-7, 0, 0, 0) X_1$
$Z_2 = (C_2 - \pi A_2) X_2 = (-10, 5, 0, 0) X_2$

los problemas a minimizar son;

$$\text{Min } Z_1 = -7X_1 + 0X_2 + 0X_7 + 0X_8$$
s.a
$$B_1 X_1 = b_1$$
$$X_1 \geq 0$$

y

$$\text{Min } Z_2 = -10X_3 - 5X_4 + 0X_9 + 0X_{10}$$
s.a

$$B_2X_2 = b_2 \quad X_2 \geq 0$$

la sola solucion para $Z_1 = -728$ con el siguiente rayo extremo;

$d_1^2 = (4,0,0,6) + \alpha(1,0,1,3)$ sea $\alpha = 100$

$d_1^2 = (104,0,100,306)$

la solucion para $Z_2 = -40$ con el vector solucion;

$X_2^2 = (4,0,2,0)$

como $\hat{\pi} = (1E5,1E5)$

$$Z_1 - \hat{\pi}_1 = -100728$$
$$Z_2 - \hat{\pi}_2 = -100040$$

evaluando X_0^2

$$C_0^2 - Z_0^2 = 0 - C_B B^{-1}(0,-1,0,0) = 5$$

el minimo corresponde al subproblema 1

y seleccionamos μ_1^2 para entrar a la base del problema maestro.

$$P_1^2 = A_1 d_1^2 = (520,104)^t$$
$$f_1^2 = C_1 d_1^2 = -208$$

modificando la columna a ser introducida

$$B^{-1}(P_1^2 0)^t = (728,0.9811,0,0)^t$$

determinando la variable que sale de la base B, se tiene;

$$\text{Min } [\ X_B / (B^{-1}(P_1^2 0)\ ]$$

el minimo esta en el segundo renglon, y sale la segunda variable; μ_1^1

la nueva base es;

$$B = \begin{pmatrix} 1 & 520 & 0 & 0 \\ 0 & 104 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad ; \quad B^{-1} = \begin{pmatrix} 1 & -5 & 0 & 0 \\ 0 & 9.615E-3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

El nuevo vector de variables basicas es; $V = (X_0^1, \mu_1^2, a_1, a_2)$ con $C_B = (0,-208,1E5,1E5)$

El nuevo vector de coeficientes es;

$$X_B = B^{-1} X_B = (24.995, 9.6153E-6, 1, 1)^t$$

El vector de precios es;

$$\pi = C_B B^{-1} = (0,-2,1E5,1E5)$$
$$\hat{\pi} = (0,-2) \quad ; \quad \hat{\hat{\pi}} = (1E5,1E5)$$

Calculando las nuevas funciones objetivo

$Z_1 = (C_1 - \pi A_1)X_1 = (0,7,0,0)X_1$

$Z_2 = (C_2 - \pi A_2)X_2 = (-3,12,0,0)X_2$

Entonces tenemos que encontrar los puntos extremos solucion de los siguientes problemas de programacion lineal;

$$\text{Min } Z_1 = 0X_1 + 7X_2 + 0X_7 + 0X_8$$
s.a
$$B_1X_1 = b_1 \quad X_1 \geq 0$$

y

$$\text{Min } Z_2 = -3X_3 + 12X_4 + 0X_9 + 0X_{10}$$
s.a
$$B_2X_2 = b_2 \quad X_2 \geq 0$$

la solucion para $Z_1 = 42$ con el vector solucion

$X_1^2 = (0,6,8,0)$

la solucion para $Z_2 = -12$ con el vector solucion;

$X_2^3 = (4,0,2,0)$

como $\hat{\pi} = (1E5,1E5)$

$$Z_1 - \hat{\pi}_1 = -99958$$
$$Z_2 - \hat{\pi}_2 = -100012$$

evaluandondo X_0^2

$$C_0^2 - Z_0^2 = 0 - C_B B^{-1}(0,-1,0,0)$$
$$= -2$$

el minimo corresponde al subproblema 2
y seleccionamos λ_2^3 para entrar a la base del problema maestro.

$$P_2^3 = A_2 X_2^3 = (-48,4)^t$$
$$f_2^3 = C_2 X_2^3 = -20$$

modificando la columna a ser introducida;

$$B^{-1}(P_2^3 \, e_2)^t = (-68,3.846E-2,0,1)^t$$

determinando la variable que sale de la base B, se tiene;

$$\text{Min } [\, X_B / (B^{-1}(P_2^3 \, e_2) \,]$$

el minimo esta en el segundo renglon, y sale la variable; μ_1^2
la nueva base es;

$$B = \begin{pmatrix} 1 & -48 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad ; \quad B^{-1} = \begin{pmatrix} 1 & 12 & 0 & 0 \\ 0 & .25 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -.25 & 0 & 1 \end{pmatrix}$$

El nuevo vector de variables basicas es; $V=(X_0^1, \lambda_2^3, a_1, a_2)$
con $C_B = (0,-20,1E5,1E5)$

El nuevo vector de coeficientes es;

$$X_B = B^{-1} X_B = (25.012,2.5E-4,1,0.99975)^t$$

El vector de precios es;

$$\pi = C_B B^{-1} = (0,-25005,1E5,1E5)$$
$$\hat{\pi} = (0,-25005) \quad ; \quad \hat{\hat{\pi}} = (1E5,1E5)$$

Calculando las nuevas funciones objetivo;

$$Z_1 = (C_1 - \hat{\pi} A_1) X_1 = (25003,25010,0,0)$$
$$Z_2 = (C_2 - \hat{\pi} A_2) X_2 = (25000,25015,0,0)$$

los problemas a minimizar son;

$$\text{Min } Z_1 = 25003X_1 + 25010X_2 + 0X_7 + 0X_8$$
s.a
$$B_1 X_1 = b_1 \qquad X_1 \geq 0$$

y

$$\text{Min } Z_2 = 25000X_3 + 25015X_4 + 0X_9 + 0X_{10}$$
s.a
$$B_2 X_2 = b_2 \qquad X_2 \geq 0$$

la sola solución para $Z_1 = 70016.8$ con el vector solución;
$X_1^4 = (1.6,1.2,0,0)$
la solución para $Z_2 = 0$ con el vector solucion;
$X_2^4 = (0,0,10,4)$
como $\tilde{\pi} = (1E5,1E5)$

$$Z_1 - \tilde{\pi}_1 = -29983.2$$
$$Z_2 - \tilde{\pi}_2 = -100000$$

evaluando X_0^2

$$C_0^2 - Z_0^2 = 0 - C_B B^{-1}(0,-1,0,0)$$
$$= -25005$$

el minimo corresponde al subproblema 2
y seleccionamos λ_2^4 para entrar a la base del problema maestro.

$$P_2^4 = A_2 d_2^4 = (0,0)$$
$$f_2^4 = C_2 d_2^4 = 0$$

modificando la columna a ser introducida

$$B^{-1}(P_2^4 \, e_2) = (0,0,0,1)$$

determinando la variable que sale de la base B, se tiene;

$$\text{Min } [\, X_B / (B^{-1}(P_2^4 \, e_2) \,]$$

el minimo esta en el cuarto renglon, y sale la variable; λ_2
la nueva base es;

$$B = \begin{pmatrix} 1 & -48 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix} \quad ; \quad B^{-1} = \begin{pmatrix} 1 & 12 & 0 & 0 \\ 0 & .25 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -.25 & 0 & 1 \end{pmatrix}$$

El nuevo vector de variables basicas es; $V=(X_0^1, \lambda_2^3, a_1, \lambda_2^4)$
con $C_B = (0,-20,1E5,0)$

El nuevo vector de coeficientes es;

$$X_B = B^{-1}X_B = (25.012, 2.5E-4, 1, 0.99975)^t$$

El vector de precios es;

$$\pi = C_B B^{-1} = (0,-5,1E5,0)$$

$$\pi = (0,-5) \quad ; \hat{\pi} = (1E5,0)$$

Calculando las nuevas funciones objetivo;

$$\text{i.e } Z_p = (C_p - \pi A_p) X_p$$

$Z_1 = (C_1 - \pi A_1) X_1 = (3,10,0,0) X_1$
$Z_2 = (C_2 - \pi A_2) X_2 = (0,15,0,0) X_2$
los problemas a minimizar son;

$$\text{Min } Z_1 = 3X_1 + 10X_2 + 0X_7 + 0X_8$$
s.a
$$B_1 X_1 = b_1 \quad X_1 \geq 0$$

y

$$\text{Min } Z_2 = 0X_3 + 15X_4 + 0X_9 + 0X_{10}$$
s.a
$$B_2 X_2 = b_2 \quad X_2 \geq 0$$

la sola solucion para $Z_1 = 12$ con el vector solucion;

$X_1^5 = (4,0,0,6)$

la solucion para $Z_2 = 0$ con el vector solucion;

$X_2^5 = (0,0,10,4)$

como $\hat{\pi} = (1E5,0)$

$$Z_1 - \hat{\pi}_1 = -99988$$
$$Z_2 - \hat{\pi}_2 = 0$$

evaluando X_0^2

$$C_0^2 - Z_0^2 = 0 - C_B B^{-1}(0,-1,0,0)$$
$$= -5$$

el minimo corresponde al subproblema 1
y seleccionamos λ_1^5 para entrar a la base del problema maestro.

$$P_1^5 = A_1 d_1^5 = (20,4)^t$$
$$f_1^5 = C_1 d_1^5 = -8$$

modificando la columna a ser introducida

$$B^{-1}(P_1^5 Q_1) = (68,1,1,-1)$$

determinando la variable que sale de la base B, se tiene;

$$\text{Min } [\ X_B / (B^{-1}(P_1^5 Q_1)\ ]$$

el minimo esta en el segundo renglon, y sale la variable; λ_2^3
la nueva base es;

$$B = \begin{pmatrix} 1 & 20 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad ; \quad B^{-1} = \begin{pmatrix} 1 & -5 & 0 & 0 \\ 0 & .25 & 0 & 0 \\ 0 & -.25 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

El nuevo vector de variables basicas es; $V=(X_0^1, \lambda_1^5, a_1, \lambda_2^4)$
con $C_B = (0,-8,1E5,0)$

El nuevo vector de coeficientes es;

$$X_B = B^{-1}X_B = (24.995, 2.5E-4, 1, 1)^t$$

El vector de precios es;

$$\pi = C_B B^{-1} = (0,-25002,1E5,0)$$

$$\pi = (0,-25002) \quad ; \quad \hat{\pi} = (1E5,0)$$

Calculando las nuevas funciones objetivo

$Z_1 = (C_1-\pi A_1)X_1 = (25000,25007,0,0)$

$Z_2 = (C_2-\pi A_2)X_2 = (24997,25012,0,0)$

los problemas a minimizar son;

$$\text{Min } Z_1 = 25000X_1 + 25007X_2 + 0X_7 + 0X_8$$
$$\text{s.a}$$
$$B_1 X_1 = b_1 \qquad X_1 \geq 0$$

y

$$\text{Min } Z_2 = 24997X_3 + 25012X_4 + 0X_9 + 0X_{10}$$
$$\text{s.a}$$
$$B_2 X_2 = b_2 \qquad X_2 \geq 0$$

la solucion para $Z_1 = 70008.4$ con el vector solucion

$X_1^6 = (1.6,1.2,0,0)$

la solucion para $Z_2 = 0$ con el vector solucion;

$X_2^6 = (0,0,10,4)$

como $\hat{\pi} = (1E5,0)$

$$Z_1 - \hat{\pi}_1 = -29991.6$$

$$Z_2 - \hat{\pi}_2 = 0$$

evaluandondo X_0^2

$$C_0^2 - Z_0^2 = 0 - C_B B^{-1}(0,-1,0,0)^t$$
$$= -25002$$

el minimo corresponde al subproblema 1

y seleccionamos λ_1^6 para entrar a la base del problema maestro.

$$P_1^6 = A_1 X_1^6 = (5.6,2.8)^t$$

$$f_1^6 = C_1 X_1^6 = 2.8$$

modificando la columna a ser introducida;

$$B^{-1}(P_1^6 e_1) = (-8.4,.7,.3,0)^t$$

determinando la variable que sale de la base B, se tiene;

$$\text{Min } [\, X_B/(B^{-1}(P_1^6 e_1)\,]$$

el minimo esta en el segundo renglon, y sale la variable;

la nueva base es;

$$B = \begin{pmatrix} 1 & 5.6 & 0 & 0 \\ 0 & 2.8 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad ; \quad B^{-1} = \begin{pmatrix} 1 & -2 & 0 & 0 \\ 0 & .35714 & 0 & 0 \\ 0 & -.35714 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

