Centro de Investigación en Matemáticas A.C.

Descomposición para problemas de programación lineal multi - divisionales

T E S I S

Que para obtener el grado de

Maestro en Ciencias

con Orientación en

Matemáticas Aplicadas

P R E S E N T A:

Jony Alexander Rojas Rojas

Comité de Evaluación:

Dr. José Ignacio Barradas Bribiesca
(Presidente)

Dr. Armando Sánchez Nungaray
(Secretario)

Dr. Francisco Sánchez Sánchez
(Vocal y Director de Tesis)

GUANAJUATO, GTO JULIO 2012

Centro de investigación en Matemáticas A.C.

Descomposición para problemas de programación lineal multi - divisionales

T E S I S

Que para obtener el grado de

Maestro en ciencias

con Orientación en

Matemáticas Aplicadas

P R E S E N T A:

Jony Alexander Rojas Rojas

Director de Tesis:

Dr. Francisco Sánchez Sánchez

GUANAJUATO, GTO JULIO 2012

A
Gabriel R., Brittany R., Gabriela.
Mercedita R. and Henry R.

Agradecimientos

Quisiera aprovechar esta oportunidad para manifestar mi agradecimiento a las personas e instituciones sin las cuales este trabajo no hubiera sido posible:

- A Dios por darme vida, salud, paciencia y sabiduría para poder concluir esta tesis.
- A la mujer que me ha dado la vida y que a pesar de sus limitaciones ha estado conmigo y siempre a confiado en mi; gracias madre, éste también es un logro tuyo.
- A Gabriela Johana Flores por su apoyo y comprensión durante estos dos años que he estado fuera.
- A mis hermanos por darme su apoyo, comprensión y cuidar de mi madre durante todo este tiempo.
- Al director de esta tesis, el Dr. Francisco Sańchez Sánchez, por su tiempo, paciencia, tolerancia y constante apoyo a largo de este trabajo de investigación.
- A los doctores Armando Sánchez Nungaray e Ignacio Barradas por aceptar ser parte de mi comité de evaluación, y por el tiempo dedicado a revisar este trabajo.
- Al personal acádemico y administrativo del CIMAT por su constante apoyo incluso desde antes de ser parte de esta institución. Muy especialmente a Arturo Hernández Aguirre, Jannet Vega Gutiérrez, Ma. Dolores Aguilera Mújica y Eduardo Aguirre Hernández.
- Al Consejo Nacional de Ciencia y Tecnología (CONACYT) por el apoyo económico otorgado durante los dos años que duro la maestría.
- A mis compañeros de cubo, en especial a William Jóse Olvera Lopéz por sus valiosas aportaciones y correcciones hechas a este trabajo de tesis y a Saúl Díaz Infante por sus constantes consejos y ayuda en la utilización del Latex.
- A mi compañero de aventuras y hermano Oliver Antonio Juárez Romero por acompañarme y apoyarme en los momentos más duros en este hermoso país (México).
- A mi tía Rafaela Rojas por quererme como un hijo y darme su apoyo incondicional.

Índice general

.

Introducción

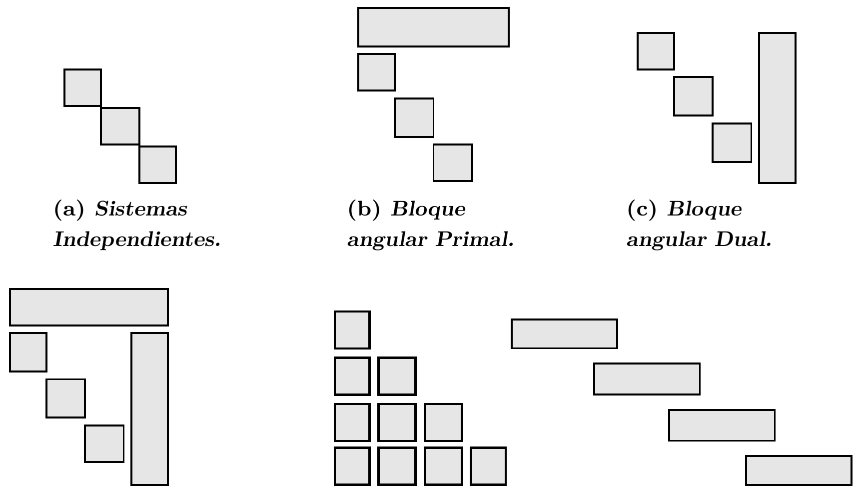
La modelación es una de las áreas más atractivas de las ciencias aplicadas. De hecho, los investigadores necesitan construir modelos para resolver problemas de la vida real. El objetivo de un modelo consiste en reproducir la realidad de la forma más sencilla posible tratando de entender cómo se comporta el mundo real y obteniendo las respuestas que pueden esperarse de determinadas acciones. Los modelos de programación lineal tratan de maximizar o minimizar una función objetivo lineal sujeta a restricciones lineales del tipo " $\leq$ ", " $\geq$ " y " $=$ ". Por ejemplo, considere la siguiente situación: la línea de producción de un taller de esculturas de piedras tiene tres artículos: figuras, figuritas y estatuas. Estos artículos necesitan ciertas horas de cortado y cincelado; además la capacidad del taller está limitada en cada una de las actividades. Estos datos se muestran en la siguiente tabla, así como el beneficio (en pesos) que proporciona al taller cada escultura:

	Tipo de producto			
Operación	Figuras	Figuritas	Estatuas	Capacidad
Cortado	30	5	60	300
Cincelado	20	8	30	180
Beneficio	280	40	510	

El taller desea un plan para seleccionar la combinación de esculturas para la siguiente temporada de ventas. El dueño del taller supone que puede vender un número ilimitado de sus esculturas.
Este problema se puede plantear de la siguiente manera:

$$\begin{array}{rl} max & 280x_1 + 40x_2 + 510x_3 \\ \text{s.a} & \\ & 30x_1 + 5x_2 + 60x_3 \leq \quad 300 \\ & 20x_1 + 8x_2 + 30x_3 \leq \quad 180 \\ & x_1, x_2, x_3 \in \mathbb{Z}^+. \end{array}$$

En los modelos de programación lineal con frecuencia aparecen problemas de gran tamaño con una estructura especial, es decir, problemas que tienen un patrón en la matriz de coeficientes de las restricciones. Los patrones más importantes están representados en la Figura 1:



(a) ***Sistemas Independientes.*** **(b)** ***Bloque angular Primal.*** **(c)** ***Bloque angular Dual.***

(d) ***Angular bordeadas.*** **(e)** ***Bloque triangular.*** **(f)** ***Escalera.***

Figura 1: Algunos patrones en los problemas de programación lineal.

Como una extensión de la programación lineal para problemas con estructuras especiales en la matriz de coeficientes de las restricciones aparecieron las técnicas de descomposición [Benders, Dantzig: 60, Dantzig: 63] también denominadas de optimización matématica a gran escala. Las técnicas de descomposición resuelven problemas de gran tamaño con una estructura especial, que desde un punto de vista teórico y computacional, se aprovechan de la solución iterativa de otros problemas de menor tamaño que se derivan del problema original. Las técnicas de descomposición se aplican a un problema cuya estructura específica permite identificar partes del mismo que son fácilmente resolubles de modo individual. Éstas constituyen una forma flexible y elegante de resolver problemas de programación lineal.

En el presente trabajo se estudiará un problema de programación lineal con un conjunto de subsistemas interrelacionados llamado problema multi- divisional. La estructura de la matriz de coeficientes de las restricciones se muestra en la Figura 2:

Figura 2: Patrón de un problema de programación lineal multi- divisional

El primero de los trabajos dentro de la optimización a gran escala donde resuelve este tipo de problema de programación lineal se debe a George B. Dantzig en 1960 y a partir de entonces, se han desarrollado algoritmos que permiten resolver de forma descentralizada los problemas multi- divisionales y otros tipos de problemas con estructura especial. En este trabajo estudiamos formas alternativas de resolver estos tipos de problemas, y con ello, estamos realizando la propuesta de dos métodos de solución para casos específicos del problema. Nuestros métodos están fuertemente basados en teoría de dualidad, ampliamente utilizada en programación lineal, y proporcionan un enfoque diferente en el tratamiento de estos problemas. La presente tesis se divide en tres capítulos, de los que a continuación se presenta un breve resumen:

- El primer capítulo contiene los conceptos más importantes de la programación lineal que serán útiles en este trabajo. Aquí se abordan los conceptos de base, variables básicas y soluciones básicas. La Sección 1.3 está dedicada a la teoría de dualidad donde se estudia el problema dual (problema de programación lineal de mínimo asociado a un problema de programación lineal de máximo), los teoremas que relacionan al problema primal (problema de programación lineal de máximo) con su dual y la interpretación de las variables duales; esta última jugará un papel importante en el presente trabajo.

- El segundo capítulo se tratan dos algoritmos de descomposición, uno para problemas multi-divisional, debido a Dantzig y el otro se aplica a problemas bietapa, el cual fue propuesto por Benders en 1962. El algoritmo de Dantzig descompone un problema multi- divisional en un problema maestro y varios subproblemas. El maestro envía a los subproblemas variables duales (precios sombra) y los subproblemas maximizan sus funciones objetivos parametrizadas por estas variables duales. Las soluciones de los subproblemas son la nueva propuesta que se hace al problema maestro. Éste utiliza todas las propuestas hechas hasta el momento para conseguir una nueva solución óptima, de la que obtiene los valores (variables duales) de las restricciones.

  Por otro lado, el algoritmo de Benders descompone un problema bietapa en un problema maestro y un subproblema. El problema maestro envía las variables de la primera etapa y el subproblema maximiza su función objetivo como función de las decisiones de la primera etapa. Las variables duales (precios sombra) del subproblema se envían al problema maestro para generar un corte que elimine la propuesta anterior, y así, dar un nuevo valor de la variable de la primera etapa.

- El tercer capítulo contiene los resultados originales que obtuvimos en la presente tesis. Primeramente presentamos un algoritmo para resolver problemas de programación lineal con una estructura de bloques y una restricción de acople, también llamada restricción del recurso común, y después presentamos otro para resolver un problema bi- divisional. Nuestro propósito es encontrar la solución óptima del problema original a través de las soluciones de los subproblemas derivados de éste. El primer algoritmo inicia distribuyendo el recurso común entre los subproblemas y resolviendo éstos para encontrar la utilidad que obtiene un subproblema por una unidad adicional del recurso común (precio sombra). Luego, transfiere un monto del problema con menor precio sombra al problema con mayor precio sombra, lo cual causa un incremento en la función objetivo del problema original.

El segundo algoritmo inicia resolviendo el problema maestro bi- divisional relajado. La solución del problema maestro bi- divisional relajado se pasa al subproblema; éste es resuelto y envía al problema maestro bi- divisional relajado el valor de su función objetivo y el vector de precios sombra asociado al recurso no utilizado por la división 1, con lo cual se añade un corte al problema maestro, y se continúa de esta manera hasta que se verifica el criterio de paro.

En la Sección 1.2 demostramos que cuando ya no es posible mejorar vía transferencia del recurso común entre los subproblemas, el óptimo de un problema con estructura de bloques y una restricción de acople se ha alcanzado y que cuando se satisface el criterio de paro del segundo algoritmo, el óptimo del problema bi- divisional se ha logrado.

El objetivo general de la presente tesis es colaborar al desarrollo de la programación lineal a gran escala, en concreto, a la solución de problemas lineales con estructura de bloques y restricciones acopladas. Los objetivos específicos son:

- Estudiar el algoritmo de descomposición de Dantzig y el algoritmo de Benders, con el fin de proporcionar nuevos algoritmos para resolver problemas con estructura de bloques y restricciones acopladas.
- Diseñar un algoritmo para resolver un problema con estructura de bloques y una restricción de acople vía transferencia.
- Adaptar el algoritmo de Benders para solucionar un problema bi- divisional.