El nuevo vector de variables basicas es; $V=(X_0^1, \lambda_1^6, a_1, \lambda_2^4)$

con $C_B = (0,2.8,1E5,0)$

El nuevo vector de coeficientes es;

$$X_B = B^{-1}X_B = (24.998,3.571428E-4,0.9996,1)^t$$

El vector de precios es;

$$\pi = C_B B^{-1} = (0,-35713.2857,1E5,0)$$

$$\pi = (0,-35713.2857) \quad ; \quad \hat{\pi} = (1E5,0)$$

Calculando las nuevas funciones objetivo

$Z_1 = (C_1-\pi A_2)X_1 = (35711.2,35718.2,0,0)X_1$

$Z_2 = (C_2-\pi A_2)X_2 = (35708.2,35723.2,0,0)X_2$

los problemas a minimizar son;

$$\text{Min } Z_1 = 35711.2X_1 + 35718.2x_2 + 0X_7 + 0X_8$$
s.a
$$B_1X_1 = b_1 \qquad X_1 \geq 0$$
y
$$\text{Min } Z_2 = 35708.2X_3 - 35723.2X_4 + 0X_9 + 0X_{10}$$
s.a
$$B_2X_2 = b_2 \qquad X_2 \geq 0$$
la solución para $Z_1 = 99999.9$ con el vector solucion
$X_1^7 = (1.6, 1.2, 0, 0)$
la solución para $Z_2 = 0$ con el vector solucion;
$X_2^7 = (0, 0, 10, 4)$
como $\widetilde{\pi} = (1E5, 1E5)$
$$Z_1 - \widetilde{\pi}_1 = -100728$$
$$Z_2 - \widehat{\pi}_2 = -100040$$
evaluando X_0^2
$$C_0^2 - Z_0^2 = 0 - C_B B^{-1}(0,-1,0,0)^t$$
$$= -35713{,}28$$
el minimo corresponde a la variable X_0^2
y entra a la base del problema maestro.
$$P_0^2 = (0 \; -1)^t$$
$$f_0^2 = 0$$
modificando la columna a ser introducida
$$B^{-1}(P_0^2 0) = (2, -.35714, .35714, 0)^t$$
determinando la variable que sale de la base B, se tiene;
$$\text{Min } [\, X_B / (B^{-1}(P_0^2 0)) \,]$$
el minimo esta en el tercer renglon, y sale la variable; a_1
la nueva base es;
$$B = \begin{pmatrix} 1 & 5.6 & 0 & 0 \\ 0 & 2.8 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad ; \quad B^{-1} = \begin{pmatrix} 1 & 0 & -5.6 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 2.8 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$
El nuevo vector de variables basicas es; $V = (X_0^1, \lambda_1^6, X_0^2, \lambda_2^4)$
con $C_B = (0, 2.8, 0, 0)$
El nuevo vector de coeficientes es;
$$X_B = B^{-1}X_B = (19.4, 1, 2.799, 1)^t$$
El vector de precios es;
$$\pi = C_B B^{-1} = (0, 0, 2.8, 0)$$
$$\widehat{\pi} = (0,0) \quad ; \quad \widetilde{\pi} = (2.8, 0)$$
Calculando las nuevas funciones objetivo
$Z_1 = (C_1 - \widehat{\pi}A_1)X_1 = (-2, 5, 0, 0)X_1$
$Z_2 = (C_2 - \widehat{\pi}A_2)X_2 = (-5, 10, 0, 0)X_2$
los problemas a minimzar son;
$$\text{Min } Z_1 = -2X_1 + 5X_2 + 0X_7 + 0X_8$$
s.a
$$B_1X_1 = b_1 \qquad X_1 \geq 0$$
y
$$\text{Min } Z_2 = -5X_3 + 10X_4 + 0X_9 + 0X_{10}$$
s.a
$$B_2X_2 = b_2 \qquad X_2 \geq 0$$
la solucion para $Z_1 = -208$ con el siguiente rayo extremo;
$d_1^8 = (4, 0, 0, 6) + \alpha(1, 0, 1, 3)$ sea $\alpha = 100$
$d_1^8 = (104, 0, 100, 306)$
la solucion para $Z_2 = -20$ con el vector solucion;

$X_2^8 = (4,0,2,0)$
como $\widetilde{\pi} = (2.8,0)$

$$Z_1 - \widetilde{\pi}_1 = -210.8$$
$$Z_2 - \widetilde{\pi}_2 = -20$$

evaluandondo X_0^1

$$C_0' - Z_0' = 0 - C_B B^{-1}(1,0,0,0)^t$$
$$= 0$$

el minimo corresponde al subproblema 1
y seleccionamos μ_1^8 para entrar a la base del problema maestro.

$$P_1^8 = A\ X = (520,104)$$
$$f_1^8 = C\ X = -208$$

modificando la columna a ser introducida;

$$B^{-1}(P_1^8 0) = (520,0,-104,0)$$

determinando la variable que sale de la base B, se tiene;

$$\text{Min } [\ X_B/(B^{-1}(P_1^8 0)\ ]$$

el minimo esta en el primer renglon, y sale la variable; X_0^1
la nueva base es;

$$B = \begin{pmatrix} 520 & 5.6 & 0 & 0 \\ 104 & 2.8 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad ; \quad B^{-1} = \begin{pmatrix} 1.92E-3 & 0 & -0.010 & 0 \\ 0 & 0 & 1 & 0 \\ 0.2 & -1 & 1.68 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

El nuevo vector de variables basicas es; $V=(\mu_1^8, \lambda_1^6, X_0^2, \lambda_2^4)$
con $C_B= (-208,2.8,0,0)$
El nuevo vector de coeficientes es;

$$X_B = B^{-1}X_B = (0.0373,1,6.679,1)^t$$

El vector de precios es;

$$\pi = C_B B^{-1} = (-0.4,0,5.04,0)$$
$$\widetilde{\pi} = (-0.4,0) \quad ; \quad \hat{\pi} = (5.04,0)$$

Calculando las nuevas funciones objetivo;

$$Z_1 = (C_1 - \pi A_1)X_1 = (0,4.2,0,0)X_1$$
$$Z_2 = (C_2 - \pi A_2)X_2 = (-9.8,12.8,0,0)X_2$$

los problemas a minimizar son;

$$\text{Min } Z_1 = 0X_1 + 4.2X_2 + 0X_7 + 0X_8$$
s.a
$$B_1 X_1 = b_1 \qquad X_1 \geq 0$$

y

$$\text{Min } Z_2 = -9.8X_3 + 12.8X_4 + 0X_9 + 0X_{10}$$
s.a
$$B_2 X_2 = b_2 \qquad X_2 \geq 0$$

la sola solucion para $Z_1 = 0$ con el vector solucion;
$X_1^9 = (4,0,0,6)$
la solucion para $Z = -39.2$ con el vector solucion;
$X_2^9 = (4,0,2,0)$
como $\hat{\pi} = (5.04,0)$

$$Z_1 - \hat{\pi}_1 = -5.04$$
$$Z_2 - \hat{\pi}_2 = -39.2$$

evaluando X_0'

$$C_0' - Z_0' = 0 - C_B B^{-1}(1,0,0,0)$$
$$= 0.4$$

el minimo corresponde al subproblema 2
y seleccionamos λ_2^9 para entrar a la base del problema maestro.

$$P_2^9 = A_2 X_2^9 = (-48,4)^t$$

$$f_2^9 = C_2 X_2^9 = -20$$

modificando la columna a ser introducida

$$B^{-1}(P_2^9 Q_2) = (-0.092, 0, -13.6, 1)^t$$

determinando la variable que sale de la base B, se tiene;

$$\text{Min } [\, X_B/(B^{-1}(P_2^9 Q_2)\,]$$

el minimo esta en el cuarto renglon, y sale la variable; λ_2^4

la nueva base es;

$$B = \begin{pmatrix} 520 & 5.6 & 0 & -48 \\ 104 & 2.8 & -1 & 4 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \; ; \quad B^{-1} = \begin{pmatrix} 1.92E-3 & 0 & -0.010 & 0.0923 \\ 0 & 0 & 1 & 2.21E-9 \\ 0.2 & -1 & 1.68 & 13.6 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

El nuevo vector de variables basicas es; $V=(\mu_1^8, \lambda_1^6, X_0^2, \lambda_2^9)$

con $C_B= (-208, 2.8, 0, -20)$

El nuevo vector de coeficientes es;

$$X_B= B^{-1}X_B = (0.12961, 1, 20.279, 1)^t$$

El vector de precios es;

$$\pi = C\, B = (-0.4, 0, 5.04, -39.2)$$

$$\pi = (-0.4, 0) \;\; ; \; \hat{\pi} = (5.04, -39.2)$$

Calculando las nuevas funciones objetivo;

$$\text{i.e } Z_\rho= (C_\rho-\pi A_\rho) X_\rho$$

$$Z_1= (C_1-\pi A_1)X_1= (0, 4.2, 0, 0)X_1$$

$$Z_2= (C_2-\pi A_2)X_2= (-9.8, 12.8, 0, 0)X_2$$

los problemas a minimizar son;

$$\text{Min } Z_1= 0X_1+ 4.2X_2+ 0X_7+ 0X_8$$

s.a

$$B_1 X_1 = b_1 \qquad X_1 \geq 0$$

y

$$\text{Min } Z_2= -9.8X_3+ 12.8X_4+ 0X_9+ 0X_{10}$$

s.a

$$B_2 X_2 = b_2 \qquad X_2 \geq 0$$

la solucion para $Z_1= 0$ con el vector solucion;

$$X_1^{10} = (4, 0, 0, 6)$$

la solucion para $Z_2= -39.2$ con el vector solucion;

$$X_2^{10} = (4, 0, 2, 0)$$

como $\hat{\pi} = (5.04, -39.2)$

$$Z_1 - \hat{\pi}_1 = -5.04$$

$$Z_2 - \hat{\pi}_2 = 0$$

evaluando X_0^1

$$C_0^1 - Z_0^1= 0 - C_B B^{-1}(1, 0, 0, 0)$$

$$= 0.4$$

el minimo corresponde al subproblema 1

y seleccionamos λ_1^{10} para entrar a la base del problema maestro.

$$P_1^{10} = A_1 X_1^{10}= (20, 4)$$

$$f_1^{10} = C_1 X_1^{10}= -8$$

modificando la columna a ser introducida

$$B^{-1}(P_1^{10} Q_1) = (0.0276, 1, 1.68, 0)$$

determinando la variable que sale de la base B, se tiene;

$$\text{Min } [\, X_B/(B^{-1}(P_1^{10} Q_1)\,]$$

el minimo esta en el segundo renglon, y sale la variable;

la nueva base es;

$$B = \begin{pmatrix} 520 & 20 & 0 & -48 \\ 104 & 4 & -1 & 4 \\ 0 & 1 & 0 & 0 \end{pmatrix} ; \qquad B^{-1} = \begin{pmatrix} 1.923E-3 & 0 & -0.0384 & 0.0923 \\ 0 & 0 & 1 & 0 \\ 0.2 & -1 & 0 & 13.6 \end{pmatrix}$$

$\begin{pmatrix}0 & 0 & 0 & 1\end{pmatrix}$ $\begin{pmatrix}0 & 0 & 0 & 1\end{pmatrix}$

El nuevo vector de variables basicas es; $V=(\mu_1^8, \lambda_1^{10}, X_0^2, \lambda_2^9)$
con $C_B= (-208,-8,0,-20)$

El nuevo vector de coeficientes es;

$$X_B= B^{-1}X_B = (0.101923077,1,18.599,1)^t$$

El vector de precios es;

$$\pi = C_B B^{-1} = (-0.4,0,0,-39.2)$$

$$\hat{\pi} = (-0.4,0) \quad ; \quad \tilde{\pi} = (0,-39.2)$$

Calculando las nuevas funciones objetivo

$Z_1= (C_1-\hat{\pi}A_1)X_1= (0,4.2,0,0)X_1$

$Z_2= (C_2-\hat{\pi}A_2)X_2= (-9.8,12.8,0,0)X_2$

los problemas a minimizar son;

$$\text{Min } Z_1 = 0X_1+ 4.2X_2+ 0X_7+ 0X_8$$
s.a
$$B_1X_1 = b_1 \quad X_1 \geq 0$$

y

$$\text{Min } Z_2= -9.8X_3+ 12.8X_4+ 0X_9+ 0X_{10}$$
s.a
$$B_2X_2 = b_2 \quad X_2 \geq 0$$

la solucion para $Z_1= 0$ con el vector solucion

$X_1'' = (4,0,0,6)$

la solucion para $Z = -39.2$ con el vector solucion;

$X_2'' = (4,0,2,0)$

como $\hat{\pi} = (0,-39.2)$

$$Z_1 - \tilde{\pi}_1 = 0$$
$$Z_2 - \tilde{\pi}_2 = 0$$

evaluandondo X_0'

$$C_0'- Z_0'= 0 - C_B B^{-1}(1,0,0,0) = 0.4$$

como $\min[\min_j(Z_p - \hat{\pi}_p),\min(C_0^j - \hat{\pi} A_0^j)] >=0$ entonces se alcanzo la condicion de optimalidad y la solución esta dada por;
como el último vector de coeficientes es;

$$X_B = (0.101923077,1,18.599,1)^t$$

y las variables basicas fueron;

$$V = (\mu_1^8, \lambda_1^{10}, X_0^2, \lambda_2^9)$$

la solución es;

$$X_0 = (X_0^1, X_0^2) = (0,18.599)$$

$$X_1 = (X_1, X_2, X_7, X_8) = \lambda_1^{10}X_1^{10} + \mu_1^8 X_1^8 = 1(4,0,0,6) +$$
$$0.101923077(104,0,100,306)$$
$$=(14.6,0,10.19231,37.18846)$$

$$X_2 = (X_3, X_4, X_9, X_{10}) = \lambda_2^9 X_2^9 = 1(4,0,2,0) = (4,0,2,0)$$

el valor de Z es;

$$Zmin = (-208,-8,0,-20)(0.101923077,1,18.599,1)^t = -49.2$$

CAPITULO V

PAQUETE QUE RESUELVE EL ALGORITMO DE DESCOMPOSICION

A continuación se presenta unos programas que en conjunto constituyen un paquete, el cual tiene como finalidad resolver el Algoritmo de Descomposición de Dantzing y Wolfe.

Para poder usar el paquete es deseable que el Problema de Programación Lineal (P.P.L.) a ser resuelto tenga la estructura angular caracteristica de los P.P.L., que son tratados por el Algoritmo de Descomposición.

Los programas están escritos en lenguaje BASIC, debido a que este es el lenguaje mas común para cualquier microcomputadora. Lo cual no quiere decir que los programas no puedan ser escritos en cualquier otro lenguaje de alto nivel. Debido a que el paquete fue hecho expresamente para usarse en microcomputadoras, este tiene sus limitaciones originadas por la capacidad de almacenamiento de la computadora que son;

- Se puede manejar como máximo 10 subproblemas.
- La dimensión de cada subproblema puede ser como máxima de 9 renglones por 9 columnas.

Las limitantes anteriores pueden modificarse dependiendo de la capacidad de la computadora que se tenga disponible.

Pero el proposito fundamental del paquete es desde el punto de vista acàdemico, por lo que las limitaciones establecidas no son tan importantes.

El paquete está diseñado para todo tipo de usuarios, por lo que no es necesario que el usuario conozca BASIC para poder utilizarlo. Lo unico necesario para poder utilizar el paquete es prender la computadora e insertar el diskete conteniendo el paquete y le cual presentará la siguiente pantalla de presentación;

> **ALGORITMO**
> **DE**
> **DESCOMPOSICION**
> **(De Dantzig y Wolfe)**
>
> **Proyecto de Tesis**
>
> **DE**
>
> **Jose Efren Martinez Romero**
>
> **presione (c) para continuar**

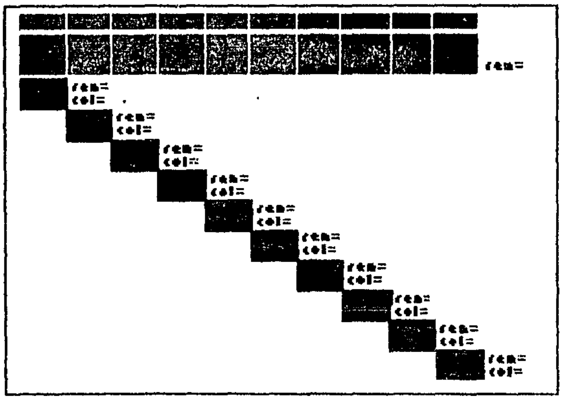
La cual permanecerá hasta que el usuario presione "c" para continuar, lo cual implica cargar en memoria el programa encargado de capturar las dimensiones de P.P.L. con la estructura angular.