Capítulo 1

Preliminares y definiciones

La programación lineal es una rama de la programación matemática que trata exclusivamente con funciones objetivos y restricciones lineales. Esta rama de la programación matemática se utiliza en campos como la ingeniería, la economía, la gestión y muchas otras áreas de la ciencia para modelar procesos de toma de decisiones. Cuando se trata de resolver un problema de este tipo, la primera etapa consiste en identificar las variables del problema concreto. Normalmente las variables son de carácter cuantitativo y se buscan los valores que optimizan el objetivo. La segunda etapa consiste en determinar el conjunto de decisiones admisibles; esto conduce a un conjunto de restricciones que se determinan teniendo presente los datos del problema en cuestión. En la tercera etapa, se calcula el costo/beneficio asociado a cada decisión admisible; esto supone determinar una función objetivo que asigna, a cada conjunto posible de valores para las variables que determinan una decisión, un valor de costo/beneficio. El conjunto de todos estos elementos define un problema de programación lineal.

En este capítulo se introduce la programación lineal por medio de un ejemplo, a través del cual se abordan conceptos básicos de la programación lineal. Hacemos especial énfasis en la teoría de dualidad, por las relaciones que existen entre el problema primal y el problema dual. La importancia de tratar estos temas radica en preparar el campo para estudiar la descomposición en programación lineal por ser una técnica para resolver problemas de gran tamaño y el objeto de esta tesis.

1.1. Planteamiento del problema

Supóngase la siguiente situación: Una empresa internacional produce y vende dos tipos de productos: A y B. En la elaboración de ambos tipos de productos hay que destacar dos procesos, el de ensamblado final y el de empaquetado, procesos p_1 y p_2. Esta empresa dispone mensualmente de 2000 horas dedicadas al proceso de ensamblado y 1000 horas dedicadas al proceso de empaquetado, además se sabe que los tiempos requeridos para realizar dichas operaciones para cada uno de los tipos de productos son los que se muestran en la siguiente tabla:

Procesos	Horas requeridas A	B	Horas mensuales disponibles
P_1	6	4	2000
P_2	2	4	1000

El beneficio neto obtenido tras la venta del producto A es 400 u.m. y tras la venta de una unidad del producto B es de 600 u.m. Si definimos las variables x_1 y x_2 como el número de productos de tipo A y B respectivamente, el problema se puede plantear de la siguiente manera:

$$\begin{aligned} max \quad z = & \quad 400x_1 + 600x_2 \\ s.a. & \end{aligned} \tag{1.1}$$

$$\begin{aligned} 6x_1 + 4x_2 &\leq 2000 \\ 2x_1 + 4x_2 &\leq 1000 \\ x_1, x_2 &\geq 0. \end{aligned}$$

El problema también puede escribirse como

$$\begin{aligned} max \quad z = & \quad c^T x \\ s.a. & \end{aligned} \tag{1.2}$$

$$\begin{aligned} Ax &\leq b \\ x &\geq 0. \end{aligned}$$

Definición 1.1. *A un punto $x \in \mathrm{R}^n$ que satisface las restricciones del problema 1.2 se denomina solución factible y a la función $c^T x$ función objetivo. El conjunto de todas esas soluciones es la región de factibilidad.*

Definición 1.2. *Un punto factible x^* tal que $c^T x \leq c^T x^*$, para cualquier otro punto factible x, se denomina una solución óptima.*

El objetivo de un problema de programación lineal es encontrar la solución factible que maximice la función objetivo.

Si por cada restricción i se agrega una variable $h_i = b_i - \sum a_{ij} x_j$ que mida la diferencia entre el lado izquierdo y derecho de la desigualdad (a estas variables se les conoce como de *holgura*), entonces el problema se escribe:

$$\begin{aligned} max \quad z = & \quad 400x_1 + 600x_2 \\ s.a. & \end{aligned}$$

$$\begin{aligned} 6x_1 + 4x_2 + h_1 &= 2000 \\ 2x_1 + 4x_2 + h_2 &= 1000 \\ x_1, x_2 &\geq 0 \end{aligned}$$

en forma matricial

$$\begin{aligned} max \quad z = \quad & c^T x \\ s.a. \quad & \end{aligned}$$

$$\begin{aligned} Ax + h = b \\ x \geq 0. \end{aligned}$$

o de la forma:

$$\begin{aligned} max \quad z = \quad & c^T x \\ s.a. \quad & \end{aligned} \tag{1.3}$$

$$\begin{aligned} \mathtt{A}x = b \\ x \geq 0 \end{aligned}$$

donde $c^T = (c^T, 0)$, $x^T = (x^T, h^T)$ y $\mathtt{A} = [A, I]$. Cuando el problema se presenta en esta forma, se dice que está en la *forma estándar*.

1.2. Soluciones básicas

Considérese un problema de programación lineal en forma estándar matricial 1.3, donde se supondrá, sin pérdida de generalidad, que el rango de la matriz $\mathcal{A}$ de dimensión $m \times n$ es m (recuérdese que $m \leq n$), y que el sistema lineal $\mathcal{A}x = b$ tiene solución. En cualquier otro caso, el problema lineal es equivalente a otro con menos restricciones, o no tiene solución factible, respectivamente.

Definición 1.3. *A un subconjunto $B \subseteq \{1, \ldots, n\}$ tal que $|B| = m$, se le denomina base.*

Definición 1.4. *A una solución factible se le denomina solución básica factible si $x_j = 0$ para toda $j \notin B$.*

Observación 1.1. *Dada una base B se tienen los siguientes vectores:*

1. *$x_B = (x_j)_{j \in B}$, que se denominan variables básicas y el resto se denominan variables no básicas.*

2. *$c_B = (c_j)_{j \in B}$.*

3. *$\mathcal{A}_B = (\mathcal{A}_j)_{j \in B}$, donde $\mathcal{A}_j$ denota la j- ésima columna de A. Esta matriz se conoce como matriz básica.*

El siguiente teorema muestra por qué las soluciones básicas factibles son importantes:

Teorema 1. *Si un problema de programación lineal tiene una solución óptima, entonces tiene al menos una solución básica óptima.*

1.3. Dualidad

Esta sección trata la dualidad en programación lineal, su importancia y significado. La teoría de la dualidad nos va a permitir, entre otras cosas, relacionar cada problema de programación lineal con otro denominado problema dual y obtener relaciones sobre el tipo de soluciones de ambos problemas. También nos va a proporcionar herramientas alternativas a las ya conocidas para comprobar la optimalidad de soluciones, así como condiciones que puedan utilizarse para el desarrollo de nuevos algoritmos de descomposición en programación lineal.

A continuación plantearemos una nueva situación en la que, de una forma natural, surge el planteamiento del problema dual asociado al problema primal 1.1 planteado en la sección anterior.

Nuestro propósito va a ser determinar los precios a los cuales esta empresa debería valorar sus recursos (horas de trabajo de los dos procesos) de tal manera que pueda determinar el mínimo valor total al cual estaría dispuesta a arrendar o vender los recursos.

Sean u_1 y u_2, la renta percibida por hora de los procesos p_1 y p_2 respectivamente. La renta total obtenida será $2000u_1 + 1000u_2$. Se desea como objetivo encontrar el mínimo valor de $2000u_1 + 1000u_2$ de modo que la empresa pueda, de manera inteligente, analizar algunas propuestas de alquiler o compra de todos los recursos como un paquete total.

Se consideran las condiciones siguientes, los precios pagados por el alquiler serán no negativos, es decir, $u_1, u_2 \geq 0$. Además los precios u_1 y u_2 deben ser competitivos con las alternativas disponibles, es decir, el beneficio que la empresa debe tener por la venta de los recursos necesarios para elaborar uno de sus productos (A o B) al menos debe ser igual al beneficio que obtendría al utilizar dichos recursos en la elaboración del producto A o B, es decir, para el producto A tendremos $6u_1 + 2u_2 \geq 400$ y para el producto B queda $4u_1 + 4u_2 \geq 600$. Con esto la empresa garantiza la obtención de precios con los que al menos iguala el beneficio obtenido al producir élla misma los productos A y B. El problema planteado queda:

$$\begin{array}{rcl} min \quad Z & = & 2000u_1 + 1000u_2 \\ \text{s.a.} & & \end{array}$$

$$\begin{aligned} 6u_1 + 2u_2 &\geq 400 \\ 4u_1 + 4u_2 &\geq 600 \\ u_1, u_2 &\geq 0. \end{aligned}$$

Este nuevo problema es el problema dual del problema planteado originalmente 1.1.

1.3.1. El problema dual

Como ya hemos comentado, asociado a un problema de programación lineal aparece otro problema de programación lineal denominado *problema dual* (asociado al primero), en general nos referiremos al primero como *problema primal* y lo denotaremos mediante [P] y al segundo como problema dual y lo denotaremos con [D].

Definición 1.5. *Dado el problema de programación lineal*

$$\begin{array}{lcl} max \quad z & = & c^T x \\ s.a. & & \\ & & Ax \leq b \\ & & x \geq 0 \end{array} \tag{P}$$

su problema dual es

$$\begin{array}{lcl} min \quad G & = & b^T u \\ s.a. & & \\ & & A^T u \geq b \\ & & u \geq 0 \end{array} \tag{D}$$

donde $u = (u_1, \ldots, u_m)$ *se denominan variables duales o precios sombra.*

Obsérvese que los mismos elementos (la matriz A, y los vectores b y c) configuran ambos problemas. El problema primal no se ha escrito en forma estándar, sino en una forma que nos permita apreciar la simetría entre ambos problemas, y mostrar así que el dual del dual es el primal. Además cada restricción del problema primal tiene asociada una variable del problema dual; los coeficientes de la función objetivo del problema primal son los términos independientes de las restricciones del problema dual y viceversa; y la matriz de restricciones del problema dual es la transpuesta de la matriz de restricciones del problema primal; y por último el problema primal es de maximización y el dual de minimización.

1.3.2. Obtención del problema dual

Un problema de programación lineal de la forma (P) tiene asociado un problema dual que puede formularse según las reglas siguientes:

Regla 1: Una restricción de igualdad en el primal (dual) hace que la correspondiente variable dual (primal) no esté restringida en signo.

Regla 2: Una restricción de desigualdad $\leq$ ($\geq$) en el primal (dual) da lugar a una variable dual (primal) no negativa.

Regla 3: Una restricción de desigualdad $\geq$ ($\leq$) en el primal (dual) da lugar a una variable dual (primal) no positiva.

Regla 4: Una variable no negativa primal (dual) da lugar a una restricción de desigualdad $\geq$ ($\leq$) en el problema dual (primal).

Regla 5: Una variable no positiva primal (dual) da lugar a una restricción de desigualdad $\leq$ ($\geq$) en el problema dual (primal).

Regla 6: Una variable no restringida en signo del problema primal (dual) da lugar a una restricción de igualdad en el dual (primal).

Ejemplo 1.1. *Considérese el problema de la sección 1.1:*

$$\begin{aligned} max \quad z = \quad & 400x_1 + 600x_2 \\ s.a. \quad & \end{aligned}$$

$$\begin{aligned} 6x_1 + 4x_2 &\leq 2000 \\ 2x_1 + 4x_2 &\leq 1000 \\ x_1, x_2 &\geq 0. \end{aligned}$$

Como se sabe, el dual de este problema es

$$\begin{aligned} min \quad z \quad = \quad & 2000u_1 + 1000u_2 \\ \text{s.a.} \quad & \end{aligned}$$

$$\begin{aligned} 6u_1 + 2u_2 &\geq 400 \\ 4u_1 + 4u_2 &\geq 600 \\ u_1, u_2 &\geq 0. \end{aligned}$$

Para obtenerlo se aplican las reglas anteriores de la siguiente forma:

Regla 2: Puesto que las restricciones del problema primal son de desigualdad $\leq$, las variables duales u_1 y u_2 son no negativas .

Regla 3: Puesto que las variables primales son no negativas, las dos restricciones duales son de desigualdad de la forma $\geq$.