Para fines prácticos sea el siguiente problema de programación lineal;

$$
\begin{array}{lrrrrrrl}
\text{Min } Z = & -X_1 & - & 2X_2 & - & X_3 & & \\
\text{s.a.} & & & & & & & \\
 & X_1 & + & X_2 & + & X_3 & <= & 12 \\
 & -X_1 & + & X_2 & & & <= & 2 \\
 & -X_1 & + & 2X_2 & & & <= & 8 \\
 & & & & & X_3 & <= & 3 \\
 & & & & & & & \\
 & X_1 & , & X_2 & , & X_3 & >= & 0
\end{array}
$$

el cual se resolverá conjuntamente con la presentación de las pantallas que integran el paquete.

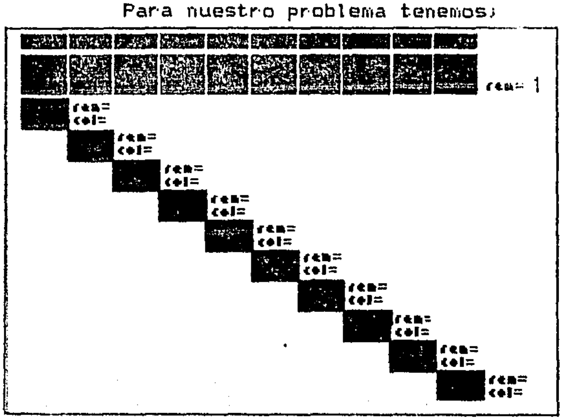
El flujo de pantallas sera;



Dar un 2 si el problema es minimizar
o un 1 si el problema es maximizar
?
Dame el numero de renglones de las restricciones generales
?
Quieres corregir el num de ren (s/n)
?
numero correcto de renglones ?

En donde se captura la información para saber si el problema es de minimización o maximización, el número de renglones de las restricciones generales y asi como la posibilidad de corregir la información dada.

Para nuestro problema tenemos;

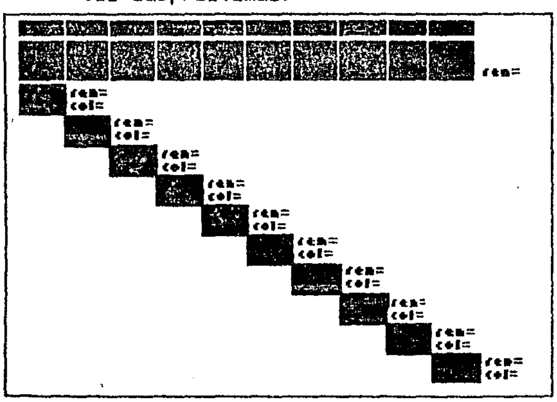


Dar un 2 si el problema es minimizar
o un 1 si el problema es maximizar
? 2
Dame el numero de renglones de las restricciones generales
? 1
Quieres corregir el num de ren (s/n)
? n
numero correcto de renglones ?

En este punto creo pertinente hacer la aclaración que el paquete valida la información dada dentro de lo razonable, es decir; verifica que se lean datos númericos o altanumericos según sea requerido y asi mismo detecta cualquier simbolo que no corresponda a un dato numerico. Pero si se da simbolos correctos y no lógicos, el paquete no detecta ese tipo de errores. Es decir datos de la forma; >5.6+9/.

Por lo que es responsabilidad del usuario el dar información lógica.

Hecha la aclaración anterior, la siguiente pantalla que se muestra tiene como finalidad la de capturar las dimensiones de los subproblemas.

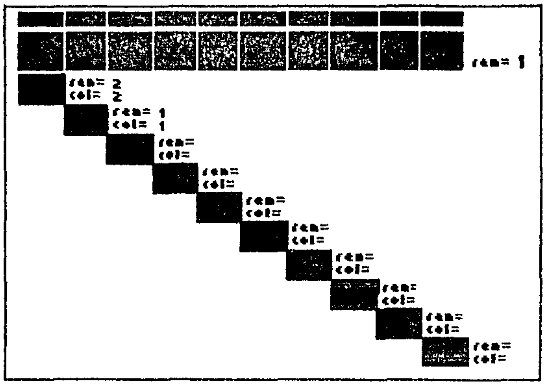


Dame dimensiones del subproblema
para terminar dar ren=0 y col=0
numero de renglones ?
numero de columnas ?

Quieres corregir alguna dimension (s/n) ?
Dame numero de subproblema
de subproblema ?
de renglones ?
de columnas ?
Quieres repetir proceso (s/n)
?
Quieres borrar o agregar algun subp
(s/n) ?

La cual captura dimensiones hasta que se tengan 10 subproblemas o se de 0 como dimensión del subproblema.

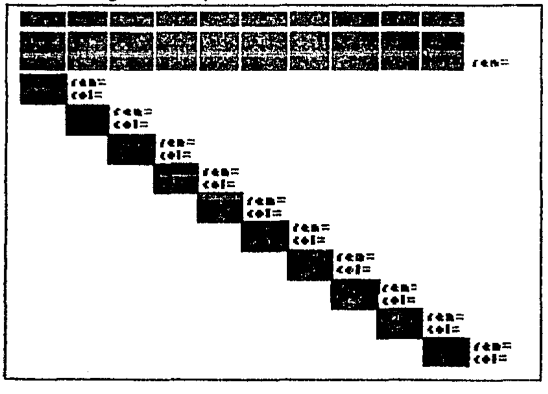
Debido a lo interactivo de la pantalla se mostrará la última pantalla resultante de la captura de las dimensiones de los subproblemas.



Dame dimensiones del subproblema 2
para terminar dar ren=0 y col=0
numero de renglones ? 1
numero de columnas ? 1
Quieres corregir alguna dimension (s/n) ? n
Dame numero de subproblema
de subproblema ?
de renglones ?
de columnas ?
Quieres repetir proceso (s/n)
? n
Quieres borrar o agregar algun subp
(s/n) ? n

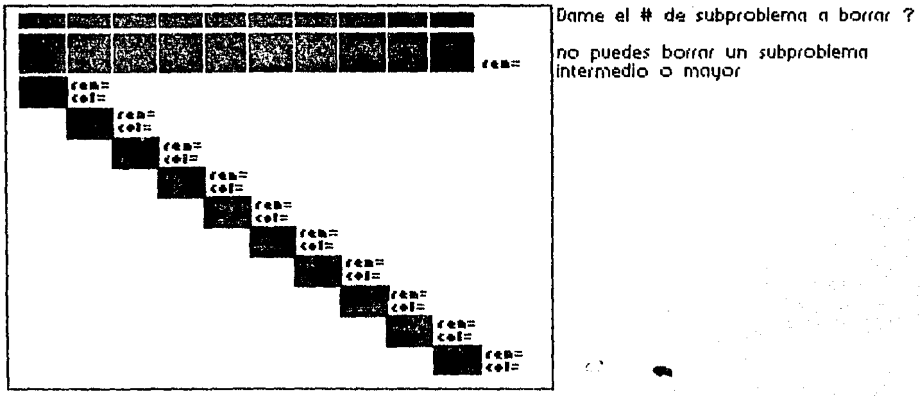
Una vez capturadas las dimensiones de los subproblemas, es posible corregirlas si se desea y la cual se muestra como opción.

Dado que es posible que se desee agregar o borrar algún subproblema, tambien se tiene la opción. Por lo que para saber que se desea hacer (borrar o agregar subproblemas), se muestra la siguiente pantalla;

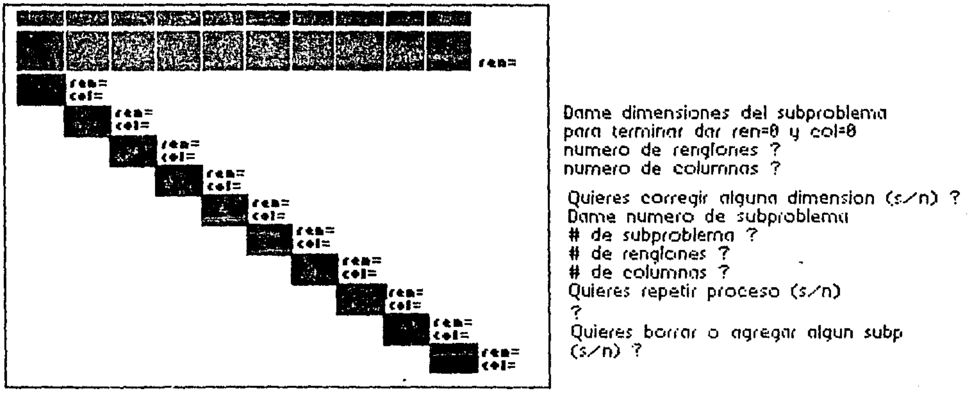


Quieres borrara o agregar (b/a) ?

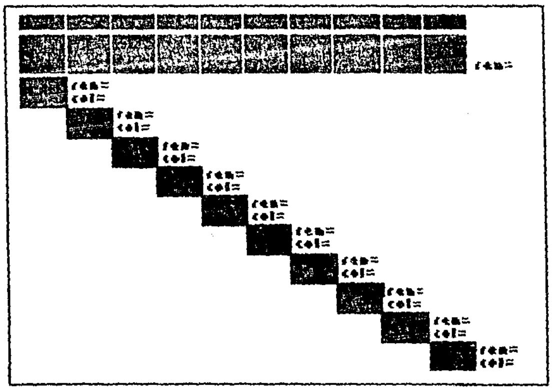
En nuestro problema se tiene el número correcto de subproblemas, pero si se desea borrar un subproblema aparecerá la siguiente pantalla;



y si se desea agregar algún subproblema, aparecera la siguiente pantalla;



Al finalizar de dar la información y al estar está según lo requiere el usuario, aparecerá la siguiente pantalla;



presiona c para continuar ?

La cual da como terminada la captura de la información del P.P.L. y espera que el usuario de "c" para continuar con la captura de los coeficientes y variables de los subproblemas, restricciones generales y función objetivo del P.P.L.

La captura de información de los subproblemas, se inicia con la siguiente pantalla;

Dame las variables separadas
por un espacio del subproblema I
?

(I)

en la cual se dara las variables utilizadas en el subproblema I

(I representa el número de subproblema a considerar). La forma de leer toda la información en el paquete será por renglones, es decir; se dará una variable o número y espacio, variable o número y espacio y asi sucesivamente hasta dar toda la información requerida.

La pantalla para el problema es;

```
Dame las variables separadas
por un espacio del subproblema 1
? X1 X2
```

Una vez capturadas las variables se muestran para su posible corrección, por lo que se presenta la siguiente pantalla;

```
Dame las variables separadas
por un espacio del subproblema 1
?
las variables capturadas son

quieres corregir o agregar datos (s/n)
dame el renglon a corregir
dame la columna a corregir
Dame la variable o dato correcto
```

(II)

para el problema queda como;

```
Dame las variables separadas
por un espacio del subproblema 1

las variables capturadas son
X1 X2
quieres corregir o agregar datos (s/n) n
dame el renglon a corregir
dame la columna a corregir
Dame la variable o dato correcto
```

si se realizo alguna corrección se muestran las variables corregidas con la siguiente pantalla;

```
el problema corregido es;




deseas corregir algun dato (s/n)
```

(III)

Teniendo las variables correctas para el subproblema se capturarán sus coeficientes, restricciones y términos independientes, por medio de la siguiente pantalla;

```
Dame los coeficientes y restricciones
del subproblema 1
DE LA FORMA A B <= C donde A,B y C
SON LOS COEFICIENTES DEL SUBPROBLEMA
?
```

(IX)

para el problema queda;

```
Dame los coeficientes y restricciones
del subproblema 1
DE LA FORMA A B <= C donde A,B y C
SON LOS COEFICIENTES DEL SUBPROBLEMA
? -1 + 1 <= 2
? -1 + 2 <= 8
```

Teniendo el subproblema capturado se presentará este para su posible corrección;

```
Dame los coeficientes y restricciones
del subproblema 1
DE LA FORMA A B <= C donde A,B y C
SON LOS COEFICIENTES DEL SUBPROBLEMA
EL PROBLEMA CAPTURADO ES
   -1 + 1 <= 2
   -1 + 2 <= 3

quieres corregir o agregar datos (s/n)
dame el renglon a corregir
dame la columna a corregir
Dame la variable o dato correcto
```

(IV)

Una vez mostrado los coeficientes del subproblema y hecha alguna corrección se mostrará la pantalla (III) para realizar mas correcciones si se desean.

En nuestro subproblema se tiene un error y la pantalla que tendriamos con su corrección es;

```
Dame los coeficientes y restricciones
del subproblema 1
DE LA FORMA A B <= C donde A,B y C
SON LOS COEFICIENTES DEL SUBPROBLEMA
EL PROBLEMA CAPTURADO ES
   -1 + 1 <= 2
   -1 + 2 <= 3

quieres corregir o agregar datos (s/n) s
dame el renglon a corregir  2
dame la columna a corregir  4
Dame la variable o dato correcto 8
```

y tendriamos el problema corregido en la siguiente pantalla;

```
el problema corregido es;
-1 + 1 <= 2
-1 + 2 <= 8



deseas corregir algun dato (s/n)
```

Por lo que la captura de información de los subproblemas, está determinada por la secuencia de las pantallas (I),(II),(III),(IV) y (V) mostrandose estas tantas veces como número de subproblemas se tenga.

Para nuestro ejemplo el segundo ejempplo se capturará de forma análoga al primero.