1.3.3. Teoremas relativos a problemas primales y sus duales asociados

Mientras no se indique lo contrario vamos a trabajar con los siguientes problemas primal y dual.

$$\begin{array}{llr} & max & z = c^T x \\ (P) & \text{s.a.} & \\ & & Ax \leq b \\ & & x \geq 0 \end{array} \qquad \begin{array}{llr} & min & G = b^T u \\ (D) & \text{s.a.} & \\ & & A^T u \geq c^T \\ & & u \geq 0. \end{array}$$

La importancia del problema dual se establece en los siguientes teoremas.

Teorema 2. *Si $\bar{x}$ y $\bar{u}$ son soluciones factibles de un par de problemas primal y dual (P) y (D), respectivamente, entonces se verifica*

$$z(\bar{x}) = c^T \bar{x} \leq b^T \bar{u} = G(\bar{u}).$$

Es decir, para cualquier par de soluciones factibles de un problema primal y su dual el valor de la función objetivo del problema de máximo es menor o igual que el valor de la función objetivo del problema de mínimo.

Teorema 3. *Si existen soluciones factibles para los problemas primal y dual (P) y (D) tales que los valores correspondientes de las funciones objetivo coinciden, entonces dichas soluciones factibles son óptimas para sus respectivos problemas.*

Teorema 4. *Sea x^* una solución básica óptima de (P) y sea $\mathcal{A}_B$ la matriz asociada a su base B, entonces $\pi = c_B \mathcal{A}_B^{-1}$ es una solución óptima del problema dual (D).*

Observación 1.2. *De la formula $\pi = c_B \mathcal{A}_B^{-1}$ se deduce que el valor de las variables duales o precios sombra sólo de penden de la base óptima B.*

Teorema 5. *Dados $\bar{x}$ y $\bar{u}$ soluciones factibles de los problemas (P) y (D), respectivamente, entonces $\bar{x}$ y $\bar{u}$ son soluciones óptimas para sus problemas respectivos si y sólo si se verifica*

$$\begin{aligned} \bar{x}_j \bar{v}_j &= 0 \quad \forall j = 1, \ldots, n \\ \bar{u}_i \bar{h}_i &= 0 \quad \forall i = 1, \ldots, m \end{aligned}$$

donde $\bar{h}$ son las holguras de las restricciones del problema primal y $\bar{v}$ las holguras correspondientes al problema dual.

A partir de la observación 1.1 y los teoremas 4 y 5, se sigue que un problema de programación lineal queda resuelto si se cuenta con una base óptima B, ya que por medio de $\mathcal{A}_B^{-1}$ se pueden determinar los tres elementos principales de la solución del problema:

- variables de decisión: $x_B = \mathcal{A}_B^{-1} b$
- valor de la función objetivo: $z = c_B^T x_B$
- variables duales: $y^T = c_B^T \mathcal{A}_B^{-1}$.

Supongamos que se han resuelto los problemas (P) y (D). Si al modificar el lado derecho de (P) la base óptima sigue siendo la misma, su solución se puede calcular mediante las relaciones anteriores. Esto crea la necesidad de determinar la región donde se preserva la base. Se dirá que no hay cambio de base (NCB), si los dos problemas (el original y el modificado) admiten la misma base óptima. Uno de los cambios más comunes es: $\hat{b} = b + \theta d$ y es el único que se considera en esta tesis.

Teorema 6. *NCB para $\hat{b} = b + \lambda d \Leftrightarrow \lambda \mathcal{A}_B^{-1} d \geq -x_B$.*

Obsérvese que el máximo valor que puede tomar λ cuando $d = e_i$, se puede interpretar como la cantidad máxima que pueden aumentarse los recursos de la restricción i sin que haya cambio de base. Esta idea será ampliamente utilizada en el siguiente capítulo.

1.3.4. Interpretación de las variables duales

En esta sección se establecen las ideas principales que se desarrollarán en esta tesis, por tal motivo le daremos un tratamiento diferente al que tradicionalmente se le da. Durante esta sección el problema de referencia será:

$$
(P) \quad \begin{array}{ll} max & z = c^T x \\ \text{s.a.} & \\ & Ax \leq b \\ & x \geq 0 \end{array}
$$

Teorema 7. *Si el problema (P) es resuelto dos veces con lados derechos b y $\hat{b} = b + e_i$ obteniéndose como soluciones a z_1^* y z_2^* respectivamente, con la misma base óptima, entonces $z_2^* = z_1^* + u_i$.*

Demostración. Sea $\mathcal{A}_B$ la matriz básica asociada a la base B, entonces

$$
z_2^* = c_B^T \mathcal{A}_B^{-1}(b + e_i) = u^T b + u^T e_i = z_1^* + u_i. \tag{1.4}
$$

lo cual prueba el teorema.

De la expresión 1.4 se desprende que el valor óptimo de la variable dual u_i^* asociada a la restricción i del problema primal representa la cantidad en la que se modifica el valor de la función objetivo de (P) por cada unidad del recurso i (siempre que la modificación conserve la base óptima). Teniendo en cuenta esto, si nos situamos, por ejemplo, en un problema primal de maximización con restricciones de tipo $\leq$, lo resolvemos y calculamos la solución del problema dual, entonces los valores óptimos de las variables duales (que son ≥ 0) representan la variación unitaria de la función objetivo del primal por cada unidad adicional del recurso correspondiente que pudiéramos conseguir. Esto nos permite también interpretar estos valores como la mayor cantidad que estaríamos dispuestos a pagar por una unidad adicional de recurso.

El siguiente ejemplo pone de manifiesto las ideas anteriores.

Ejemplo 1.2. *Supongamos que hemos resuelto el problema*

$$
\begin{array}{ll} max \quad z = & 400x_1 + 600x_2 \\ s.a. & \end{array}
$$

$$
\begin{array}{c} 6x_1 + 4x_2 \leq 2000 \\ 2x_1 + 4x_2 \leq 1000 \\ x_1, x_2 \geq 0. \end{array}
$$

obteniendo un valor de la función objetivo de 175000 *y con* $\pi = c_B \mathcal{A}_B^{-1} = (25, 125)$

El valor de 25 nos dice que si aumentamos o disminuimos el lado derecho de la primera restricción en una unidad, la función objetivo aumenta o disminuye en 25 unidades (siempre que se mantenga la base óptima). Así, si 2000 se disminuye hasta 1400 y resolvemos el problema con $\hat{b}_1 = 1400$ el valor de la función objetivo es de 160000, es decir, se disminuyó en el valor de la variable dual primera 25 por el número de unidades disminuidas 600, tal como habíamos previsto.

Para finalizar esta sección, recordamos que cuando usamos LINDO o cualquier software de similares características para resolver un problema de programación lineal, no sólo obtenemos la solución del mismo (en caso de que exista), sino también los precios sombra o valores de las variables duales.

.

Capítulo 2

Descomposición en programación lineal

El tamaño de un problema de programación lineal puede ser muy grande: podemos encontrar problemas de interes práctico con varios cientos de miles de ecuaciones o variables. Existen técnicas de descomposición que permiten que estos tipos de problemas puedan resolverse de forma descentralizada o distribuida.

Para que una técnica de descomposición sea útil, el problema debe tener una estructura apropiada. El problema de programación lineal a tratar en esta sección tiene una estructura de bloques con un conjunto de restricciones acopladas, lo cual permite descomponerlo en una serie de problemas relativamente pequeños. A los problemas de programación lineal con esta estructura se les denomina *problema multi - dividisional.*

Ejemplo 2.1. *Considérese el problema*

$$\begin{array}{llcll}
max & 9x_1 + 8x_2 & + & 7x_3 + 6x_4 & \\
s.a & & & & \\
 & 8x_1 + 6x_2 & + & 7x_3 + 5x_4 & = 80 \qquad (1) \\
 & 3x_1 + x_2 & & & = 10 \\
 & 2x_1 + x_2 & & & = 10 \\
 & & & 3x_3 + 2x_4 & = 8 \\
 & & & x_3 + x_4 & = 4 \\
 & & & x_1, x_2, x_3, x_4 & \geq 0
\end{array}$$

Si el problema no tuviera la restricción (1), el problema anterior se puede resolver, mediante la solución de los siguientes dos subproblemas:

Subproblema 1

$$\begin{array}{ll} max & 9x_1 + 8x_2 \\ \text{s.a} & \\ & 3x_1 + x_2 = 10 \\ & 2x_1 + x_2 = 10 \\ & x_1, x_2 \geq 0 \end{array}$$

Subproblema 2

$$\begin{array}{ll} max & 7x_3 + 6x_4 \\ \text{s.a} & \\ & 3x_3 + 2x_4 = 8 \\ & x_3 + x_4 = 4 \\ & x_3, x_4 \geq 0 \end{array}$$

Pero, dado que la ecuación (1) del problema original

$$8x_1 + 6x_2 + 7x_3 + 5x_4 = 80$$

involucra todas las variables, no se puede aplicar una solución por bloques. A las restricciones del tipo (1) se les denomina restricciones de *acople* o de recurso común.

Los procedimientos de descomposición son técnicas computacionales que consideran indirectamente las restricciones de acople. En estos procedimientos, en lugar de resolver el problema original con restricciones de acople, se resuelven una serie de problemas, llamados *subproblemas*, iterativamente.

A continuación formalizaremos la idea presentada anteriormente: Consideremos el problema de programación lineal

$$\begin{array}{lll} max & \sum_{j=1}^{n} c_j x_j & \\ \text{s.a} & & \\ & \sum_{j=1}^{n} b_{ij} x_j = b_i; \quad i = 1, \ldots, q & (2) \\ & \sum_{j=1}^{n} a_{ij} x_j = d_i; \quad i = 1, \ldots, m & \\ & x_j \geq 0; \quad j = 1, \ldots, n & \end{array} \tag{2.1}$$

donde las restricciones (2) tienen una estructura descomponible en r bloques, cada uno de tamaño n_k $(k = 1, 2, \ldots, r)$, es decir, las restricciones (2), pueden escribirse como

$$\sum_{j=n_k+1}^{n_k} b_{ij} x_j = b_i \qquad i = q_{k-1} + 1, \cdots, q_k; \qquad k = 1, 2, \ldots, n.$$

Nótese que $n_0 = q_0 = 0, q_r = q$, $n_r = n$. Con ello, el problema 2.1 puede escribirse como

$$\begin{aligned}
max \quad & \sum_{k=1}^{r} c_k^T x_k \\
\text{s.a} \quad & \\
& B_k x_k = b; \qquad k = 1, \dots, r \\
& \textstyle\sum_{k=1}^{r} A_k x_k = d \qquad\qquad (3) \\
& x_k \geq 0; \qquad k = 1, \dots, r.
\end{aligned}$$

En esta sección vamos a suponer que el conjunto $S_j = \{x_j : B_j x_j = b, x_j \geq 0\}$ es no vacío, convexo y acotado. Dado que las restricciones (3) contienen todas las variables de decisión, éstas son las restricciones de acople.

Si ignoramos las restricciones de acople, el problema original se convierte en

$$\begin{aligned}
max \quad & \sum_{k=1}^{r} c_k^T x_k \\
\text{s.a} \quad & \\
& B_k x_k = b; \quad k = 1, \dots, r \\
& x_k \geq 0; \quad k = 1, \dots, r.
\end{aligned}$$

Este problema es llamado *la versión relajada del problema*. El *k-ésimo subproblema descompuesto* es

$$\begin{aligned}
max \quad & c_k^T x_k \\
\text{s.a} \quad & \\
& B_k x_k = b \\
& x_k \geq 0
\end{aligned}$$

y puede escribirse como

$$\begin{aligned}
max \quad & \sum_{j=n_{k-1}+1}^{n_k} c_j x_j \\
\text{s.a} \quad & \\
& \sum_{j=n_{k-1}+1}^{n_k} b_{ij} x_j = b_i \\
& x_j \geq 0 \qquad j = n_{k-1}+1, \dots, n_k.
\end{aligned}$$

El siguiente ejemplo muestra un problema de programación lineal descomponible con dos restricciones de acople.

Ejemplo 2.2. *El problema*

$$\begin{array}{lcccc}
max & 6x_1 + 5x_2 & + & 3x_3 + 4x_4 & & \\
s.a & & & & & \\
 & x_1 + x_2 & & & \leq & 4 \\
 & 3x_1 + 2x_2 & & & \leq & 12 \\
 & & & x_3 + 2x_4 & \leq & 6 \\
 & & & 2x_3 + x_4 & \leq & 3 \\
 & x_1 + x_2 & + & x_3 + x_4 & \leq & 7 \\
 & 2x_1 + x_2 & + & x_3 + 2x_4 & \leq & 17 \\
 & & & x_1, x_2, x_3, x_4 & \geq & 0
\end{array}$$

tiene una estructura descomponible en dos bloques, donde las dos últimas desigualdades

$$x_1 + x_2 + x_3 + x_4 \leq 7$$

$$2x_1 + x_2 + x_3 + 2x_4 \leq 17$$

son las restricciones de acople.

2.1. Descomposición de Dantzig y Wolfe

En esta sección se describirá el algoritmo de Dantzig y Wolfe, el cual resuelve problemas de programación lineal de gran tamaño con restricciones acopladas y estructura de bloques.

Supongamos que cada uno de los subproblemas es resuelto p veces con diferentes y arbitrarias funciones objetivos, y supongamos que las p soluciones básicas factibles de los subproblemas son

$$\begin{gathered} x_1^{(1)}, x_2^{(1)}, \ldots, x_n^{(1)} \\ \vdots \\ x_1^{(p)}, x_2^{(p)}, \ldots, x_n^{(p)} \end{gathered}$$

donde $x_j^{(s)}$ es la j- ésima componente de la s- ésima solución, donde todas las variables de todos subproblemas son consideradas, y los correspondientes p valores de las funciones objetivos son

$$z^{(1)}, z^{(2)}, \ldots, z^{(p)}$$

donde $z^{(s)}$ es el valor de la función objetivo en la s-ésima solución. Los valores de las m restricciones de acople para las p soluciones son

$$\begin{gathered} r_1^{(1)}, r_2^{(1)}, \ldots, r_m^{(1)} \\ \vdots \\ r_1^{(p)}, r_2^{(p)}, \ldots, r_m^{(p)} \end{gathered}$$

donde $r_i^{(s)}$ es el valor de la i- ésima restricción de acople de la s- ésima solución , es decir,

$$r_i^{(s)} = \sum_{j=1}^{n} a_{ij} x_j^{(s)}. \tag{2.2}$$

Las p soluciones básicas factibles pueden ser usadas para producir una solución factible del problema original, resolviendo el siguiente problema, llamado *problema maestro*

$$\begin{array}{ll} max & \sum_{s=1}^{p} z^{(s)} u_s \\ \text{s.a} & \\ & \sum_{s=1}^{p} r_i^{(s)} u_s = d_i \quad : \quad \lambda_i \quad i = 1, \ldots, m \\ & \sum_{s=1}^{p} u_s = 1 \qquad : \quad \sigma \\ & u_s \geq 0; \qquad\qquad s = 1, \ldots, p, \end{array} \tag{2.3}$$

donde las correspondientes variables duales λ_i y σ están indicadas. El dual del problema 2.3 es

$$\begin{array}{ll} min & \sum_{i=1}^{m} \lambda_i d_i + \sigma \\ \text{s.a} & \\ & \sum_{i=1}^{m} r_i^{(s)} \lambda_i + \sigma \geq z^{(s)} \quad s = 1, \ldots, p. \end{array} \tag{2.4}$$

El conjunto $\{u_1, \ldots, u_p\}$ es óptimo en el problema 2.3 si (λ_i, σ) es factible en el problema 2.4, pero (λ_i, σ) es factible si

$$\sigma \geq z^{(s)} - \sum_{i=1}^{m} r_i^{(s)} \lambda_i = \sum_{j=1}^{n} (c_j - \sum_{i=1}^{m} a_{ij} \lambda_i) x_j^{(s)};$$

entonces, (λ_i, σ) es factible si

$$\sigma \geq \text{máx} \left\{ \sum_{j=1}^{n} \left(c_j - \sum_{i=1}^{m} a_{ij} \lambda_i \right) x_j^{(s)} \right\}$$

donde $x_j^{(s)}$ satisface $B_j x_j^{(s)} = b_j \quad j = 1, \ldots, n$ y $s = 1, \ldots, p$. Luego para encontrar x_j debemos resolver el siguiente problema de programación lineal

$$\begin{array}{ll} max & \sum_{j=n_{k-1}+1}^{n_k} \left(c_j - \sum_{i=1}^{m} a_{ij} \lambda_i \right) x_j \\ \text{s.a} & \\ & \sum_{n_{k-1}+1}^{n_k} b_{ij} x_j = b_i; \qquad i = q_{k-1} + 1, \ldots, n_k \\ & x_j \geq 0. \end{array} \tag{2.5}$$

Obsérvese que el problema 2.5 es similar a los subproblemas relajados con la diferencia de que la función objetivo está modificada por la función

$$\sum_{i=1}^{m} a_{ij} \lambda_i x_j$$

que se puede interpretar como un costo total por la utilización de $\sum_{i=1}^{m} a_{ij}x_j$ recursos.

2.1.1. Algoritmo

En esta sección presentamos los pasos del algoritmo de Dantzig y Wolfe y un ejemplo de su aplicación.

Entrada. Un problema de programación lineal con restricciones de acople.

Salida. La solución del problema de programación lineal obtenido después de usar el algoritmo de descomposición de Dantzig y Wolfe.

Paso 0: Inicialización. Inicie la iteración, $\nu = 1$. Obtener $p^{(\nu)}$ soluciones distintas al resolver $p^{(\nu)}$ veces $(l = 1, \cdots, p^{(\nu)})$ cada uno de los siguientes r subproblemas $(k = 1, \ldots, r)$

$$
\begin{array}{lll}
max & \displaystyle\sum_{j=n_{k-1}+1}^{n_k} \hat{c}_j^{(l)} x_j & \\
\text{s.a} & & \\
& \sum_{n_{k-1}+1}^{n_k} b_{ij}x_j = b_i; & i = q_{k-1}+1, \ldots, n_k \\
& x_j \geq 0 & j = n_{k-1}+1, \ldots, n_k,
\end{array}
$$

donde $\hat{c}_j^{(l)}$ $(j = n_{k-1}+1, \cdots, n_k; k = 1, \ldots, r; l = 1, \ldots, p^{(\nu)})$ son coeficientes de costos arbitrarios para obtener las $p^{(\nu)}$ soluciones iniciales de los r subproblemas.

Paso 1: Solución del problema maestro. Resolver el problema

$$
\begin{array}{llll}
max & \displaystyle\sum_{s=1}^{p^{(\nu)}} z^{(s)} u_s & & \\
\text{s.a} & & & \\
& \sum_{s=1}^{p^{(\nu)}} r_i^{(s)} u_s = d_i & : \quad \lambda_i & i = 1, \ldots, m \\
& \sum_{s=1}^{p^{(\nu)}} u_s = 1 & : \quad \sigma & \\
& u_s \geq 0; & & s = 1, \ldots, p^{(\nu)},
\end{array}
$$

para obtener la solución $u_1^{(\nu)}, \ldots, u_{p^{(\nu)}}^{(\nu)}$, y los valores de las variables duales $\lambda_1^{(\nu)}, \ldots, \lambda_m^{(\nu)}$ y $\sigma^{(\nu)}$.

Paso 2: Solución de los subproblemas. Generar una solución del problema relajado al resolver los siguientes r subproblemas $(k = 1, \ldots, r)$

$$
\begin{array}{lll}
max & \displaystyle\sum_{j=n_{k-1}+1}^{n_k} \left(c_j - \sum_{i=1}^{m} a_{ij}\lambda_i^{\nu} \right) x_j & \\
\text{s.a} & & \\
& \sum_{n_{k-1}+1}^{n_k} b_{ij}x_j = b_i; & i = q_{k-1}+1, \ldots, n_k \\
& x_j \geq 0 & j = n_{k-1}+1 \ldots, n_k.
\end{array}
$$

para obtener la solución $x_1^{(p^{(\nu)}+1)}, \ldots, x_n^{(p^{(\nu)}+1)}$, y el valor de su función objetivo

$$v^{(\nu)} = \sum_{j=1}^{n} \left(c_j - \sum_{i=1}^{m} a_{ij} \lambda_i^{(\nu)} \right) x_j^{(p^{(\nu)}+1)}.$$

El valor de la función objetivo del problema original es

$$z^{(p^{(\nu)}+1)} = \sum_{j=1}^{n} c_j x_j^{(p^{(\nu)}+1)}$$

y el valor de toda restricción de acople es

$$r_i^{(p^{(\nu)}+1)} = \sum_{j=1}^{n} a_{ij} x_j^{(p^{(\nu)}+1)}; \qquad i = 1, \ldots, m.$$

Paso 3: Verificando convergencia. Si $v^{(\nu)} \leq \sigma^{(\nu)}$, hemos encontrado la solución del problema original y ésta se obtiene como la combinación convexa de las soluciones de los subproblemas, es decir,

$$x_j^* = \sum_{s=1}^{p^{(\nu)}+1} u_s^{(\nu)} x_j^{(s)}; \qquad j = 1, \ldots, n$$

y por lo tanto el algoritmo termina. Por el contrario, si $v^{(\nu)} \geq \sigma^{(\nu)}$, la solución corriente del problema relajado puede usarse para mejorar la solución del problema maestro. Se debe entonces actualizar el número de iteración, $\nu \leftarrow \nu + 1$, el número de soluciones disponibles del problema relajado, $p^{(\nu+1)} = p^{(\nu)} + 1$, y regresar al paso 1.

Ejemplo 2.3. *Considerar el siguiente problema*

$$\begin{array}{lccccc}
max & 3x & + & 4y & & \\
s.a & & & & & \\
 & x & & & \leq 1 & \\
 & & & y & \leq 1 & \\
 & 2x & + & 2y & \leq 3 & \\
 & x & + & 3y & \leq 4 & \\
 & & & & x, y & \geq 0
\end{array}$$

Este problema tiene una estructura descomponible y dos restricciones de acople. Su solución es

$$x = \frac{1}{2}, \quad y = 1.$$

Resolvemos este problema en los siguientes pasos usando el algoritmo de Dantzig y Wolfe tal como se describió anteriormente.

Paso 0: Inicialización. La cuenta de interación se inicia en $\nu = 1$. Se obtienen dos ($p^{(\nu)} = 2$) soluciones para el problema relajado resolviendo cada subproblema dos veces.
Los primeros coeficientes de utilidad son $\hat{c}_1^{(1)} = 1$ y $\hat{c}_2^{(1)} = 1$ y los subproblemas para la primera solución son

$$max \quad x$$

s.a

$$0 \leq x \geq 1$$

cuya solución es $x^{(1)} = 1$, y

$$max \quad y$$

s.a

$$0 \leq y \geq 1$$

cuya solución es $y^{(1)} = 1$. El valor de la función objetivo del problema relajado es $z^{(1)} = 7$ y los valores de las restricciones de acople son $r_1^{(1)} = r_2^{(1)} = 4$.

Usando $\hat{c}_1^{(2)} = -1$ y $\hat{c}_2^{(2)} = 4$ los subproblemas se resuelven de nuevo para obtener la segunda solución del problema relajado. Esta solución es $x^{(2)} = 0$ y $y^{(2)} = 1$, la cual nos lleva a tener los siguientes valores $z^{(2)} = 4$, $r_1^{(2)} = 2$ y $r_2^{(2)} = 3$.