Para la captura de la función objetivo se mostrará la pantalla;

```
DAME LA FUNCION OBJETIVO
separando los coeficientes
por un espacio en blanco
?
```

en nuestro ejemplo;

```
DAME LA FUNCION OBJETIVO
separando los coeficientes
por un espacio en blanco
? -1  -2  -1
```

Teniendo los coeficientes de la función objetivo capturados, se presentarán estos para su posible corrección;

```
DAME LA FUNCION OBJETIVO
separando los coeficientes
por un espacio en blanco
EL PROBLEMA CAPTURADO ES




quieres corregir o agregar datos (s/n) ?
dame el renglon a corregir
dame la columna a corregir
Dame la variable o dato correcto
```

en nuestro ejemplo;

```
DAME LA FUNCION OBJETIVO
separando los coeficientes
por un espacio en blanco
EL PROBLEMA CAPTURADO ES
-1 -2 -1



quieres corregir o agregar datos (s/n) ? n
dame el renglon a corregir
dame la columna a corregir
Dame la variable o dato correcto
```

Si se efectuo alguna corrección a la función objetivo se mostrará la pantalla (III), o de lo contrario se proseguirá con la captura de las restricciones generales por medio de la siguiente pantalla;

```
DAME LAS RESTRICCIONES GENERALES
separando los coeficientes y
restricciones por un espacio en blanco
?
```

en nuestro ejemplo;

```
DAME LAS RESTRICCIONES GENERALES
separando los coeficientes y
restricciones por un espacio en blanco
? 1 1 1 <= 12
```

De manera análoga a lo anterior se mostrará lo capturado para su posible corrección, por lo que se presentará la siguiente pantalla;

```
DAME LAS RESTRICCIONES GENERALES
separando los coeficientes y
restricciones por un espacio en blanco
El PROBLEMA CAPTURADO ES




quieres corregir o agregar datos (s/n) ?
dame el renglon a corregir
dame la columna a corregir
Dame la variable o dato correcta
```

en nuestro ejemplo;

```
DAME LAS RESTRICCIONES GENERALES
separando los coeficientes y
restricciones por un espacio en blanco
El PROBLEMA CAPTURADO ES
1 1 1 <= 12



quieres corregir o agregar datos (s/n) ? n
dame el renglon a corregir
dame la columna a corregir
Dame la variable o dato correcta
```

si se efectuo alguna corrección se mostrará la pantalla (III) y con la opción de continuar corrigiendo la información capturada.

Por medio de las pantallas anteriores se tiene toda la información del P.P.L. y se muestra este con la siguiente pantalla, en la cual se tiene la opción de recapturar el Problema de Programación Lineal;

```
El problema de P.L. capturado es

      Z =
      S.a.



las variables capturadas son

Esta correcto el P.P.L. (s/n) ?
```

en nuestro ejemplo;

```
El problema de P.L. capturado es

 Min   z =  -1  -2  -1
       s.a.
              1   1   1  <= 12
             -1   1   <= 2
             -1   2   <= 8
                      1  <= 3


las variables capturadas son
X1  X2  X3

Esta correcto el P.P.L. (s/n) ? s
```

Si se responde en forma negativa se retornará a la pantalla (I), de lo contrario se proseguirá con la resolución del P.P.L.

Para lo cual se tienen 2 opciones:

1) Conocer el incremento de la solución del P.P.L. en cada iteración del Algoritmo de Descomposición, para lo cual se desplegará el valor de las variables basicas y el valor de la función objetivo Z.

2) Conocer únicamente la solución óptima del P.P.L., obteniendo como resultado la solución en términos de las variables basicas y el valor óptimo de la función objetivo.

Para nuestro ejemplo seleccionamos la opción 2) con la siguiente pantalla;

Procesando
Quieres ir viendo el incremento de la so-
lucion del problema (s/n) ? N

y en 5a iteración, el Algoritmo de Descomposición nos muestra la siguiente pantalla con la solución del P.P.L.:

Procesando
Iteracion # 5
la solucion optima del problema original es;
LAS SOLUCIONES SON;
X1= 5.31612904
X2= 6.68387097
X3= 0

el valor minimo de Z es; -18.683871

Quieres repetir proceso (s/n) ? N

En caso afirmativo de querer repetir proceso, se volvera a resolver otro Problema de Programación Lineal.

BIBLIOGRAFIA

LINEAR PROGRAMMING AND EXTENSIONS

Dantzig, George Bernard

Princeton N.J.

Princeton University, 1963.

Chapter 23

A Decomposition Principle

for Linear Programs.

***

METHODS AND APPLICATIONS OF LINEAR PROGRAMMING

L. Cooper and D. Steinberg

Philadelphia; Saunders, 1974.

Chapter 13

13-3 The Decomposition Principle of

Dantzig and Wolfe.