Paso 1: Solución del problema maestro. Tenemos que resolver el siguiente problema maestro

$$\begin{array}{ll} max & 7u_1 + 4u_2 \\ \text{s.a} & \\ & \begin{array}{rcl} 4u_1 + 2u_2 \leq 3 & : & \lambda_1 \\ 4u_1 + 3u_2 \leq 4 & : & \lambda_2 \\ u_1 + u_2 = 1 & : & \sigma \\ u_1, u_2 \geq 0. & & \end{array} \end{array}$$

La solución a este problema es $u_1^{(1)} = \frac{1}{2}$ y $u_2^{(1)} = \frac{1}{2}$ con valor en las variables duales de $\lambda_1^{(1)} = \frac{1}{2}, \lambda_2^{(1)} = 0$ y $\sigma^{(1)} = 1$.

Paso 2: Solución del problema relajado. Se resuelven los subproblemas para obtener una solución del problema relajado.

La función objetivo para el primer subproblema es

$$\left(c_1 - \lambda_1^{(1)} a_{11} - \lambda_2^{(1)} a_{21}\right) x = (3 - 3 - 0)x = 0$$

y su solución es $x^{(3)} = 0$.
La función objetivo del segundo subproblema es

$$\left(c_2 - \lambda_1^{(1)} a_{12} - \lambda_2^{(1)} a_{22}\right) y = (4 - 3 - 0)y = y$$

y su solución es $y^{(3)} = 1$.
Para estas soluciones $(x^{(3)} = 0, y^{(3)} = 1)$, el valor de la función objetivo del problema original es $z^{(3)} = 4$ y los valores de las restricciones de acople son $r_1^{(3)} = 2$ y $r_2^{(3)} = 3$, respectivamente.

Paso 3: Verificando convergencia. El valor de la función objetivo del problema relajado corriente es

$$v^{(1)} = 0x^{(3)} + y^{(3)} = 1.$$

Nótese que $\sigma^{(1)} = 1 \leq v^{(1)} = 1$, lo cual implica que la solución óptima del problema original ha sido alcanzada, es decir,

$$\begin{aligned}(x^*, y^*) &= u_1^{(1)} \left(x^{(1)}, y^{(1)}\right) + u_2^{(2)} \left(x^{(2)}, y^{(2)}\right) \\ &= \frac{1}{2}(1,1) + \frac{1}{2}(0,1) \\ &= \left(\frac{1}{2}, 1\right).\end{aligned}$$

2.2. Descomposición de Benders

En esta sección trataremos un problema de programación lineal donde la matriz de coeficientes de las restricciones tiene una estructura por etapas. A este tipo de problemas de programación lineal se les denomina *problemas lineales multietapa*. El método de descomposición y el algoritmo que aquí se presentará es para un *problema lineal bietapa* PL-1, el cual se representa matemáticamente de la forma siguiente: considerando el vector x_1 las variables de la primera etapa y x_2 las variables de la segunda:

$$\begin{array}{llcccl} min & c_1^T x_1 & + & c_2^T x_2 & & \\ \text{s.a} & & & & & \\ & A_1 x_1 & & & = & b_1 \\ & B_1 x_1 & + & A_2 x_2 & = & b_2 \\ & & & x_1, x_2 \geq 0 & & \end{array} \tag{2.6}$$

donde $A_1 \in \mathbb{R}^{m_1 \times n_1}$ y $A_2 \in \mathbb{R}^{m_2 \times n_2}$. Las dimensiones de los demás vectores y matrices se derivan de éstas. La estructura de la matriz de restricciones del problema (denominada triangular inferior por bloques) se presenta en la Figura 2.1.

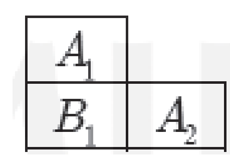


Figura 2.1: Estructura de la matriz de coeficientes de las restricciones en PL-1

El método de descomposición de Benders (Bd) [Benders, Geoffrion:70, Van Slyke] recibe también el nombre de descomposición primal porque el problema maestro fija variables del primal y *descomposición en L* porque se aplica a problemas con matriz de restricciones con dicha forma.

Descompone el problema lineal bietapa PL-1 en un problema maestro y un subproblema. El problema maestro representa la primera etapa más las condiciones necesarias, denominadas *cortes*, derivadas de la segunda etapa. El algoritmo es iterativo y alterna entre la solución del problema maestro y el subproblema.

Más formalmente el problema lineal bietapa PL-1 (2.6) se puede descomponer en dos problemas de programación lineal:

$$\begin{aligned} min \quad & c_1^T x_1 + \theta(x_1) \\ \text{s.a} \quad & \\ & A_1 x_1 = b_1 \\ & x_1 \geq 0 \end{aligned} \tag{2.7}$$

donde la *función de recursos*, $\theta(x_1) \in \mathbb{R}$, es una función poligonal convexa dependiente de x_1 y

$$\begin{aligned} \theta(x_1) = \quad min \quad & c_2^T x_2 \\ \text{s.a} \quad & \\ & A_2 x_2 = b_1 - A_1 x_1 \quad : \quad u \\ & x_2 \geq 0 \end{aligned} \tag{2.8}$$

siendo u las variables duales (o precios sombra) de las restricciones.
Al problema (2.7) se le conoce como *problema maestro* y al (2.8) como *subproblema* en la descomposición de Bd. Se supone que el subproblema es factible para cualquier valor de x_1. Si el subproblema no es factible para cualquier valor de x_1 se verá más adelante cuál es la modificación del algoritmo.

Expresando al subproblema en su forma dual se obtiene:

$$\begin{aligned} \theta(x_1) = \quad max \quad & (b_2 - B_1 x_1)^T u \\ \text{s.a} \quad & \\ & A_2^T u \leq c_2 \end{aligned}$$

Sea $\Gamma = \{u_1, \dots, u_\nu\}$ el conjunto finito de vértices del poliedro convexo definido por la región factible $A_2^T u \leq c_2$. Observemos que la región factible del problema dual no depende del valor de x_1. Se sabe que la solución óptima de un problema lineal reside en un vértice, por lo tanto, el problema se podría resolver, a través del siguiente problema de programación lineal:

$$\begin{aligned} \theta(x_1) = \quad min \quad & \theta \\ \text{s.a} \quad & \\ & \theta \geq (b_2 - B_1 x_1)^T u_1 \\ & \vdots \\ & \theta \geq (b_2 - B_1 x_1)^T u_\nu \end{aligned}$$

donde $\theta \in \mathbb{R}$ y las restricciones se denominan *cortes o planos de corte*.

Entonces, el problema PL-1 se puede expresar como:

$$\begin{array}{ll} min & c_1^T x_1 + \theta \\ \text{s.a} & \\ & A_1 x_1 = b_1 \\ & \theta \geq (b_2 - B_1 x_1)^T u_1 \\ & \vdots \\ & \theta \geq (b_2 - B_1 x_1)^T u_\nu \\ & x_1 \geq 0 \end{array} \tag{2.9}$$

Esta formulación se denomina *problema maestro completo*, ya que contiene todos los cortes posibles. Presenta todas las restricciones de la primera etapa más todas las condiciones necesarias derivadas de la segunda etapa. Observemos que la variable θ es libre de signo.

La solución del problema maestro completo implica disponer de forma explícita de todos los cortes de Benders, lo cual es práticamente imposible en problemas de tamaño realista. En lugar de incluir todos los cortes (lo que implicaría disponer de forma explícita de la función de recursos $\theta(x_1)$), el algoritmo introduce uno de ellos en cada iteración. De esta forma, el *problema maestro relajado* para la iteración j se define como:

$$\begin{array}{ll} min & c_1^T x_1 + \theta \\ \text{s.a} & \\ & A_1 x_1 = b_1 \\ & \theta \geq (b_2 - B_1 x_1)^T u_l \quad l = 1, \ldots, j \\ & x_1 \geq 0 \end{array} \tag{2.10}$$

siendo $\theta \in \mathbb{R}$, l el índice de iteraciones. x_1 y θ variarán en cada iteración.

De cara a una implementación eficiente del algoritmo de descomposición, los cortes de Benders aceptan la siguiente formulación:

$$\theta \geq u_j^T(b_2 - B_1 x_1) = u_j^T(b_2 - B_1 x_1^j + B_1 x_1^j - B_1 x_1) = u_j^T[b_2 - B_1 x_1^j] + u_j^T B_1[x_1^j - x_1]$$

donde x_1 y $f^j = u_j^T[b_2 - B_1 x_1^j]$ son los valores de las variables de la primera etapa y el de la función objetivo de la segunda etapa para la iteración j, respectivamente. Luego el problema maestro y el subproblema para la iteración j tienen la siguiente expresión:

Problema Maestro Benders

$$\begin{array}{ll} min & c_1^T x_1 + \theta \\ \text{s.a} & \\ & A_1 x_1 = b_1 \\ & \theta \geq f^l + u_l^T B_1[x_1^l - x_1] \quad l = 1, \ldots, j \\ & x_1 \geq 0 \end{array} \tag{2.11}$$

Subproblema Benders

$$\begin{aligned} f^j = \quad & min \quad c_2^T x_2 \\ & \text{s.a} \\ & \qquad A_2 x_2 = b_2 - B_1 x_1^j \quad : \quad u_j \\ & \qquad x_2 \geq 0 \end{aligned} \tag{2.12}$$

2.2.1. Cortes de infactibilidad

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

Si el subproblema es factible para un valor de x_1, los precios sombra de sus restricciones son los obtenidos para su solución óptima y los cortes formados se denominan *cortes de optimalidad*. Si el subproblema es infactible los precios sombra que se obtienen al resolver el problema (2.13) de suma de infactibilidades son los que se utilizan para los cortes de infactibilidad.

$$\begin{aligned} min \quad & e^T v^+ + e^T v^- \\ \text{s.a} \quad & \\ & A_2 x_2 + I v^+ - I v^- = b_2 - B_1 x_1 \quad : \quad u_j \\ & x_2, v^+, v^- \geq 0 \end{aligned} \tag{2.13}$$

El problema maestro, considerando ambos tipos de cortes, de optimalidad e infactibilidad, se formula de la siguiente manera:

$$\begin{aligned} min \quad & c_1^T x_1 + \theta \\ \text{s.a} \quad & \\ & A_1 x_1 = b_1 \\ & \delta^l \theta \geq f^l + u_l^T B_1 [x_1^l - x_1] \quad l = 1, \ldots, j \\ & x_1 \geq 0 \end{aligned} \tag{2.14}$$

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

2.2.2. Algoritmo

En cada iteración del algoritmo se resuelve el problema maestro relajado y se pasa el valor del vector x_1^j al subproblema. Éste optimiza x_2 con los recursos $b_2 - B_1 x_1^j$ y pasa las variables duales u_j al problema maestro. Una cota superior $\bar{z}$ del valor óptimo de la función objetivo del problema PL-1 en la iteración j viene dada por $c_1^T x_1^j + c_2^T x_2^j$ siendo x_1^j y x_2^j soluciones factibles en maestro y subproblema en esa iteración. En cada iteración, el valor obtenido por la función objetivo del problema maestro relajado $\underline{z}$, dada por $c_1^T x_1^j + \theta^j$, es una cota inferior del problema PL-1. La *condición de convergencia* para la terminación del algoritmo es la coincidencia de ambas cotas con una tolerancia relativa ε (por ejemplo, 10^{-4}). La sucesión de estas cotas inferiores es monótona creciente dado que en cada iteración el problema maestro relajado contiene mayor número de restricciones, mientras que la cota superior no es necesariamente decreciente. Por esta razón, se toma como cota superior para evaluar convergencia el mínimo de todas las cotas superiores previas.

Esquemáticamente, el algoritmo se formula a continuación:

Paso 1: Inicialización. $j = 0$, $\bar{z} = \infty$, $\underline{z} = -\infty$, $\varepsilon = 10^{-4}$.

Paso 2: Resolución del problema maestro.

$$
\begin{array}{ll}
min & c_1^T x_1 + \theta \\
\text{s.a} & \\
& A_1 x_1 = b_1 \\
& \delta^l \theta \geq f^l + u_l^T B_1 [x_1^l - x_1] \quad l = 1, \ldots, j \\
& x_1 \geq 0
\end{array}
$$

Obtener la solución (x_1^j, θ^j) y evaluar la cota inferior $\underline{z} = c_1^T x_1^j + \theta^j$.
Mientras no se halla generado ningún corte de optimalidad, se fija el valor de la variable de recurso θ a cero, pues en otro caso el problema maestro es no acotado. Una vez obtenido algún corte de infactibilidad, esta variable pasa a ser libre.

Paso 3: Resolución del subproblema de suma de infactibilidades.

$$
\begin{array}{lll}
f^j = & min & e^T v^+ + e^T v^- \\
& \text{s.a} & \\
& & A_2 x_2 + I v^+ - I v^- = b_2 - B_1 x_1 \quad : \quad u_j \\
& & x_2, v^+, v^- \geq 0
\end{array}
$$

Si $f^j \geq 0$, obtener u_j, formar un corte de infactibilidad y añadirlo al problema maestro, incrementar el número de iteraciones $j = j + 1$ e ir al paso 2.
Si $f^j = 0$, ir al paso 4.

Paso 4: Resolución del subproblema.

$$
\begin{array}{lll}
f^j = & min & c_2^T x_2 \\
& \text{s.a} & \\
& & A_2 x_2 = b_2 - B_1 x_1^j \quad : \quad u_j \\
& & x_2 \geq 0
\end{array}
$$

Obtener x_2^j y actualizar cota superior $\bar{z} = c_1^T x_1^j + c_2^T x_2^j$.

Paso 5: Criterio de paro. Si $\frac{|\bar{z}-\underline{z}|}{\bar{z}} \leq \varepsilon$ detener el algoritmo. En otro caso, obtener u_j, formar un corte de optimalidad y añadirlo al problema maestro, incrementar el número de iteraciones $j = j + 1$ e ir al paso 2.

A continuación se presenta con un ejemplo la aplicación de los pasos del algoritmo de descomposición de Benders.

Ejemplo 2.4. *Consideremos el siguiente problema de programación lineal*

$$\begin{array}{rlcr} min & -2x - y & & \\ s.a & & & \\ & x & \leq & 4 \\ & x + y & \leq & 5 \\ & 2x + 3y & \leq & 12 \\ & x, y \geq 0 & & \end{array}$$

cuya solución es $(x, y) = (4, 1)$*. Suponemos una solución inicial* $(x, y) = (0, 4)$*,* x *variable de la primera etapa e* y *variable de la segunda etapa.*

Paso 4: El primer subproblema es

$$\begin{array}{rll} \theta(x) = & min & -y \\ & \text{s.a} & \\ & & y \leq 5 - 0 = 5 \\ & & 3y \leq 12 - 2(0) = 12 \\ & & y \geq 0, \end{array}$$

que al resolverlo da lugar a $f^1 = -4$ y $u^1 = (0, -\frac{1}{3})^T$. Luego la cota inferior del problema es $\underline{z} = -\infty$ y la cota superior es $\bar{z} = -4$.

Paso 2: El problema maestro es

$$\begin{array}{rl} min & -2x + \theta \\ \text{s.a} & \\ & x \leq 4 \\ & \theta - \frac{2}{3}x \geq -4 \\ & x \geq 0, \end{array}$$

cuya solución óptima es $x = 4$, $\theta = -\frac{4}{3}$, y la cota inferior del problema es $\underline{z} = -\frac{28}{3}$.

Paso 4: Para $x = 4$ el subproblema es

$$\begin{array}{rll} \theta(x) = & min & -y \\ & \text{s.a} & \\ & & y \leq 1 \\ & & 3y \leq 12 - 2(4) = 4 \\ & & y \geq 0, \end{array}$$

que al resolverlo resulta, $f^1 = -1$ y $u^1 = (-1, 0)^T$. Luego la cota superior es $\bar{z} = min\{-9, -4\} = -9$.

Paso 2: El problema maestro es

$$\begin{array}{rl} min & -2x + \theta \\ \text{s.a} & \\ & x \leq 4 \\ & \theta - \frac{2}{3}x \geq -4 \\ & \theta - x \geq -5 \\ & x \geq 0, \end{array}$$

cuya solución óptima es $x = 4$, $\theta = -1$ y la cota inferior del problema es $\underline{z} = -9$. Una vez formulado y resuelto el subproblema se obtiene $\bar{z} = min\{-9, -9\} = -9$ y, por lo tanto, el algoritmo converge.

.

Capítulo 3

Descomposición en problemas multi - divisionales

En este capítulo construimos dos algoritmos, el primero es para un problema de programación lineal con una estructura de bloques y una restricción de acople y el otro es para un problema bi - divisional. La idea básica de nuestros algoritmos es reducir el problema original en una secuencia de subproblemas más sencillos y, a partir de estos, encontrar el óptimo.

En la Sección 3.1 descomponemos un problema con estructura de bloques y una restricción de acople en n subproblemas, cada uno de ellos contiene una parte del recurso común y un subconjunto del total de variables, y son derivados de los bloques que contiene el problema original. En la Sección 3.2 presentamos el algoritmo para resolver un problema con estructura de bloques y una restricción de acople, así, como la prueba de que si la condición de paro se verifica, entonces se ha logrado el óptimo de este tipo de problemas.

En la Sección 3.3 expresamos a un problema bi - divisional PL-2 en dos problemas lineales que interactúan entre sí para lograr una solución óptima del problema PL-2. En esta sección se prueba que si el valor de la variable libre del problema maestro bi - divisional relajado es igual al valor de la función objetivo del subproblema bi - divisional, entonces la solución óptima del problema PL-2 se ha alcanzado.

A continuación presentamos la segmentación de un problema con estructura de bloques y una restricción de acople:

3.1. Segmentación

Consideremos el siguiente problema de programación lineal, el cual denotaremos por *problema cero*

$$
\begin{array}{lllllll}
max & \sum_{k=1}^{n} c_k^T x_k & & & & & \\
\text{s.a.} & & & & & & \\
& A_1x_1 & + & A_2x_2 & + \ldots + & A_nx_n & \leq d \\
& B_1x_1 & & & & & \leq b_1 \\
& & & B_2x_2 & & & \leq b_2 \\
& & & & \vdots & & \vdots \\
& & & & & B_nx_n & \leq b_n \\
& & & & & x_k & \geq 0
\end{array}
$$

donde $A_i \in \mathbb{R}^{1\times n_i}$ y $B_i \in \mathbb{R}^{m_i\times n_i}$. Las dimensiones de los vectores se derivan de las dimensiones de A_i y B_i.

Suposición 3.1. *En este capítulo suponemos que el conjunto*

$$S_k = \{x_k : B_kx_k \leq b_k \quad y \quad x_k \geq 0\}, \forall k = 1, \ldots, n$$

es no vacío, compacto y convexo, lo cual implica que la función objetivo del problema cero está acotada y que los subproblemas que se derivan del problema cero tienen solución.

Si suponemos una distribución a priori de d, entonces el problema cero puede segmentarse en n subproblemas de programación lineal, donde el k- *ésimo subproblema* es

$$
\begin{array}{lrcl}
max & c_k^T x_k & & \\
\text{s.a.} & & & \\
& A_kx_k \leq d_k & : & u(d_k) \\
& B_kx_k \leq b_k & : & v_k \\
& x_k \geq 0 & &
\end{array}
$$

donde $\sum_{k=1}^{n} d_k = d$ y las variables duales $u(d_k)$ y v_k están indicadas.

Observación 3.1. *Para cualesquiera números positivos* d_k $(k = 1, \ldots, n)$ *tales que* $\sum_{k=1}^{n} d_k = d$*, las soluciones óptimas de los subproblemas forman una solución factible del problema cero.*

Observación 3.2. *El problema principal es repartir las d unidades del recurso común entre los subproblemas de tal manera que las soluciones óptimas de los subproblemas formen una solución óptima del problema cero.*

Supongamos que el problema cero modela las decisiones de una empresa con n divisiones independientes, la cual quiere maximizar sus ganancias através de una distribución adecuada del recurso d entre sus divisiones (subproblemas). En este contexto, $u(d_k)$ se puede interpretar como el incremento que obtendría la división k por cada unidad adicional del recurso d.

Los precios $u(d_k)$ son reportados al centro, donde se comparan entre ellos. Si la división h, dice que paga más por cada unidad del recurso d que la división k $(u(d_h) > u(d_k))$, entonces la división k debe trasferir una parte de su recurso a la división h, hasta que esta situación cambie. Esta transferencia causará un incremento en la función objetivo del problema h, que es mayor que el correspondiente decrecimiento en la función objetivo del problema k. Luego, la función objetivo del problema cero puede ser mejorada por estas transferencias.

A continuación formalizaremos la idea anterior:

Definición 3.1. *A $u^-(d_k)$ y $u^+(d_k)$ se les denominará los precios sombra a la izquierda y a la derecha del recurso común d_k, en el k-ésimo subproblema.*

Se sabe que, si se resuelve un problema de programación lineal se obtiene, entre otras cosas, el valor de sus variables duales y las regiones donde éstas gobiernan. Así, si resolvemos el k-ésimo subproblema obtenemos el valor de $u(d_k)$ (precio sombra del recurso cumún d_k cuando al subproblema k se le asignaron d_k unidades del bien común) y la región donde éste gobierna, que la denotaremos por $I_k = [\alpha_k, \beta_k]$. Por la Definición 3.1, en I_k se tiene que $u(d_k) = u^-(d_k) = u^+(d_k)$, pero en el caso de que β_k sea finito se tiene $u(\beta_k) = u^-(\beta_k) \neq u^+(\beta_k)$, porque a la derecha de β_k ocurre un cambio de base, y por lo tanto, es otro precio sombra el que gobierna. Por un argumento similar al anterior, $u(\alpha_k) = u^+(\alpha_k) \neq u^-(\alpha_k)$. En lo sucesivo, $N = \{1, \ldots, n\}$ será el conjunto formado por las numeraciones de los subproblemas.

Intuitivamente, α_k indica hasta dónde puede ser disminuido d_k de tal forma que la base actual siga siendo óptima y β_k, hasta dónde puede ser aumentado. Por ello, $\alpha_k \leq d_k \leq \beta_k$.

La siguiente definición será de mucha importancia en esta sección.

Definición 3.2. *u_s^- y u_t^+ son tales que*

$$\begin{aligned} s &= argmin\{u^-(d_k) : k \in N \quad y \quad d_k \neq 0\} \\ t &= argmax\{u^+(d_k) : k \in N\}. \end{aligned}$$

Observación 3.3. *Nótese que aunque d_k sea uno de los extremos del intervalo $[\alpha_k, \beta_k]$, $u^+(d_k)$ es el precio sombra al que recibe el subproblema k cada unidad del bien común y $u^-(d_k)$ es al precio sombra al que transfiere.*

3.2. Algoritmo

El algoritmo que se va a presentar trabaja con los precios sombra de cada subproblema y con el recurso común d por medio de la transferencia de un monto adecuado del recurso común del subproblema que valora menos cada unidad del recurso d al subproblema que lo valora más. Repitiendo este proceso se logra la solución óptima del problema cero.

Observación 3.4. *Se trasferirá del subproblema s al subproblema t*

Algoritmo 1 Para encontrar la solución óptima del problema cero

Entrada: El problema cero

Salida: La solución óptima del problema cero

1. Asignar a cada subproblema una parte del recurso d y resolverlos para obtener el valor $u(d_k)$ y el rango I_k, $\forall k \in N$.
2. Calcular u_s^- y u_t^+.
3. Si $u_s^- < u_t^+$, entonces transferir θ unidades del recurso d_s con $\theta = min\{d_s - \alpha_s, \beta_t - d_t\}$.
   - Actualizar los lados derechos y los precios sombra de los subproblemas s y t y regresar al paso 2.
4. Si no, el algoritmo termina y la solución es la formada por las soluciones óptimas de los subproblemas.