***

METODOS Y MODELOS DE INVESTIGACION DE OPERACIONES

Vol. 1 Modelos Deterministicos

Prawda Witenberg, Juan

Mèxico; Limusa, 1976

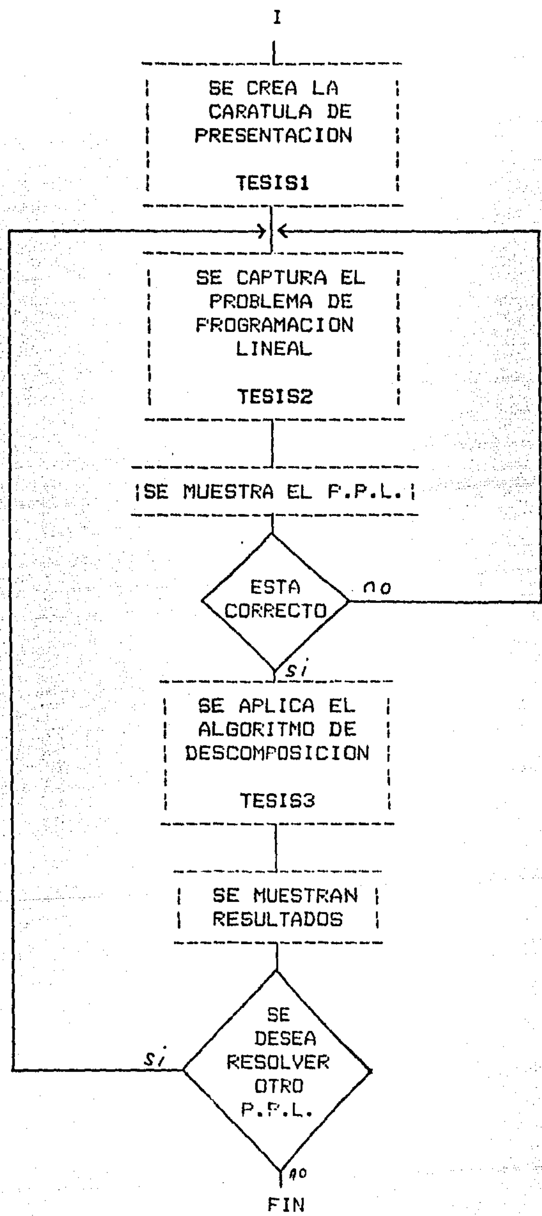
Capitulo 2

2.6. Metodos de Descomposicion Lineal.

***

APENDICE

Diagrama de flujo para el Algoritmo de Descomposición



```
10 '-----------------------------------------------------------
20 '
30 ' Nombre del programa: TESIS2
40 ' Objetivo : Capturar el problema de programacion lineal,
50 '            agregando variables de holgura y generando la
60 '            base factible inicial.
70 '
80 '-----------------------------------------------------------
90  COMMON  VS%( ),SP,RG,TP,X(   ),TT,DT,VV,VW,AC%(   ),VI(   ),AS%(
    ),AUX,AA,LL,WX,CQ
100 COMMON RC( ),RS( ),FZ(   ),PM,MS,SC,PM, VS$( ),PE
110 '  MS CONTIENE EL MAXIMO DE COLUMNAS DE LOS SUBPROBLEMAS  EN
     FORMA ESTANDAR
120 ' SC CONTIENE LA SUMA DE COLUMNAS DE LOS SUBPROBLEMAS
130 ' PM INDICA SI TENGO VARIABLES ARTIFICIALES (PM=1)
140 ' PE CONTIENE EL FACTOR CON EL CUAL SE PENALIZA  LA  FUNCION
     OBJETIVO
150 '  LA  MATRIZ  VS$( ) CONTIENE LAS  VARIABLES  DEL  PROBLEMA
     ORIGINAL
160 CLS
170 GOSUB 4340
180 ' SE CREA LA MASCARA DE CAPTURA
190 GOSUB 2730
200 '  SE  CAPTURA  DIMENSIONES DEL  PROBLEMA  Y  SE  VALIDA  LA
     INFORMACION
210 GOSUB 2980
220 ' SE CAPTURA LOS ELEMENTOS DE LOS SUBPROBLEMAS Y  SE  GUARDAN
    EN LAS MATRICES CORRESPONDIENTES
230 GOSUB 430
240  SCREEN 2:SCREEN 0:PRINT "El problema de P.P.L.  que me diste
    es;"
250 IF TP=1 THEN PRINT "Max z= "; ELSE PRINT "Min z= ";
260 FOR H=1 TO 1+RG+AV
270 FOR H1=1 TO WB+2:DC=VAL(MO$(H,H1)):DS$=MO$(H,H1):DV$=MID$(DS$
    ,3,1)
280 IF DV$="=" OR DV$=">" OR DV$="<" OR DV$="0" THEN DC=1
290 IF DC<>0 THEN PRINT DS$;" "; ELSE PRINT "    ";
300 NEXT H1
310 PRINT
320 IF H=1 THEN PRINT "s.a"
330 NEXT H
340 PRINT "Las variables son"
350 FOR H=1 TO SP
360 FOR H1=1 TO VS%(H,2)
370 PRINT VS$(H,H1);" ";
380 NEXT H1
390 NEXT H
400 PRINT:INPUT "Esta correcto el P.P.L. (s/n)";C$
410 IF C$="s" OR C$="S"  THEN CHAIN  "tesis3",ALL,DELETE  10-4410
    ELSE  PRINT  " Vuelve  a  teclear  el  P.P.L. ":INPUT  "Da  c
    para continuar";C$:RUN "tesis2"
420 END
430  ' SUBRUTINA QUE CAPTURA LOS ELEMENTOS DE LOS SUBPROBLEMAS  Y
    SE GUARDAN LAS MATRICES CORRESPONDIENTES
440  '  ----- SE APLICA  EL  METODO  DE  PENALIZACION ---------
```

```
450 IF SP=0 THEN PRINT "No tengo subproblemas para procesar,  por
    lo que termino proceso":END
460 ' SE CALCULA EL NUMERO DE COLUMNAS DE RC
470 WX=0:AV=0:AQ=1
480 FOR I=1 TO SP
490 WX=WX+VS%(I,2)
500 NEXT I
510 WB=WX:WC=0
520 DIM X(2*WX+WX),VS$(SP,10):TT=WX:DT=WX:VV=RG+SP+1:VW=VV+1
530 ' SE CALCULA EL MAXIMO DE RENGLONES DE LOS SUBPROBLEMAS
540 X(1)=RG
550 FOR I=1 TO SP
560 X(I+1)=VS%(I,1):AV=AV+VS%(I,1)
570 NEXT I
580 N=SP+2:' PARAMETRO DE ORDENA NUMEROS
590 GOSUB 2620
600 CQ=X(SP+1)
610 AUX=CQ:AA=CQ:LL=CQ
620 ' SE  PUEDE TENER VARIABLES DE HOLGURA Y/O ARTIFICIALES
630 AR=WX:XZ=1
640 FOR I=0 TO SP:WX=WX+2*VS%(I,1):NEXT I
650 DIM  SC(RG,WX),SS(CQ,WX),SZ(WX),TS$(CQ,WX),VH(CQ,2*CQ),QR(SP,
    WX),RC(RG,WX),FZ(1,WX),VI(CQ,SP+1),RS(CQ,WX),AC%(1,2+2*SP*SP)
    ,AS%(1,2*SP*SP),ES(SP),MO$(1+RG+AV,WB+5)
660 ' PRIMERO  LEO  LOS SUBPROBLEMAS EN LA MATRIZ RS
670 QM=1:L=0:WC=1+RG:TB=0
680 FOR U=1 TO SP
690 SCREEN 1:SCREEN 0
700 ' SE LEEN LAS VARIABLES DEL PROBLEMA
710 PRINT "Dame las variables separadas ":PRINT "por un  espacio
    del subproblema ";U
720 Z=1:S=1:VS=1:'vs=1 INDICA QUE VOY A LEER VARIABLES
730 WZ=VS%(U,2)+2:GOSUB 2290
740 ' SE COPIAN LAS VARIABLES LEIDAS
750 FOR H=1 TO VS%(U,2)
760 VS$(U,H)=TS$(1,H)
770 NEXT H
780 Z=1:S=VS%(U,1)
790   SCREEN   1:SCREEN   O:PRINT  "Dame   los   coeficientes   y
    restricciones":PRINT  "del subproblema ";U:PRINT "DE LA FORMA
    A  B<=   C   donde   A,B  Y  C":PRINT  "SON LOS  COEFICIENTES
    DEL SUBPROBLEMA":VS=0:DK=2:WZ=VS%(U,2)+2:GOSUB 2290
800 FOR R=Z TO S
810 WC=WC+1
820 FOR H=1 TO WZ
830 MO$(WC,H+TB)=TS$(R,H)
840 NEXT H
850 NEXT R
860 TB=VS%(U,2)+TB
870 CW=VS%(U,1):CR=VS%(U,2)
880 GOSUB 2000
890 CX=CW:CL=U
900 GOSUB 2120
910 FOR H=1 TO VS%(U,2)
920 L=L+1
```

```
930 FOR A=1 TO VS%(U,1)
940 RS(A,L)=VAL(TS$(A,H))
950 NEXT A
960 NEXT H
970 FOR H=1 TO CM
980 L=L+1
990 FOR A=1 TO VS%(U,1)
1000 RS(A,L)=VH(A,H)
1010 NEXT A
1020 NEXT H
1030 QM=QM+1
1040 IF U=1 THEN AS%(1,U)=1:AS%(1,U+1)=VS%(U,2)+CM
1050 IF U>1 AND (U/2=U\2) THEN AS%(1,U+1)=AS%(1,U)+1:AS%(1,U+2)=
     AS%(1,U)+VS%(U,2)+CM
1060 IF U>1 AND (U/2<>U\2) THEN AS%(1,U+2)=AS%(1,U+1)+1:AS%(1,U+
     3)=AS%(1,U+2)+VS%(U,2)+CM-1
1070 ' SE COPIAN LOS ELEMENTOS INDEPENDIENTES
1080 FOR H=1 TO VS%(U,1)
1090 VI(H,U+1)=VAL(TS$(H,J))
1100 NEXT H
1110 QP=0
1120 VS%(0,2)=0:Z=1:S=RG:VS%(0,1)=RG:FOR IV=1 TO SP:VS%(0,2)=VS%
     (0,2)+VS%(IV,2):NEXT IV
1130 SCREEN  1:SCREEN 0:PRINT "DAME LA  FUNCION  OBJETIVO":PRINT
     "separando  coeficientes ":PRINT "por un espacio en blanco"
1140 Z=1:S=1
1150 U=0:WZ=VS%(0,2)+2:GOSUB 2290
1160 FOR H=1 TO WZ-2
1170 MO$(1,H)=TS$(1,H)
1180 NEXT H
1190 FOR FV=1 TO J
1200 SZ(FV)=VAL(TS$(1,FV))
1210 NEXT FV
1220 ' SE CALCULA EL COEFICIENTE MAXIMO DE LA FUNCION OBJETIVO
1230 N=J+1
1240 FOR FV=1 TO J
1250 X(FV)=SZ(FV)
1260 NEXT FV
1270 GOSUB 2620
1280  MP=X(J):' EN MP ESTA EL ELEMENTO MAS GRANDE DE  LA  FUNCION
     OBJETIVO
1290 SCREEN   1:SCREEN   O:PRINT   "DAME   LAS   RESTRICCIONES
     GENERALES":PRINT "separando los coeficientes y": PRINT
     "restricciones por un espacio en blanco"
1300 U=0:Z=1:S=RG:GOSUB 2290
1310 FOR R=1 TO RG
1320 FOR H=1 TO WZ
1330 MO$(R+1,H)=TS$(R,H)
1340 NEXT H
1350 NEXT R
1360 CR=0
1370 CW=RG:FOR H=1 TO SP:CR=CR+VS%(H,2):NEXT H
1380 GOSUB 2000
1390 CL=0:CX=RG:GOSUB 2120
1400 ' SE COPIA LA MATRIZ VH EN RC
```

```
1410 FOR I=1 TO RG
1420 FOR A=1 TO CM
1430 RC(I,A)=VH(I,A)
1440 NEXT A
1450 NEXT I
1460 AC%(1,1)=1:AC%(1,2)=CM
1470 W=0
1480 FOR I=1 TO SP
1490 FOR X=1 TO RG
1500 C=AS%(1,I+I-1)+CM
1510 FOR A=1 TO VS%(I,2)
1520 QP=QP+1
1530 RC(X,C)=VAL(TS$(X,QP))
1540 C=C+1
1550 N=QP
1560 NEXT A
1570 QP=W
1580 NEXT X
1590 W=N:QP=W
1600 'APUNTADORES PARA RC
1610 AC%(1,1+2*I)=AC%(1,2*I)+1
1620 ' PARA I PAR
1630 AC%(1,2+2*I)=AC%(1,2*I)+VS%(I,2)+ES(I)
1640 NEXT I
1650 '  SE GUARDA TERMINOS INDEPENDIENTES DE  LAS  RESTRICCIONES
     GENERALES
1660 FOR CZ=1 TO VS%(0,1)
1670 VI(CZ,1)=VAL(TS$(CZ,J))
1680 NEXT CZ
1690 ' SE PENALIZA LA FUNCION OBJETIVO
1700 IF TP=1 THEN PE=-ABS(MP)*10000
1710 IF TP=2 THEN PE=ABS(MP)*10000
1720 IF  ES(0)<>0 THEN FOR  I=1  TO  ES(0):FZ(1,QR(0,I))=PE:NEXT
     I:PM=1:'  SI  PM=1  => TENGO VARIABLES ARTIFICIALES Y  TENGO
     QUE MODIFICAR LA FUNCION OBJETIVO
1730 DW=ES(0)
1740 P=0
1750 FOR I=1 TO SP
1760 FOR A=1 TO VS%(I,2)
1770 DW=DW+1:P=P+1
1780 FZ(1,DW)=SZ(P)
1790 NEXT A
1800 IF ES(I)<>0 THEN 1810 ELSE 1840
1810 FOR H=1 TO ES(I)
1820 IF QR(I,H)<>0 THEN FZ(1,QR(I,H)+DW)=PE
1830 NEXT H
1840 DW=AC%(1,2*I+2)
1850 NEXT I
1860 '  COMO  EL NUMERO DE COLUMNAS AUMENTA  AL  CONSIDERAR  LAS
     VARIABLES  DE HOLGURA Y/O ARTIFICIALES,  MODIFICO EL NUMEROS
     DE COLUMNAS
1870 FOR I=1 TO SP
1880 VS%(I,2)=AS%(1,2*I)-AS%(1,2*I-1)+1
1890 NEXT I
1900 SC=0:' CONTIENE LA SUMA DE LAS COLUMNAS DE LOS SUBPROBLEMAS
```

```
1910 '  SE  CALCULA  EL  NUMERO  MAXIMO  DE  COLUMNAS  DE  LOS
     SUBPROBLEMAS EN FORMA ESTANDAR
1920 ' EL RESULTADO ESTA EN MS
1930 FOR I=1 TO SP
1940 X(I)=VS%(I,2):SC=SC+VS%(I,2)
1950 NEXT I
1960 N=SP+1
1970 GOSUB 2620
1980 MS=X(SP)
1990 RETURN
2000 '   SE VERIFICA QUE EN EL VECTOR  INDEPENDIENTE  NO HAYA
     ELEMENTOS <0
2010 ' RECIBE COMO ENTRADA CW=# DE RENGLONES, CR=# DE COLUMNAS
2020 FOR M=1 TO CW:' RENGLONES
2030 EF=VAL(TS$(M,J))
2040 IF EF>0 THEN 2100
2050 IF EF<0 THEN FOR N=1 TO CR:TS$(M,N)=STR$(-1*VAL(TS$(M,N)))
     :NEXT N
2060 TS$(M,J)=STR$(-1*VAL(TS$(M,J)))
2070 EW$=MID$(TS$(M,J-1),3,2)
2080 IF EW$=">=" THEN TS$(M,J-1)="  "+"<=":GOTO 2100
2090 IF EW$="<=" THEN TS$(M,J-1)="  "+">="
2100 NEXT M
2110 RETURN
2120 ' SE AGREGAN VARIABLES DE HOLGURA Y/O ARTIFICIALES
2130 ' RECIBE COMO ENTRADA EL NUMERO DE RENGLONES DE LA MATRIZ A
     PROCESAR
2140 ' CX= # DE RENGLONES: CL= # DE SUBPROBLEMA
2150 '  EN  LA  MATRIZ QR  PONGO  LOS  APUNTADORES  A  VARIABLES
     ARTIFICIALES
2160 '  EN  LA  MATRIZ  VH SE PONEN  VARIABLES  DE  HOLGURA  Y/O
     ARTIFICIALES
2170 FOR I=1 TO CX:FOR Z=1 TO 2*CX:VH(I,Z)=0:NEXT Z:NEXT I
2180 VH=0:R=0:C=0:CM=0:VM=0:CA=0:' CM= CUANTAS COLUMNAS AUMENTE
2190 ' VH= APUNTADOR DE VARIABLES ARTIFICIALES
2200 FOR H=1 TO CX
2210 B$=MID$(TS$(H,J-1),3,2)
2220 IF B$="<=" THEN R=R+1:CA=CA+1:VH(R,CA)=1:CM=CM+1
2230 IF B$=">=" THEN R=R+1:CA=CA+1:VH(R,CA)=-1:VH=VH+1:QR(CL,VH)
     =C
2240 IF B$=">=" THEN CM=CM+2
2250 IF B$="="  THEN R=R+1:CA=CA+1:VH(R,CA)=0:CM=CM+1:VH=VH+1:QR
     (CL,VH)=C
2260 NEXT H
2270 ES(CL)=CM:' EN LA MATRIZ ES GUARDO LAS COLUMNAS DE LA MATRIZ
     VI
2280 RETURN
2290 ' ------------- LECTURA EN FORMATO LIBRE --------------
2300 FOR R=Z TO S
2310 FOR I1=10 TO 16:LOCATE I1,1:PRINT      "
     ":NEXT I1
2320 LOCATE 4+R,1:INPUT A$
2330 D$=" ":J=0:I=0
2340 I=I+1
2350 C$=MID$(A$,I,1)
```

```
2360 IF C$=" " OR I=LEN(A$)+1 THEN  J=J+1:TS$(R,J)=D$:D$="  ":IF
     VS=0 THEN GOSUB 3690
2370 D$=D$+C$
2380 IF I<=LEN(A$) THEN 2340
2390 A$=" "
2400 NEXT R
2410 LOCATE 4,1:' SE MUESTRA EL PROBLEMA CAPTURADO
2420 IF  VS=1 THEN PRINT "las variables  capturadas  son":  ELSE
     PRINT "EL PROBLEMA CAPTURADO ES"
2430 FOR I1=1 TO 16:LOCATE 5+I1,1:PRINT  "                 ":NEXT I1
2440 LOCATE  5,1:FOR R=Z TO S:FOR EM=1 TO  WZ:PRINT  TS$(R,EM);"
     ";:NEXT EM:PRINT:NEXT R
2450 LOCATE 15,1:INPUT "quieres corregir o agregar datos(s/n)";Y$
2460 IF Y$="S" OR Y$="s" THEN PREGUNTA=1:ELSE PREGUNTA=0
2470 WHILE PREGUNTA
2480 LOCATE 16,1:INPUT "dame el renglon a corregir";C$:RP=VAL(C$)
     :GOSUB  32600
2490 LOCATE 17,1:INPUT "dame la columna a corregir";C$:CP=VAL(C$)
     :GOSUB  3260:IF BOOL=0 OR BP=1 THEN GOTO 2490  ELSE  IF  CP>
     VS%(U,2)+2 OR CP<= 0  THEN  GOSUB  3630: LOCATE 17,28: PRINT
     "      ":GOTO 2490
2500 IF   VS=1   THEN  LOCATE   18,1 : INPUT  "Dame   la   variable
     correcta"; TS$(RP,CP)  ELSE  LOCATE  18,1  :  INPUT  "dame  el
     numero correcto o restriccion";TS$(RP,CP)
2510 SCREEN 1:SCREEN 0:PRINT "el problema corregido es;"
2520 FOR R=Z TO S
2530 FOR EM=1 TO WZ
2540 PRINT TS$(R,EM);" ";
2550 NEXT EM
2560 PRINT
2570 NEXT R
2580 LOCATE 15,1:INPUT "deseas corregir algun otro dato (s/n)";Y$
2590 IF Y$="S" OR Y$="s" THEN PREGUNTA=1:ELSE PREGUNTA=0
2600 WEND
2610 RETURN
2620 ' SUBRUTINA QUE ORDENA NUMEROS
2630 IT=1
2640 PA=1
2650 IF PA>N-1 OR IT=2 THEN 2720
2660 IT=2
2670 FOR JY=1 TO (N-PA-1)
2680 IF X(JY)>X(JY+1) THEN IT=1:HO=X(JY):X(JY)=X(JY+1):X(JY+1)=HO
2690 NEXT JY
2700 PA=PA+1
2710 GOTO 2650
2720 RETURN
2730 '  SUBRUTINA  QUE  CREA  LA  MASCARA  DE  CAPTURA  DE  LOS
     SUBPROBLEMAS
2740 KEY OFF
2750 SCREEN 2
2760 LINE (1,1)-(320,1):LINE(320,1)-(320,200):LINE(320,200)-(1,
     200):LINE(1,200)-(1,1)
2770 LINE (32,3)-(272,11),,BF
2780 LINE (32,13)-(272,37),,BF
2790 X1=32:Y1=38
```

```
2800 FOR I=1 TO 10
2810 X2=X1+24:Y2=Y1+16
2820 LINE(X1,Y1)-(X2,Y2),,BF
2830 X1=X2:Y1=Y2
2840 NEXT I
2850 X=56
2860 FOR I=3 TO 37
2870 PRESET(X,I)
2880 X=X+24
2890 NEXT I
2900 LOCATE 5,35,0:PRINT "ren= ";
2910 X1=8:Y1=6:X2=8:Y2=7
2920 FOR I=1 TO 10
2930 LOCATE Y1,X1:PRINT "ren= ";
2940 LOCATE Y2,X2:PRINT "col= ";
2950 X1=X1+3:Y1=Y1+2:X2=X1:Y2=Y2+2
2960 NEXT I
2970 RETURN
2980 ' SUBRUTINA QUE CAPTURA LAS DIMENSIONES DEL PROBLEMA
2990 LOCATE 2,43,0:PRINT "Dar un 2 si el problema es minimizar"
3000 LOCATE 3,43,0:PRINT "o un 1 si el problema es maximizar"
3010 LOCATE  4,43,0:INPUT  T$:C$=T$:GOSUB  3260:IF  BOOL=0  THEN
     LOCATE 4,43:PRINT "      ";:GOTO 3010
3020 TP=VAL(T$):IF  TP>2 OR TP=0 OR TP<0 THEN GOSUB  3630:LOCATE
     4,43,0:PRINT "      ";:GOTO 3010
3030 LOCATE 6,43:PRINT "Dame el numero de renglones de las"
3040 LOCATE 7,43:PRINT "restricciones generales"
3050 LOCATE  7,67:INPUT  C$:GOSUB 3260:IF BOOL=0  OR  BP=1  THEN
     LOCATE 7,67:PRINT "      ":GOTO 3050
3060 C=VAL(C$):LOCATE 9,43:PRINT "Quieres corregir el num de  ren
     (s/n)";
3070 LOCATE  10,43:INPUT  C$:GOSUB  4250:IF  BT=1  THEN  LOCATE
     10,43:PRINT "    ";:GOTO 3070
3080 IF C$="n" OR C$="N" THEN 3110
3090 LOCATE 11,43:INPUT "numero correcto de renglones";C$:C=VAL(
     C$)
3100 GOSUB  3260:IF  BOOL=0 OR BP=1 THEN  LOCATE  11,69:PRINT  "
     ":GOTO 3090
3110 NS=1:GOSUB 3360:PRINT C;:RG=C:VS%(0,1)=RG
3120 IF RG=0 THEN RETURN
3130 SP=0:GOSUB 3800
3140 LINE (41,23)-(65,24),,BF
3150 LOCATE 22,43:PRINT "Quieres borrar o agregar algun subp";
3160 LOCATE 23,43:PRINT "                                    "
     ;:LOCATE   23,43:PRINT "(s/n)";:LOCATE  23,51:INPUT   C$:
     GOSUB 4250:IF BT=1 THEN LOCATE 23,51:PRINT "   ";:GOTO 3160
3170 IF C$="s" OR C$="S" THEN FOR I1=1 TO 23:LOCATE I1,42:PRINT "
     ":NEXT I1:ELSE GOTO 3220
3180 LOCATE 23,43:INPUT "quieres borrar o  agregar  (b/a)";C$:IF
     C$<>"a"  AND C$<>"A" AND C$<>"b" AND C$<>"B" THEN BEEP:BEEP:
     GOTO 3180
3190 IF  C$="a"  OR C$="A" THEN FOR I1=1  TO  50:NEXT  I1:LOCATE
     23,43:PRINT "                                    ";:GOSUB 3800
3200 IF  C$="b"  OR C$="B"  THEN FOR I1=1  TO  50:NEXT  I1:LOCATE
     23,43:PRINT "                                    ";:GOSUB 4190
```

```
3210 GOTO 3150
3220 LOCATE 22,43:PRINT "                                          "
     ;:LOCATE 23,43:PRINT "                                    ";
3230 LOCATE 23,42:PRINT "Presiona c para continuar";
3240 LOCATE  23,75:INPUT  C$: IF  C$<>"c"  AND  C$<>"C"  THEN
     BEEP:BEEP:LOCATE 23,75:PRINT "   ";: GOTO 3220
3250 RETURN
3260 ' SUBRUTINA QUE VALIDA DATOS
3270 BOOL=0:BP=0:'BOOL=0 INDICA CANTIDAD NO NUMERICA, bp=1 indica
     NUMERO FUERA DE RANGO
3280 FOR Y=1 TO LEN(C$)
3290 Z$=MID$(C$,Y,1):IF Z$="." OR Z$="-"  THEN  BOOL=BOOL+1 :GOTO
     3320
3300 VN=ASC(Z$)
3310 IF (VN>=48 AND VN<=57) THEN BOOL=BOOL+1
3320 NEXT Y
3330 IF BOOL>LEN(C$) THEN BOOL=0:GOSUB 3630
3340 IF VAL(C$)<=0 OR VAL(C$)>9 THEN BP=1:GOSUB 3630
3350 RETURN
3360 ' SUBRUTINA QUE POSICIONA EL CURSOR EN EL PROBLEMA DESEADO
3370 ' COMO ENTRADA NECESITA EL # DE SUBPROBLEMA Y LO RECIBO EN
     NS
3380 ' EL SUBPROBLEMA O ES EL DE LAS RESTRICCIONES GENERALES
3390 ON NS GOSUB 3410,3430,3450,3470,3490,3510,3530,3550,3570,
     3590,3610
3400 RETURN
3410 LOCATE 5,39,0:' PARA LAS RESTRICCIONES GENERALES
3420 RETURN
3430 IF PO=1 THEN LOCATE 6,12,0 ELSE LOCATE 7,12,0
3440 RETURN
3450 IF PO=1 THEN LOCATE 8,15,0 ELSE LOCATE 9,15,0
3460 RETURN
3470 IF PO=1 THEN LOCATE 10,18,0 ELSE LOCATE 11,18,0
3480 RETURN
3490 IF PO=1 THEN LOCATE 12,21,0 ELSE LOCATE 13,21,0
3500 RETURN
3510 IF PO=1 THEN LOCATE 14,24,0 ELSE LOCATE 15,24,0
3520 RETURN
3530 IF PO=1 THEN LOCATE 16,27,0 ELSE LOCATE 17,27,0
3540 RETURN
3550 IF PO=1 THEN LOCATE 18,30,0 ELSE LOCATE 19,30,0
3560 RETURN
3570 IF PO=1 THEN LOCATE 20,33,0 ELSE LOCATE 21,33,0
3580 RETURN
3590 IF PO=1 THEN LOCATE 22,36,0 ELSE LOCATE 23,36,0
3600 RETURN
3610 IF PO=1 THEN LOCATE 24,39,0 ELSE LOCATE 25,39,0
3620 RETURN
3630 ' SUBRUTINA QUE INDICA QUE SE LEYO UN DATO ERRONEO
3640 IF  VS=1  OR  DK=2 THEN LOCATE  22,1  ELSE  LOCATE  24,45:'
     PREGUNTO SI ESTOY EN PANTALLA DE ALTA RESOLUCION O DE TEXTO
3650 PRINT "dato erroneo";:FOR I=1 TO 3:BEEP:NEXT I
3660 FOR I=1 TO 20:NEXT I
3670 IF VS=1 OR DK=2 THEN LOCATE 22,1:PRINT  "                  ";
     ELSE LOCATE 24,45:PRINT "              ";
```

```
3680 RETURN
3690 ' CHECO QUE TENGA DATOS VALIDOS
3700 DS=0
3710 FOR I6=1 TO LEN(TS$(R,J))
3720 K=0
3730 Z$=MID$(TS$(R,J),I6,1)
3740 IF (K<=15) THEN K=K+1 ELSE GOTO 3770
3750 IF Z$=CV$(K) THEN DS=DS+1:GOTO 3770
3760 GOTO 3740
3770 NEXT I6
3780 IF DS<>LEN(TS$(R,J)) THEN GOSUB 4290
3790 RETURN
3800 ' SE CAPTURA LAS DIMENSIONES DE LOS SUBPROBLEMAS
3810 CONDI=1
3820 ' SP=0:' INDICA EL NUMERO DE SUBPROBLEMAS LEIDOS
3830 WHILE CONDI
3840 SP=SP+1
3850 LOCATE 9,43:PRINT "Dame dimensiones del subproblema ";SP
3860 LOCATE 10,43:PRINT "para terminar da ren=0 y col=0";
3870 LOCATE 11,43:PRINT "numero de renglones ":LOCATE 11,62:
     PRINT "      ":LOCATE 11,62:INPUTC$:C=VAL(C$):
     PO=1:NS=SP+1:GOSUB 3360:PRINT C;:CR=C
3880 IF C$<>"0" THEN GOSUB 3260:IF BOOL=0 OR BP=1 THEN GOSUB
     3360:PRINT "  ":GOTO 3870
3890 LOCATE 12,43:PRINT "numero de columnas      ":LOCATE
     12,62:PRINT "      ":LOCATE 12,62:INPUT C$:C=VAL(C$):CC=C
3900 PO=2:GOSUB 3360:PRINT C;:' R=C
3910 IF C$<>"0" THEN GOSUB 3260:IF BOOL=0 OR BP=1 THEN GOSUB
     3360:PRINT "  ":GOTO 3890
3920 IF ((CR=0 AND CC<>0) OR (CR<>0 AND CC=0)) OR ((CR>0 AND
     CC<0) OR (CR<0 AND CC>0)) OR ((CR<0 AND CC<0)) THEN
     GOSUB 3630:PO=1:GOSUB 3360:PRINT "   ":PO=2:GOSUB 3360:
     PRINT "     ":GOTO 3850
3930 VS%(SP,1)=CR:VS%(SP,2)=CC:VS%(SP,3)=CC
3940 IF ((R=0 AND C=0) OR SP=11) THEN CONDI=0:GOTO 3960
3950 LOCATE 11,62:PRINT "   ":LOCATE 12,62:PRINT "   "
3960 WEND
3970 SP=SP-1:' SE DECREMENTA EN 1 LOS SUBPROBLEMAS LEIDOS
3980 LOCATE 14,43:PRINT "Quieres corregir alguna dimension(s/n)"
3990 LOCATE 14,78:LOCATE 15,42:INPUT C$:GOSUB 4250:IF BT=1 THEN
     LOCATE 15,42:PRINT "     ";:GOTO 3990
4000 IF C$="n" OR C$="N" THEN RETURN ELSE CONDI=1
4010 LOCATE 15,43:PRINT "Dame el numero de subproblema"
4020 WHILE CONDI
4030 LOCATE 17,65:PRINT "   ":LOCATE 18,65:PRINT "   ":LOCATE
     19,65:PRINT "   "
4040 LOCATE 17,43:PRINT "# de subproblema ":LOCATE 17,65:INPUT
     C$:C=VAL(C$):GOSUB 3260:IF BOOL=0 THEN GOSUB 3360:PRINT
     PRINT "    ";:GOTO 4040 ELSE NS=C+1
4050 IF C>SP THEN GOSUB 3630:LOCATE 17,65:PRINT "     ":GOTO 4040
4060 GOSUB 3260:IF BP=1 THEN LOCATE 17,65:PRINT "     ":GOTO 4040
4070 LOCATE 18,43:PRINT "# de renglones":LOCATE 18,65:INPUT
     C$:C=VAL(C$)
4080 GOSUB 3260:IF BOOL=0 OR BP=1 THEN LOCATE 18,65:PRINT "
     ":GOTO 4070
```

```
4090 PO=1:GOSUB 3360:PRINT "      ":GOSUB 3360:PRINT C
4100 VS%(NS-1,1)=C
4110 LOCATE  19,43:PRINT  "#  de  columnas":LOCATE   19,65:INPUT
     C$:C=VAL(C$)
4120 GOSUB  3260:IF  BOOL=0 OR BP=1 THEN  LOCATE  19,65:PRINT  "
     ":GOTO 4110
4130 PO=2:GOSUB 3360:PRINT "      ":GOSUB 3360:PRINT C
4140 VS%(NS,2)=C:VS%(NS,3)=C
4150 LOCATE 20,43:PRINT "Quieres repetir proceso  (s/n)";:LOCATE
     21,42:PRINT  "   ";:LOCATE 21,43:INPUT C$:GOSUB 4250:IF BT=1
     THEN LOCATE 21,42:PRINT "   ";:GOTO 4150
4160 IF C$="n" OR C$="N" THEN CONDI=0
4170 WEND
4180 RETURN
4190 ' SUBRUTINA QUE DA DE BAJA UN SUBPROBLEMA
4200 IF  SP=0  THEN LOCATE 4,43:PRINT  "Has  borrado  todos  los
     subproblemas":LOCATE 5,43:PRINT "por   lo   que   no   puede
     continuar":LOCATE   6,43:PRINT  "vuelve  a  comenzar   desde
     el principio":FOR I1=1 TO 5:BEEP:NEXT I1:GOTO 4200
4210 LOCATE  4,77:PRINT "    ";:LOCATE 4,43:INPUT "Dame el #  de
     subproblema a borrar";SZ
4220 IF SZ<SP OR SZ>SP THEN LOCATE 5,43:PRINT "no puedes  borrar
     un  subproblema ";:LOCATE 6,43 :PRINT "intermedio o  mayor "