Note que por cada iteración sólo dos problemas secundarios son modificados, esto implica que sólo estos dos problemas se actualizarán en el paso 3.

La idea general del Algoritmo 1 es como sigue: Primero distribuye el recurso d entre los subproblemas y resuelve éstos para obtener los precios sombra correspondientes al recurso común y los rangos donde éstos gobiernan (paso 1). Después elige a los subproblemas con menor precio sombra a izquierda y mayor precio sombra a derecha (s y t), respectivamente (paso 2). Luego, si el precio sombra a izquierda del subproblema s es menor que el precio sombra a derecha del subproblema t, entonces se transfiere un monto del recurso común con la propiedad de que al disminuir el lado derecho del subproblema s y aumentar el del subproblema t no hay cambio de base (paso 3). Entonces regresamos al paso 2. El Algoritmo 1 termina hasta que $u_s^- \geq u_t^+$, lo que significará que el óptimo del problema cero ha sido alcanzado.

En la siguiente sección probaremos que un número finito de pasos las soluciones factibles generadas por el Algoritmo 1 convergen a la solución óptima del problema cero.

3.2.1. Convergencia

Supongamos que cada uno de los subproblemas se resuelven con lado derecho (d_k, b_k) para algunos números positivos d_k $(k = 1, \ldots, n)$ tales que $\sum_{k=1}^{n} d_k = d$, con lo cual obtenemos sus soluciones óptimas, los valores de sus funciones objetivos y el valor de $u(d_k)$ $(k = 1, \ldots, n)$.

Lema 1. *Sean $p, q \in N$. Si se transfieren θ unidades del recurso común del subproblema p al subproblema q, donde $\theta = min\{d_p - \alpha_p, \beta_q - d_q\}$, entonces el valor de la función objetivo del problema cero se modifica en $\theta(u^+(d_q) - u^-(d_p))$.*

Demostración.

Sean z_k con $k \in N$ los valores óptimos de las funciones objetivos de los subproblemas k con lados derechos (d_k, b_k). Entonces, el valor de la función objetivo del problema cero es $z = \sum_{k\in N} z_k$. Dado que $(d_p - \theta) \in I_p$ y $(d_q + \theta) \in I_q$, entonces por el Teorema 7 z_q aumenta en $\theta u^+(d_q)$ y z_p disminuye en $\theta u^-(d_p)$. Como los valores de z_k con $k \in N \backslash \{p, q\}$ no han sido modificados, entonces el nuevo valor de la función objetivo del problema cero es

$$\begin{aligned}\bar{z} &= \sum_{k\in N\backslash\{p,q\}} z_k + z_p - \theta u^-(d_p) + z_q + \theta u^+(d_q) \\ &= \sum_{k\in N} z_k + \theta(u^+(d_q) - u^-(d_p))\end{aligned}$$

lo cual prueba el lema. □

Proposición 1. *Si $u_s^- \geq u_t^+$, entonces $u^-(d_p) \geq u^+(d_q)$, $\forall p, q \in N$ y $p \neq q$.*

Demostración.

Sean $p, q \in N$ tales que $p \neq q$. Por hipótesis y la Definición 3.2 se obtiene

$$u^-(d_p) \geq u_s^- \geq u_t^+ \geq u^+(d_q)$$

lo cual implica que $u^-(d_p) \geq u^+(d_q)$. □

Proposición 2. *Después de un número finito de pasos, se tiene que $u_s^- \geq u_t^+$.*

Demostración.

Afirmación 1: El número de iteraciones es finito.

Como el número de matrices básicas de un problema de programación lineal acotado con un número finito de restricciones es siempre finito y los precios sombra sólo dependen de la matriz básica, entonces el número posible de precios sombra entre todos los subproblemas es finito. Lo anterior implica que el número de iteraciones posibles es finito.

Afirmación 2: El algoritmo 1 no tiene ciclo.

Si el algoritmo tuviera un ciclo, entonces por el Lema 1 la función objetivo del problema cero crece indefinidamente pero, por la suposición 3.1, la función objetivo del problema cero está acotada; luego, el algoritmo no tiene ciclo.

Si la desigualdad $u_s^- < u_t^+$ se mantiene indefinidamente, entonces por el Lema 1 la función objetivo del problema cero crece indefinidamente pero, por la suposición 3.1, la función objetivo del problema cero está acotada, las u_s^- son monótonas crecientes respecto a las iteraciones del algoritmo, las u_t^+ son monótonas decrecientes y por la Afirmación 2 el algoritmo no tiene ciclo. Luego, en alguna iteración $u_s^- \geq u_t^+$. □

Teorema 8. *Si la condición de paro se verifica, entonces se ha encontrado una solución óptima del problema cero.*

DEMOSTRACIÓN.

Por el Lema 1 en cada iteración la función objetivo del problema cero crece y por la Proposición 1, si se realiza una transferencia entre cualesquiera dos subproblemas la función objetivo del problema cero decrece, lo cual implica que el óptimo se ha alcanzado. □

Ejemplo 3.1. *Consideremos el problema*

$$\begin{array}{llcll}
max & 2x_1 + 3x_2 & + & 5x_3 + 3x_4 & & \\
s.a & & & & & \\
 & x_1 + 6x_2 & + & 4x_3 + 2x_4 & \leq & 20 \\
 & 2x_1 + x_2 & & & \leq & 8 \\
 & 5x_1 + 6x_2 & & & \leq & 20 \\
 & & & x_3 + x_4 & \leq & 6 \\
 & & & 2x_3 + 3x_4 & \leq & 15 \\
 & & & x_i & \geq & 0.
\end{array}$$

Note que este problema tiene estructura de bloque con una restricción en común

$$x_1 + 6x_2 + 4x_3 + 2x_4 \leq 20.$$

La solución óptima es

$$x_1^* = 4, \quad x_2^* = 0, \quad x_3^* = \frac{9}{4}, \quad x_4^* = \frac{7}{2}.$$

Encontremos la solución óptima de este problema de programación lineal aplicando los pasos del Algoritmo 1 tal como se muestra acontinuación:

Paso 1: Asignamos a cada subproblema 10 unidades del recurso común, es decir, $d_1 = d_2 = 10$. Los subproblemas son:

$$\begin{array}{llcc}
max & 2x_1 + 3x_2 & & \\
s.a & & & \\
 & x_1 + 6x_2 & \leq & d_1 = 10 \\
 & 2x_1 + x_2 & \leq & 8 \\
 & 5x_1 + 6x_2 & \leq & 20 \\
 & x_1, x_2 & \geq & 0
\end{array}
\qquad
\begin{array}{llcc}
max & 5x_3 + 3x_4 & & \\
s.a & & & \\
 & 4x_3 + 2x_4 & \leq & d_2 = 10 \\
 & x_3 + x_4 & \leq & 6 \\
 & 2x_3 + 3x_4 & \leq & 15 \\
 & x_3, x_4 & \geq & 0
\end{array}$$

cuyas soluciones y precios sombra asociado al recurso común son:

$$x_1 = \frac{5}{2}, \;\; x_2 = \frac{5}{4}, \;\; x_3 = 0, \;\; x_4 = 5$$

$$\begin{array}{ll}
u(d_1 = 10) = \dfrac{1}{4} & \qquad I_1 = [4, 20] \\
u(d_2 = 10) = \dfrac{5}{4} & \qquad I_2 = [10, 18].
\end{array}$$

Paso 2: Calculamos s y t de acuerdo a la Definición 3.2

$$\begin{aligned} 1 = s = argmin\{u^-(d_1 = 10) = \frac{1}{4}, u^-(d_2 = 10) = \frac{3}{2}\} \\ 2 = t = argmax\{u^+(d_1 = 10) = \frac{1}{4}, u^+(d_2 = 10) = \frac{5}{4}\}. \end{aligned}$$

Así,

$$u_s^- = \frac{1}{4}, \;\; u_t^+ = \frac{5}{4}.$$

Paso 3: Como $u_s^- < u_t^+$, entonces se transfiere

$$6 = \theta = \text{mín}\{10 - 4, 18 - 10\}$$

del subproblema 1 al subproblema 2 y los nuevos subproblemas son

$$\begin{array}{llcc} max & 2x_1 + 3x_2 & & \\ s.a & & & \\ & x_1 + 6x_2 & \leq & d_1 = 4 \\ & 2x_1 + x_2 & \leq & 8 \\ & 5x_1 + 6x_2 & \leq & 20 \\ & x_1, x_2 & \geq & 0 \end{array} \qquad \begin{array}{llcc} max & 5x_3 + 3x_4 & & \\ s.a & & & \\ & 4x_3 + 2x_4 & \leq & d_2 = 16 \\ & x_3 + x_4 & \leq & 6 \\ & 2x_3 + 3x_4 & \leq & 15 \\ & x_3, x_4 & \geq & 0 \end{array}$$

cuyas soluciones y precios sombra asociado al recurso común son:

$$x_1 = 4, \;\; x_2 = 0, \;\; x_3 = \frac{9}{4}, \;\; x_4 = \frac{7}{2}$$

$$\begin{aligned} u(d_1 = 4) &= 2 \qquad\qquad & I_1 &= [0, 4] \\ u(d_2 = 10) &= \frac{5}{4} \qquad\qquad & I_2 &= [10, 18]. \end{aligned}$$

Con ello regresamos al paso 2.

Paso 2: Calculamos s y t de acuerdo a la Definición 3.2

$$\begin{aligned} 2 = s = argmin\{u^-(d_1 = 4) = 2, u^-(d_2 = 16) = \frac{5}{4}\} \\ 2 = t = argmax\{u^+(d_1 = 4) = \frac{1}{4}, u^+(d_2 = 16) = \frac{5}{4}\}. \end{aligned}$$

Así,

$$u_s^- = \frac{5}{4}, \;\; u_t^+ = \frac{5}{4}.$$

Paso 4: Como $u_s^- = u_t^+$, entonces el algoritmo termina y la solución óptima es

$$x^* = \left(4, 0, \frac{9}{4}, \frac{7}{2}\right).$$

Ejemplo 3.2. *Consideremos el siguiente problema que contiene tres bloques y una restricción de acople*

$$\begin{array}{lccccccl}
max & 5x_1+3x_2 & + & 5x_3+3x_4 & + & x_5+7x_6 & & \\
s.a & & & & & & & \\
 & x_1+6x_2 & + & 4x_3+2x_4 & + & x_5+3x_6 & \leq 30 \\
 & 2x_1+x_2 & & & & & \leq 8 \\
 & 5x_1+6x_2 & & & & & \leq 20 \\
 & & & x_3+x_4 & & & \leq 6 \\
 & & & 2x_3+3x_4 & & & \leq 15 \\
 & & & & & 5x_5+6x_6 & \leq 30 \\
 & & & & & x_5+3x_6 & \leq 16 \\
 & & & & & x_i & \geq 0.
\end{array}$$

La solución óptima de este problema es

$$x_1^*=4, x_2^*=0$$
$$x_3^*=\frac{3}{8}, x_4^*=\frac{19}{4}$$
$$x_5^*=0, x_6^*=5.$$