     :FOR I1=1 TO 5:BEEP:NEXT I1:LOCATE 5,43:PRINT "            "
     ;:LOCATE 6,43:PRINT "                  ";:GOTO 4210
4230 IF  SZ=SP  THEN  SP=SP-1: NS=SP+2:PO=1:GOSUB  3360:PRINT  "
     ";:PO=0:GOSUB 3360:PRINT "   ";
4240 RETURN
4250 ' SUBRUTINA QUE CHECA QUE LEA LA RESPUESTA CORRECTA
4260 BT=0
4270 IF  C$<>"n"  AND  C$<>"N"  AND  C$<>"s"  AND  C$<>"S"  THEN
     BEEP:BEEP:BT=1
4280 RETURN
4290 ' SUBRUTINA QUE CORRIGE LOS DATOS ERRONEOS CAPTURADOS COMO
     COEFICIENTES DE LOS SUBPROBLEMAS
4300 BEEP:BEEP:LOCATE 15,1:PRINT "                              "
     ;:LOCATE 15,1:PRINT "dato incorrecto en la posicion ";J
4310 LOCATE 16,1:PRINT "                                    "
     ;:LOCATE 16,1:INPUT "dame la cantidad correcta ";TS$(R,J)
4320 GOSUB 3690
4330 RETURN
4340 DIM  VS%(10,3),CV$(16):'  EN EL VECTOR CV$(   )  GUARDO  LOS
     ELEMENTOS DE VALIDACION
4350 ' SE LEEN LOS DATOS DE VALIDACION
4360 FOR K=1 TO 15
4370 READ CV$(K)
4380 NEXT K
4390 DATA 0,1,2,3,4,5,6,7,8,9,.,-,<,>,=
4400 CV$(16)=" "
4410 RETURN
```

```
10 '--------------------------------------------------------------
20 '
30 ' Nombre del programa : TESIS3
40 ' Objetivo :Resolver el problema de programacion lineal
45 '           aplicando  el Algoritmo de Descomposicion de
50 '           de  Dantzig  y Wolfe.
60 '
70 '--------------------------------------------------------------
80 DIM B(2*VV,2*VW),MI(VV,VW),W1(VV,2*VW),W2(VV,2*VW),ZI(2*VV+VW),
   AN(3*(RG+SP)),SU(2*TT)
90 DIM MU(VV,2*VW):DIM ZA(2*SP,3*TT),PU(3*TT),RZ(SP),BN(SP)
100 ' EN IB SE PONEN LAS VARIABLE4S BASICAS Y NO BASICAS
110 DIM IB(DT+DT):FOR A=1 TO TT:IB(A)=A:NEXT A
120 DIM BV(SP,DT+RG)
130 ' SE INICIALIZA EL VECTOR CON VARIABLES BASICAS
140 ' SE LIMPIA LA PANTALLA ANTES DE PRESENTAR RESULTADOS
150 SCREEN 0:SCREEN 1
160 PRINT "                PROCESANDO":BEEP:BEEP
170 ' ---------------------------------------------
180  PRINT  "Quieres ir viendo el incremento de la  solucion  del
     Problema (s/n)"
190 INPUT C$
200 IF C$="s" OR C$="S" THEN RASTREO=1 ELSE RASTREO=0
210 ' SE DETERMINAN LAS VARIABLES BASICAS
220 ' LAS CUALES INICIAN CON @ Y X
230 DIM VA$(RG+SP),VA(RG+SP,2*TT),VV$(RG+SP,2),ZX(DT+RG),VP(RG+S
    P,2*TT)
240 FOR I=1 TO RG
250 VA$(I)=CHR$(64)+STR$(0)+STR$(I):VV$(I,1)=VA$(I):'  SE  CREAN
    XA,XB,..,XZ
260 NEXT I
270 ' SE CREAN @A,@B,..,@Z
280 FOR I=1 TO SP
290 VA$(RG+I)=CHR$(64)+STR$(I)+CHR$(64+I):VV$(RG+I,1)=VA$(RG+I)
300 NEXT I
310 ' EN VA SE VAN A GUARDAR LOS PUNTOS EXTREMOS  GENERADOS  POR
    LAS TABLAS SIMPLEX
320 ' EN VA SE INICIALIZA CON LOS PUNTOS EXTREMOS DE LAS LAMBDAS
330 FOR IL=1 TO RG+SP:FOR JL=1 TO MS:VA(IL,JL)=0:NEXT JL:NEXT IL
340 FOR IL=1 TO RG+SP
350 X=VAL(MID$(VV$(IL,1),3,1)):UW=VS%(X,2)
360 IF X<>0 THEN FOR JL=VS%(X,1) TO 1 STEP -1:VA(IL,UW)=VI(JL,X+
    1):VP(IL,UW)=VA(IL,UW):UW=UW-1:NEXT JL
370 NEXT IL
380 ' SE CREA LA MATRIZ EN DONDE SE VA A GUARDAR LA BASE  DE  LA
    TABLA SIMPLEX
390 DIM BA(RG+SP,RG+SP),A(RG+SP,RG+SP)
400 ' AL PRINCIPIO BA ES UNA MATRIZ IDENTIDAD
410 ' SE CREA LA MATRIZ IDENTIDAD
420 FOR I=1 TO RG+SP
430 FOR J=1 TO RG+SP
440 IF I=J THEN BA(I,J)=1:ELSE BA(I,J)=0
450 A(I,J)=BA(I,J)
460 NEXT J
470 NEXT I
```

```
480 ZT=0
490 DIM CB(RG+SP),PI(RG+SP),RE(DT+RG+2*SP),WO(DT+RG),AI(DT+RG,DT
    +RG),BQ(DT+RG),XB(RG+SP),AB(RG+SP)
500  FOR  I=1 TO RG+SP:CB(I)=PE:NEXT I:' INICIO EL VECTOR CB  CON
LOS COEFICIENTES DE PENALIZACION EN PROB ORIG.
510 ' SE CALCULA PI=CB*B-1
520 ZT=ZT+1:' SE CUENTAN ITERACIONES
530 SCREEN 0:SCREEN 1
540 LOCATE 1,8:PRINT "Procesando ":LOCATE 1,20:PRINT "Iteracion #
    ",ZT:BEEP:BEEP
550 FOR I=1 TO RG
560 XB(I)=VI(I,1):AB(I)=XB(I)
570 NEXT I
580 FOR I=RG+1 TO RG+SP
590 XB(I)=1:AB(I)=XB(I)
600 NEXT I
610 ' EN A ESTA LA MATRIZ INVERSA PEDIDA
620 FOR I=1 TO RG+SP:PI(I)=0:NEXT I
630 ' SE CALCULA CB*B-1
640 FOR I=1 TO RG+SP
650 FOR J=1 TO RG+SP
660 PI(I)=CB(J)*A(J,I)+PI(I)
670 NEXT J
680 NEXT I
690 ' SE CALCULA (CP-PIAP)XP
700 ' EN EL VECTOR ZI SE VA A GUARDAR LA FUNCION OBJETIVO
710 X=0:R=0:S=2:H=0:D=2
720 FOR I=1 TO SP:' # DE FUNCIONES OBJETIVO A CALCULAR
730 FOR V=1 TO RG+SP:RE(V)=0:NEXT V
740 ' SE CALCULA PI*AP
750 FOR A=AC%(1,S+I) TO AC%(1,S+I+1)
760 R=R+1
770 FOR B=1 TO RG
780 X=X+1
790 IF R<=(RG+SP) THEN RE(R)=PI(X)*RC(X,A)+RE(R)
800 NEXT B
810 X=0
820 NEXT A
830 T=0
840 ' SE CALCULA (CP-PI*AP)
850 FOR J=AC%(1,D+I) TO AC%(1,D+I+1)
860 H=H+1:T=T+1
870 ZI(H)=FZ(1,J)-RE(T)
880 NEXT J
890 D=D+1
900 R=0:S=S+1
910 NEXT I
920 '  EN LA MATRIZ DE RESTRICCIONES DE LOS SUBPROBLEMAS  VOY  A
    GUARDAR LAS TABLAS SIMPLEX GENERADAS
930 QD=0:XV=0:D=0:ZZ=0:' EN RZ TENGO LAS NUEVAS Z'S CALCULADAS
940 ' SE VAN A RESOLVER LAS TABLAS SIMPLEX
950 FOR LS=1 TO SP:' # DE TABLAS SIMPLEX A RESOLVER
960 ' SE INICIALIZA EL VECTOR CON VARIABLES BASICAS
970 FOR L=1 TO SP:' # DE SUBPROBLEMAS
980 OP=VS%(L,2)-VS%(L,1)
```

```
990 FOR A=1 TO VS%(L,1):BV(L,A)=OP+A:NEXT A:NEXT L
1000 ' SE PASAN LOS PARAMETROS A LA SUBRUTINA SIMPLEX
1010 RE=VS%(LS,1):CO=VS%(LS,2)
1020 ' SE COPIA EN B TODA LA MATRIZ A PROCESAR
1030 FOR IZ=1 TO VS%(LS,1)
1040 S=0
1050 FOR JZ=AS%(1,LS+D) TO AS%(1,LS+D+1)
1060 S=S+1
1070 B(IZ,S)=RS(IZ,JZ)
1080 NEXT JZ
1090 NEXT IZ
1100 D=D+1:C=0
1110 FOR I=AS%(1,LS+QD) TO AS%(1,LS+QD+1):' SE COPIA LA  FUNCION
     OBJETIVO
1120 C=C+1
1130 B(VS%(LS,1)+1,C)=ZI(I):ZX(C)=ZI(I)
1140 NEXT I
1150 QD=QD+1:B(VS%(LS,1)+1,VS%(LS,2)+1)=0:'  SE PONE EN CERO  EL
     VALOR DE LA FUNCION OBJETIVO
1160 ' SE COPIA EL VECTOR INDEPENDIENTE
1170 FOR I=1 TO VS%(LS,1)
1180 B(I,VS%(LS,2)+1)=VI(I,LS+1)
1190 NEXT I
1200  '  SI TENGO VARIABLES ARTIFICIALES  MODIFICA=O  LA  FUNCION
     OBJETIVO, DONDE
1210 'SI PM=1 ENTONCES TENGO VARIABLES ARTIFICIALES
1220  IF  PM=1 THEN GOSUB 5330:' SE MODIFICA LA FUNCION  OBJETIVO
     PARA TENER UNA
1230 'BASE INICIAL FACTIBLE DE LA TABLA SIMPLEX
1240 FOR U=1 TO TT:PU(U)=0:NEXT U
1250 GOSUB 3850
1260 IF BN(LS)=0 THEN FOR GH=1 TO VS%(LS,1):PU(BV(LS,GH))=B(GH,C
     O+1):NEXT GH
1270 FOR MM=1 TO VS%(LS,2):ZA(LS,MM)=PU(MM):NEXT MM
1280 ' SE COPIA EN RS LA TABLA SIMPLEX OBTENIDA
1290 ZZ=0
1300 GOTO 1430
1310 FOR J=1 TO RE
1320 FOR I=AS%(1,XV+LS) TO AS%(1,XV+LS+1)
1330 ZZ=ZZ+1
1340 RS(J,I)=B(J,ZZ)
1350 NEXT I
1360 ZZ=0
1370 NEXT J
1380 ' SE COPIA EL VECTOR INDEPENDIENTE
1390 FOR I=1 TO RE
1400 VI(I,LS+1)=B(I,CO+1)
1410 NEXT I
1420 E=.00001:   ' INDICA LA EPSILON PARA CONSIDERAR COMO CERO UNA
     CANTIDAD
1430 IF BN(LS)<>1 THEN RZ(LS)=-1*B(RE+1,CO+1)
1440 XV=XV+1
1450 NEXT LS:' VUELVO A CALCULAR LA SIGTE TABLA SIMPLEX
1460  '   SE CALCULA Z-PI Y SE GUARDAN EN X(I) Y EN MA  SE  GUARDA
     X(SP)
```

```
1470 MA=X(SP):MI=X(1)
1480 FOR I=1 TO SP
1490 X(I)=RZ(I)-PI(RG+I)
1500 NEXT I
1510 N=SP+1:GOSUB 4410
1520 FOR I=1 TO RG+SP:PP(I,1)=0:NEXT I
1530 ' VER SI EL VECTOR CORRESPONDE A UN PUNTO EXTREMO
1540 ' IDENTIFICANDOI A QUE SUBPROBLEMA CORRESPONDE EL MAX O MIN
1550 FOR I=1 TO SP
1560 IF TP=1 THEN IF X(SP)=RZ(I)-PI(RG+I) THEN SB=I
1570 IF TP=2  THEN IF X(1)=RZ(I)-PI(RG+I) THEN SB=I
1580 NEXT I
1590 ' EN SB ESTA A QUE SUBPROBLEMA CORRESPONDE EL MIN O MAX
1600 ' EN WV DEJO LA HOLGURA A ENTRAR
1610 FH=0: ' SI FH=1 ENT. METO UNA HOLGURA DE RC
1620 FOR J=1 TO RG:' SE EVALUA SI ENTRA XOJ
1630 PW=0
1640 FOR I=1 TO RG
1650 PW=PW+PI(I)*RC(I,J)
1660 NEXT I
1670 IF TP=2 THEN IF -1*PW<X(1) THEN X(1)=-1*PW:FH=1:WV=J
1680 IF TP=1 THEN IF -1*PW>X(SP) THEN X(SP) =-1*PW:FH=1:WV=J
1690 NEXT J
1700 IF FH=1 THEN FOR I=1 TO RG:PP(I,1)=RC(I,WV):NEXT I
1710 IF FH=1 THEN FP=0
1720 IF FH=1 THEN O$=CHR$(78)+STR$(0)+STR$(WV)
1730 IF TP=1 THEN IF X(SP)<=E THEN SCREEN 0:SCREEN  1:SR=1:PRINT
     "la  solucion optima del problema original  es;":GOSUB 4870
     :GOTO 5520
1740 IF TP=2 THEN IF X(1)>=-1*E THEN SCREEN 0:SCREEN 1:SR=1:PRINT
     "la  solucion  optima del problema original es;":GOSUB  4870
     :GOTO 5520
1750 IF FH=1 THEN 1940
1760 ' SE CREA LA VARIABLE A GUARDAR I.E X(SB)(LETRA)
1770 O$=CHR$(64)+STR$(SB)+CHR$(ET)
1780 ET=ET+1:FL=0
1790 ' SE CALCULA PPJ=AP*XPJ
1800 FOR I=1 TO RG+SP:PP(I,1)=0:NEXT I
1810 ' EN SB ME VA AINDICAR QUE MATRIZ TOMAR I.E A0,A1,A2,..
1820 FOR I=1 TO RG
1830 IF I=1 AND(SB/2=SB\2) THEN FF=2*SB+1:A=0
1840 IF I=1 AND (SB/2<>SB\2) THEN FF=2*SB:A=I
1850 W=0
1860 FOR Y=AC%(1,FF+A) TO AC%(1,FF+A+1)
1870 W=W+1
1880 PP(I,1)=PP(I,1)+RC(I,Y)*ZA(SB,W)
1890 NEXT Y
1900 NEXT I
1910 IF BN(SB)=1 THEN GOTO 1940
1920 ' SE CALCULA (PIJ EI)
1930 PP(RG+SB,1)=1
1940 ' SE CALCULA B-1*(PIJ EI) Y SE GUARDA EN CN(RG+SP,1)
1950 FOR Y=1 TO RG+SP
1960 CN(Y,1)=0
1970 NEXT Y
```

```
1980 FOR I=1 TO RG+SP
1990 W=0
2000 FOR J=1 TO RG+SP
2010 W=W+1
2020 CN(I,1)=CN(I,1)+A(I,J)*PP(W,1)
2030 NEXT J
2040 NEXT I
2050 '  CALCULO EL COCIENTE Y ENCUENTRO EL MINIMO DE XB  Y  B-1
     *(PIJ EI)
2060 RL=1000000!
2070 FOR J=1 TO RG+SP
2080 IF CN(J,1)>0 THEN IF RL>AB(J)/CN(J,1) THEN RL=AB(J)/CN(J,1)
     :PR=J
2090 NEXT J
2100 IF FH=1 THEN 2200
2110 ' SE CALCULA FPJ=CP*XPJ DEJANDO EN FP
2120 FP=0
2130 IF I=1 AND (SB/2=SB\2) THEN FF=2*SB+1:A=0
2140 IF I=1 AND (SB/2<>SB\2) THEN FF=2*SB:A=I
2150 W=0
2160 FOR Y=AC%(1,FF+A) TO AC%(1,FF+A+1)
2170 W=W+1
2180 FP=FP+FZ(1,Y)*ZA(SB,W)
2190 NEXT Y
2200 V1=FP:IF ZT=1 THEN RQ=RG+SP+1 ELSE RQ=AQ
2210 ' SE METE A LA BASE LA COLUMNA GENERADA
2220 ' EN PC ESTA LA COLUMNA PIVOTE
2230 ' EN PR ESTA EL RENGLON PIVOTE
2240 AQ=PR
2250 ' SE REEMPLAZA LA VARIABLE BASICA
2260 VV$(PR,1)=O$:FOR I=1 TO VS%(SB,2):VA(PR,I)=ZA(SB,I):VP(PR,I)
     =VA(PR,I):NEXT I
2270 ' SE METE LA VARIABLE BASICA CON SU VECTOR ASOCIADO
2280 '  SE RECORRE DE LA MATRIZ B LOS ELEMENTOS  PARA  DEJAR  LA
     PENULTIMA COLUMNA LIBRE
2290 FOR I=1 TO RG+SP
2300 BA(I,PR)=PP(I,1)
2310 NEXT I
2320 FOR I=1 TO RG+SP
2330 FOR J=1 TO RG+SP
2340 A(I,J)=BA(I,J)
2350 NEXT J
2360 NEXT I
2370 ND=RG+SP:GOSUB 2510
2380 ' SE CALCULA EL NUEVO VECTOR DE TERMINOS INDEPENDIENTES
2390 FOR I=1 TO RG+SP: ' RENGLONES
2400 RL=0
2410 FOR J=1 TO RG+SP:' COLUMNAS
2420 RL=RL+A(I,J)*XB(J)
2430 NEXT J
2440 AB(I)=RL
2450 NEXT I
2460 ' SE AGREGA LA COLUMNA A CB(J)
2470 CB(PR)=V1
2480 FOR I=1 TO SP:BN(I)=0:NEXT I
```

```
2490 IF  RASTREO=1  THEN  SCREEN  0:SCREEN  1:GOSUB  4870:LOCATE
     23,1:INPUT "Presiona c para continuar";C$
2500 GOTO 510
2510 ' -------------------------------------------------------------
2520 '                SE CALCULA LA INVERSA DE UNA MATRIZ
2540 ' -------------------------------------------------------------
2550 N=ND:GOSUB 2750
2560 ' SE CHECA SI LA MATRIZ ES SINGULAR
2570 S=N:DE=1:IF ND>N THEN S=ND
2580 FOR G=1 TO S:DE=DE*A(G,G):NEXT G
2590 IF DE=0 THEN PRINT "MATRIZ SINGULAR":END
2600 ' SE CALCULA INVERSA
2610 FOR W=1 TO ND
2620 FOR L=1 TO N
2630 BQ(L)=0
2640 NEXT L
2650 BQ(W)=1
2660 GOSUB 3560
2670 FOR L=1 TO N
2680 AI(L,W)=BQ(L)
2690 NEXT L
2700 NEXT W
2710 FOR I=1 TO ND:FOR J=1 TO N:A(I,J)=AI(I,J):NEXT J:NEXT I
2720 RETURN
2730 ' -------------------------------------------------------------
2740                          'SUBRUTINA DECOMP
2750 ' -------------------------------------------------------------
2760 'DESCOMPONE UNA MATRIZ REAL POR ELIMINACION GAUSSIANA
2770 IP%(N)=1
2780 IF N=1 GOTO 3500
2790 NM=N-1
2800 AN=0
2810 FOR J=1 TO N
2820 T=0
2830 FOR I=1 TO N
2840 T=T+ABS(A(I,J))
2850 NEXT I
2860 IF (T>AN) THEN AN=T
2870 NEXT J
2880 FOR K=1 TO NM
2890 KP=K+1
2900 M=K
2910 FOR I=KP TO N
2920 IF (ABS(A(I,K))>ABS(A(M,K))) THEN M=I
2930 NEXT I
2940 IP%(K)=M
2950 IF (M<>K) THEN IP%(N)=-IP%(N)
2960 T=A(M,K)
2970 A(M,K)=A(K,K)
2980 A(K,K)=T
2990 IF (T=0) GOTO 3120
3000 FOR I=KP TO N
3010 A(I,K)=-A(I,K)/T
3020 NEXT I
3030 FOR J=KP TO N
```

```
3040 T=A(M,J)
3050 A(M,J)=A(K,J)
3060 A(K,J)=T
3070 IF T=0 GOTO 3110
3080 FOR I=KP TO N
3090 A(I,J)=A(I,J)+A(I,K)*T
3100 NEXT I
3110 NEXT J
3120 NEXT K
3130 FOR K=1 TO N
3140 T=0
3150 IF (K=1) GOTO 3200
3160 KM=K-1
3170 FOR I=1 TO KM
3180 T=T+A(I,K)*WO(I)
3190 NEXT I
3200 EK=1
3210 IF (T<0) THEN EK=-1
3220 IF (A(K,K)=0) GOTO 3520
3230 WO(K)=-(EK+T)/A(K,K)
3240 NEXT K
3250 FOR KB=1 TO NM
3260 K=N-KB
3270 T=0
3280 KP=K+1
3290 FOR I=KP TO N
3300 T=T+A(I,K)*WO(K)
3310 NEXT I
3320 WO(K)=T
3330 M=IP%(K)
3340 IF (M=K) GOTO 3380
3350 T=WO(M)
3360 WO(M)=WO(K)
3370 WO(K)=T
3380 NEXT KB
3390 YN=0
3400 FOR I=1 TO N
3410 YN=YN+ABS(WO(I))
3420 NEXT I
3430 GOSUB 3600
3440 ZN=0
3450 FOR I=1 TO N
3460 ZN=ZN+ABS(WO(I))
3470 NEXT I
3480 CO=AN*ZN/YN
3490 IF (CO<1) THEN CO=1:RETURN
3500 CO=1
3510 IF (A(1,1)<>0) THEN RETURN
3520 ' EXACT SINGULARITY
3530 CO=1E+32
3540 RETURN
3550 '-------------------------------------------------------------
3560 '                           SUBRUTINA SOLVE
3570 '-------------------------------------------------------------
3580 ' SOLUCION DE SISTEMAS REALES, AX = B
```

```
3590 ' NO  SE  DEBE  DE USAR SI  LA  SUBRUTINA  DECOMP  DETECTO
     SINGULARIDAD
3600 IF (N=1) GOTO 3820
3610 NM=N-1
3620 FOR K=1 TO NM
3630 KP=K+1
3640 M=IP%(K)
3650 T=BQ(M)
3660 BQ(M)=BQ(K)
3670 BQ(K)=T
3680 FOR I=KP TO N
3690 BQ(I)=BQ(I)+A(I,K)*T
3700 NEXT I
3710 NEXT K
3720 ' BACK SUSTITUTION
3730 FOR KB=1 TO NM
3740 KM=N-KB
3750 K=KM+1
3760 BQ(K)=BQ(K)/A(K,K)
3770 T=-BQ(K)
3780 FOR I=1 TO KM
3790 BQ(I)=BQ(I)+A(I,K)*T
3800 NEXT I
3810 NEXT KB
3820 BQ(1)=BQ(1)/A(1,1)
3830 RETURN
3840 ' -------------------------------------------------------
3850 '                   ALGORITMO SIMPLEX
3860 ' -------------------------------------------------------
3870 FOR M=1 TO CO
3880 X(M)=B(RE+1,M)
3890 NEXT M
3900 N=CO+1:GOSUB 4410:' ORDENA
3910 ' SE OBTIENE LA COLUMNA PIVOTE
3920 IF TP=1 THEN IF X(CO)<=0 THEN RETURN ELSE Q=X(CO):GOSUB 4520
3930 IF TP=2 THEN IF X(1)>=0 THEN RETURN ELSE Q=X(1):GOSUB 4520
3940 ' EN PC ESTA LA COLUMNA PIVOTE
3950 ' SE BUSCA RENGLON
3960 GOSUB 4580:' 25000
3970 IF BN(LS)=1 THEN RETURN
3980 BV(LS,PR)=IB(PC):' EN PR ESTA EL PIVOTE
3990 ' SE CREA LA MATRIZ IDENTIDAD
4000 FOR I=1 TO RE+1
4010 FOR J=1 TO RE+1
4020 IF I=J THEN MI(I,J)=1
4030 IF I<>J THEN MI(I,J)=0
4040 NEXT J
4050 NEXT I
4060 ' SE CREA LA MATRIZ DE MULTIPLICADORES
4070 FOR I=1 TO RE+1
4080 IF I=PR AND B(PR,PC)<>0 THEN MI(PR,PR)=1/B(PR,PC)
4090 IF I<>PR AND B(PR,PC)<>0 THEN MI(I,PR)=-1*B(I,PC)/B(PR,PC)
4100 NEXT I
4110 ' SE COPIA EN WI MI
4120 FOR I=1 TO RE+1
```

```
4130 FOR J=1 TO RE+1
4140 W1(I,J)=MI(I,J)
4150 NEXT J
4160 NEXT I
4170 ' SE COPIA EN W2 B
4180 FOR I=1 TO RE+1
4190 FOR J=1 TO CO+1
4200 W2(I,J)=B(I,J)
4210 NEXT J
4220 NEXT I
4230 GOSUB 4300:' 15000
4240 FOR I=1 TO RE+1
4250 FOR J=1 TO CO+1
4260 B(I,J)=MU(I,J)
4270 NEXT J
4280 NEXT I
4290 GOTO 3850
4300 ' ------ MULTIPLICACION DE MATRICES ----------
4310 FOR A=1 TO RE+1
4320 FOR B=1 TO CO+1
4330 SU=0
4340 FOR K=1 TO RE+1
4350 SU=SU+W1(A,K)*W2(K,B)
4360 NEXT K
4370 MU(A,B)=SU
4380 NEXT B
4390 NEXT A
4400 RETURN
4410 ' ----------- ORDENA NUMEROS --------------
4420 IT=1
4430 PA=1
4440 IF PA>N-1 OR IT=2 THEN 4510
4450 IT=2
4460 FOR JY=1 TO (N-PA-1)
4470 IF X(JY)>X(JY+1) THEN IT=1:HO=X(JY):X(JY)=X(JY+1):X(JY+1)=HO
4480 NEXT JY
4490 PA=PA+1
4500 GOTO 4440
4510 RETURN
4520 '
4530 P=0
4540 P=P+1
4550 IF Q=B(RE+1,P) THEN PC=P:GOTO 4570
4560 GOTO 4540
4570 RETURN
4580 ' ------ PIVOTE RENGLON --------
4590 BAN=0:'BANDERA QUE INDICA SI EL PROBLEMA ES ACOTADO O NO:  1
     ES ACOTADO, O NO ES ACOTADO
4600 L=0:DP=0
4610 L=L+1
4620 IF B(L,PC)<=0 AND L<=RE THEN 4610
4630 IF L>RE THEN 4710
4640 DI=B(L,CO+1)/B(L,PC)
4650 PR=L
4660 FOR I=1 TO RE
```

```
4670 IF B(I,PC)>0 THEN DP=B(I,CO+1)/B(I,PC):IP=I
4680 IF DP<=DI AND DP>0 THEN PR=IP:DI=DP
4690 NEXT I
4700 RETURN
4710 ' SE TIENE UN PROBLEMA NO ACOTADO
4720 ' PARA EL RAYO ALFA = 100
4730 FOR GH=1 TO VS%(LS,2):PU(GH)=0:NEXT GH
4740 FOR GH=1 TO VS%(LS,1)
4750 PU(BV(LS,GH))=-1*B(GH,PC)*100+B(GH,VS%(LS,2)+1)
4760 NEXT GH
4770 PU(PC)=100
4780 BAN=1:BN(LS)=1:'  SE ACTIVA EL INDICADOR QUE INDICA  SI  SE
     TIENE UN PROBLEMA NO ACOTADO
4790 ' CALCULO EL NUEVO VALOR DE Z CON EL RAYO EXTREMO
4800 RZ(LS)=0
4810 RV=2*LS-1:RZ=AS%(1,RV)
4820 FOR GH=0 TO CO-1
4830 RZ(LS)=RZ(LS)+PU(GH+1)*ZI(RZ+GH)
4840 NEXT GH
4850 GOTO 4700
4860 '
4870 ' SUBRUTINA QUE IMPRIME LAS SOLUCIONES OPTIMAS DEL  PROBLEMA
     ORIGINAL  EN  BASE  A  LAD  LAMBDAS  OPTIMAS  CALCULADAS  Y
     EL CORRESPONDIENTE VALOR DE Z
4880 ' SE CALCULA EL NUEVO VECTOR DE TERMINOS INDEPENDIENTES
4890 FOR I=1 TO RG+SP:'renglones
4900 RL=0
4910 FOR J=1 TO RG+SP:'columnas
4920 RL=RL+A(I,J)*XB(J)
4930 NEXT J
4940 AB(I)=RL
4950 NEXT I
4960 ZP=-1*TR(RG+SP+1,RG+SP+2)
4970 IF SR=1 OR ZT>2 THEN IF P1 <>0 THEN IF ABS(ZP)>ABS(P1) THEN
     PRINT "El problema original no tiene soluciones  factibles":
     PRINT "Z = ";ZP:RETURN
4980 FOR I=1 TO RG+SP
4990 X=VAL(MID$(VV$(I,1),3,1)):IF X=0 THEN B=RG ELSE B=VS%(X,2)
5000 NEXT I
5010 ' SE CALCULA LA COMBINACION LINEAL
5020 M=0
5030 FOR Z=1 TO (RG+SP)
5040 O=VAL(MID$(VV$(Z,1),3,1)):IF O=0 THEN VA(Z,VAL(MID$(VV$(Z,1
     ),5,1)))=1
5050 M=M+1
5060 X=VAL(MID$(VV$(Z,1),3,1)):IF X=0 THEN WL=RG ELSE WL=VS%(X,2)
5070 FOR J=1 TO WL
5080 VA(M,J)=AB(Z)*VA(M,J)
5090 NEXT J
5100 NEXT Z
5110 SOP=0:' SOP= NUMERO DE SOLUCIONES IMPRESAS
5120 PRINT "LAS SOLUCIONES SON"
5130 FOR I=0 TO SP
5140 IF I=0 THEN WL=RG ELSE WL=VS%(I,2)
5150 FOR A=1 TO WL:AN(A)=0:SU(A)=0:NEXT A:M=0
```

```
5160 FOR J=1 TO RG+SP
5170 IF I=VAL(MID$(VV$(J,1),3,1)) THEN M=M+1:AN(M)=J
5180 NEXT J
5190 IF I=0 THEN H=RG ELSE IF I<=SP THEN H=VS%(I,2)
5200 FOR A=1 TO M
5210 FOR B=1 TO H
5220 SU(B)=SU(B)+VA(AN(A),B)
5230 NEXT B
5240 NEXT A
5250 IF I<>0 THEN FOR HX=1 TO VS%(I,3):PRINT TAB(10);VS$(I,HX);"
     =";SU(HX):NEXT HX
5260 SOP=SOP+VS%(I,3)
5270 IF  SOP>15 THEN PRINT "presiona p para  continuar  listando
     soluciones":C$=INKEY$:SCREEN 1:SCREEN 0
5280 NEXT I
5290 PRINT:PRINT  "EL VALOR";:IF SR=1 THEN IF TP=1 THEN PRINT  "
     maximo "; ELSE PRINT " minimo ";
5300 PRINT "de Z es;";:RT=0
5310 FOR I=1 TO RG+SP:RT=RT+AB(I)*CB(I):NEXT I:PRINT ; RT
5320 RETURN
5330 ' EN ESTA SUBRUTINA MODIFICO LA TABLA SIMPLEX,  AL NO  TENER
     UNA BASE INICIAL FACTIBLE
5350  'MODIFICANDO  LA FUNCION OBJETIVO (QUITANDO  LAS  VARIABLES
     ARTIFICIALES)
5360 FOR I=1 TO VS%(LS,1):'RENGLONES
5370 PE=B(VS%(LS,1)+1,BV(LS;I)):IF PE <> 0 THEN GOSUB 5410
5380 FOR H3=1 TO RG+SP:FOR H4=1 TO 2*TT:VA(H3,H4)=VP(H3,H4):NEXT
     H4:NEXT H3
5390 NEXT I
5400 RETURN
5410 '  EN  ESTA  SUBRUTINA  MODIFICO  LA  FUNCION  OBJETIVO,
     MULTIPLICANDO POR MENOS  LOS COEFICIENTES DE LA FUNCION
     OBJETIVO LAS COLUMNAS DE LA TABLA Y SUMANDOLAS
5420 J=0:UNO=1
5430 FOR J=1 TO VS%(LS,1)
5440 IF B(J,BV(LS,I))=1 THEN MY=J:UNO=2
5450 NEXT J
5460 ' EN MY TENGO LOCALIZADO EL RENGLON DEL ELEMENTO BASE
5470 ' ELIMINANDO LA VARIABLE DE PENALIZACION DE ESA COLUMNA
5480 FOR W=1 TO VS%(LS,2)+1
5490 B(VS%(LS,1)+1,W)=-1*PE*B(MY,W)+B(VS%(LS,1)+1,W)
5500 NEXT W
5510 RETURN
5520  ' SUBRUTINA QUE PERMITE VOLVER A RESOLVER OTRO PROBLEMA  DE
     DESCOMPOSICION
5530 LOCATE  23,1:INPUT  "Quieres repetir  proceso  (s/n)";C$:IF
     C$<>"s" AND C$<>"S" AND C$<>"n" AND C$<>"N" THEN BEEP:BEEP:
     LOCATE 23,31:PRINT "    ";:GOTO 5530
5540 IF C$="s" OR C$="S" THEN RUN "tesis2" ELSE END
```