Paso 1: Asignamos a cada subproblema 10 unidades del recurso común, es decir, $d_1=d_2=d_3=10$. Los subproblemas son:

$$\begin{array}{llcc}
max & 5x_1+3x_2 & & \\
s.a & & & \\
 & x_1+6x_2 & \leq & d_1=10 \\
 & 2x_1+x_2 & \leq & 8 \\
 & 5x_1+6x_2 & \leq & 20 \\
 & x_1,x_2 & \geq & 0
\end{array}
\qquad
\begin{array}{llcc}
max & 5x_3+3x_4 & & \\
s.a & & & \\
 & 4x_3+2x_4 & \leq & d_2=10 \\
 & x_3+x_4 & \leq & 6 \\
 & 2x_3+3x_4 & \leq & 15 \\
 & x_3,x_4 & \geq & 0
\end{array}$$

$$\begin{array}{llcc}
max & x_5+3x_6 & & \\
s.a & & & \\
 & x_5+3x_6 & \leq & d_3=10 \\
 & 5x_5+6x_6 & \leq & 30 \\
 & x_5+3x_6 & \leq & 16 \\
 & x_5,x_6 & \geq & 0
\end{array}$$

cuyas soluciones y precios sombra asociados al recurso común son:

$$x_1=4, x_2=0,$$
$$x_3=0, x_4=5$$
$$x_5=0, x_6=\frac{10}{3}.$$

$$u(d_1 = 10) = 0 \qquad\qquad I_1 = [4, \infty)$$
$$u(d_2 = 10) = \frac{5}{4} \qquad\qquad I_2 = [10, 18]$$
$$u(d_3 = 10) = \frac{7}{3} \qquad\qquad I_3 = [0, 15].$$

Paso 2: Calculamos s y t de acuerdo a la Definición 3.2

$$1 = s = argmin\{u^-(d_1 = 10) = 0, u^-(d_2 = 10) = \frac{3}{2}, u^-(d_3 = 10) = \frac{7}{3}\}$$
$$3 = t = argmax\{u^+(d_1 = 10) = 0, u^+(d_2 = 10) = \frac{5}{4}, u^+(d_3 = 10) = \frac{7}{3}\}.$$

Así,

$$u_s^- = 0, \ \ u_t^+ = \frac{7}{3}.$$

Paso 3: Como $u_s^- < u_t^+$, entonces se transfiere

$$5 = \theta = \text{mín}\{10 - 4, 15 - 10\}$$

del subproblema 1 al subproblema 3 y los nuevos subproblemas son

$$\begin{array}{lrcl} max & 5x_1 + 3x_2 & & \\ s.a & & & \\ & x_1 + 6x_2 & \leq & d_1 = 5 \\ & 2x_1 + x_2 & \leq & 8 \\ & 5x_1 + 6x_2 & \leq & 20 \\ & x_1, x_2 & \geq & 0 \end{array} \qquad \begin{array}{lrcl} max & 5x_3 + 3x_4 & & \\ s.a & & & \\ & 4x_3 + 2x_4 & \leq & d_2 = 10 \\ & x_3 + x_4 & \leq & 6 \\ & 2x_3 + 3x_4 & \leq & 15 \\ & x_3, x_4 & \geq & 0 \end{array}$$

$$\begin{array}{lrcl} max & x_5 + 3x_6 & & \\ s.a & & & \\ & x_5 + 3x_6 & \leq & d_3 = 15 \\ & 5x_5 + 6x_6 & \leq & 30 \\ & x_5 + 3x_6 & \leq & 16 \\ & x_5, x_6 & \geq & 0 \end{array}$$

cuyas soluciones y precios sombra asociados al recurso común son:

$$x_1 = 4, x_2 = 0,$$
$$x_3 = 0, x_4 = 5$$
$$x_5 = 0, x_6 = 5.$$

$$u(d_1 = 5) = 0 \qquad\qquad I_1 = [4, \infty)$$
$$u(d_2 = 10) = \frac{5}{4} \qquad\qquad I_2 = [10, 18]$$
$$u(d_3 = 15) = \frac{7}{3} \qquad\qquad I_3 = [15, \infty)$$

y regresamos nuevamente al paso 2.

Paso 2: Calculando nuevamente s y t de acuerdo a la Definición 3.2 tenemos

$$1 = s = argmin\{u^-(d_1 = 5) = 0, u^-(d_2 = 10) = \frac{3}{2}, u^-(d_3 = 15) = \frac{7}{3}\}$$
$$2 = t = argmax\{u^+(d_1 = 5) = 0, u^+(d_2 = 10) = \frac{5}{4}, u^+(d_3 = 15) = 0\}.$$

Así,

$$u_s^- = 0,\ \ u_t^+ = \frac{5}{4}.$$

Paso 3: Como $u_s^- < u_t^+$, entonces se transfiere

$$1 = \theta = \min\{5 - 4, 18 - 10\}$$

del subproblema 1 al subproblema 2 y al reformular los subproblemas se tiene

$$\begin{array}{llcl} max & 5x_1 + 3x_2 & & \\ s.a & & & \\ & x_1 + 6x_2 & \leq & d_1 = 4 \\ & 2x_1 + x_2 & \leq & 8 \\ & 5x_1 + 6x_2 & \leq & 20 \\ & x_1, x_2 & \geq & 0 \end{array} \qquad \begin{array}{llcl} max & 5x_3 + 3x_4 & & \\ s.a & & & \\ & 4x_3 + 2x_4 & \leq & d_2 = 11 \\ & x_3 + x_4 & \leq & 6 \\ & 2x_3 + 3x_4 & \leq & 15 \\ & x_3, x_4 & \geq & 0 \end{array}$$

$$\begin{array}{llcl} max & x_5 + 3x_6 & & \\ s.a & & & \\ & x_5 + 3x_6 & \leq & d_3 = 15 \\ & 5x_5 + 6x_6 & \leq & 30 \\ & x_5 + 3x_6 & \leq & 16 \\ & x_5, x_6 & \geq & 0 \end{array}$$

que al resolverlos obtenemos que sus soluciones son

$$x_1 = 4, x_2 = 0,$$
$$x_3 = \frac{3}{8}, x_4 = \frac{19}{4}$$
$$x_5 = 0, x_6 = 5$$

y los precios sombra asociados al recurso común son

$$\begin{array}{ll} u(d_1 = 4) = 5 & I_1 = [0, 4] \\ u(d_2 = 11) = \frac{5}{4} & I_2 = [10, 18] \\ u(d_3 = 15) = 0 & I_3 = [15, \infty) \end{array}$$

donde, al no cumplirse el criterio de paro, tenemos que regresar al paso 2.

Paso 2: Calculando nuevamente s y t de acuerdo a la Definición 3.2 tenemos

$$2 = s = argmin\{u^-(d_1 = 4) = 5, u^-(d_2 = 11) = \frac{5}{4}, u^-(d_3 = 15) = \frac{7}{3}\}$$
$$2 = t = argmax\{u^+(d_1 = 4) = 0, u^+(d_2 = 11) = \frac{5}{4}, u^+(d_3 = 15) = 0\}.$$

Así,

$$u_s^- = \frac{5}{4}, \;\; u_t^+ = \frac{5}{4}.$$

Paso 3: Como $u_s^- = u_t^+$, entonces el algoritmo converge y la solución óptima es

$$x^* = \left(4, 0, \frac{3}{8}, \frac{19}{4}, 0, 5\right).$$

3.3. Problema bi - divisional

En esta sección consideraremos un *problema bi - divisional* PL-2, el cual se representa matemáticamente de la forma siguiente, considerando el vector x_1 las variables de la primera división y x_2 las variables de la segunda:

$$\begin{array}{llcll} max & c_1^T x_1 & + & c_2^T x_2 & \\ s.a & & & & \\ & A_1 x_1 & + & A_2 x_2 & \leq d \\ & B_1 x_1 & & & \leq b_1 \\ & & & B_2 x_2 & \leq b_2 \\ & & & x_1, x_2 & \geq 0 \end{array} \tag{3.1}$$

donde $A_1 \in \mathbb{R}^{m \times n_1}$, $A_2 \in \mathbb{R}^{m \times n_2}$, $B_1 \in \mathbb{R}^{m_1 \times n_1}$ y $B_2 \in \mathbb{R}^{m_2 \times n_2}$. Las dimensiones de los vectores se derivan de éstas.

El problema lineal bi - divisional PL-2 se puede interpretar como

$$\begin{array}{ll} max & c_1^T x_1 + f(x_1) \\ s.a & \\ & B_1 x_1 \leq b_1 \\ & x_1 \geq 0 \end{array} \tag{3.2}$$

donde la función $f(x_1) \in \mathbb{R}$ y es la función objetivo de la división 2, como función de las decisiones de la división 1, y tiene la siguiente expresión:

$$\begin{array}{rlll} f(x_1) = & max & c_2^T x_2 & \\ & s.a & & \\ & & A_2 x_2 \leq d - A_1 x_1 & : \;\; \pi \\ & & B_2 x_2 \leq b_2 & : \;\; v \\ & & x_2 \geq 0 & \end{array} \tag{3.3}$$

siendo π y v las variables duales (o precios sombra) de las restricciones.
Al problema 3.2 se le denominará como *problema maestro divisional* y al 3.3 como *subproblema divisional*. Supondremos que el subproblema divisional es factible para cualquier $x_1 \in S_1$. El dual del problema 3.3 es

$$\begin{aligned} f(x_1) = \quad & min \quad (d - A_1 x_1)^T \pi + b_2^T v \\ & \text{s.a} \\ & \qquad A_2^T \pi + B_2^T v \geq c_2^T \\ & \qquad \pi, v \geq 0. \end{aligned} \tag{3.4}$$

Sea $\Sigma = \{(\pi_1, v_1), \dots, (\pi_\vartheta, v_\vartheta)\}$ el conjunto finito de vértices del poliedro convexo definido por $A_2^T \pi + B_2^T v \geq c_2^T$. Observemos que al igual que el caso de Benders la región factible del problema dual no depende de x_1. Dado que la solución óptima del problema 3.4 ocurre en uno de los elementos de Σ, el problema se puede resolver a través de la enumeración de todos los elementos de Σ, es decir,

$$f(x_1) = \text{mín}\left\{(d - A_1 x_1)^T \pi_l + b_2^T v_l\right\} \quad l = 1, \dots, \vartheta. \tag{3.5}$$

Si esta ecuación se expresa como problema lineal se obtiene

$$\begin{aligned} f(x_1) = \quad & max \quad \delta \\ & \text{s.a} \\ & \qquad \delta \leq (d - A_1 x_1)^T \pi_1 + b_2^T v_1 \\ & \qquad \vdots \\ & \qquad \delta \leq (d - A_1 x_1)^T \pi_\vartheta + b_2^T v_\vartheta \end{aligned} \tag{3.6}$$

donde $\delta \in \mathbb{R}$ y las restricciones, al igual que en la descomposición de Benders, se denominan cortes.
Entonces, el problema PL-2 puede expresarse como

$$\begin{aligned} & max \quad c_1^T x_1 + \delta \\ & \text{s.a} \\ & \qquad B_1 x_1 \leq b_1 \\ & \qquad \delta \leq (d - A_1 x_1)^T \pi_1 + b_2^T v_1 \\ & \qquad \vdots \\ & \qquad \delta \leq (d - A_1 x_1)^T \pi_\vartheta + b_2^T v_\vartheta \\ & \qquad x_1 \geq 0 \end{aligned} \tag{3.7}$$

A esta forma de escribir al problema PL-2 se denominará *problema maestro divisional completo*, ya que presenta las restricciones de la primera división más todos los cortes posibles derivados de la segunda división.

Observación 3.5. *El problema maestro divisional completo está acotado, ya que el conjunto S_1 es acotado y la variable δ están restringida por los cortes.*

Resolver el problema maestro divisional completo para problemas con cientos de variables es prácticamente imposible, porque implica disponer de todos los cortes (lo que implicaría disponer explícitamente de la función $f(x_1)$). Al igual que en el algoritmo de Benders se introduce uno de ellos en cada iteración; de esta forma el *problema maestro divisional relajado* para la iteración j se define como:

$$
\begin{aligned}
&max \quad c_1^T x_1 + \delta \\
&\text{s.a} \\
&\qquad B_1 x_1 \leq b_1 \\
&\qquad \delta \leq (d - A_1 x_1)^T \pi_l + b_2^T v_l \quad l = 1, \ldots, j \\
&\qquad x_1 \geq 0.
\end{aligned}
\tag{3.8}
$$

Como en el algoritmo de Benders los cortes aceptan la siguiente formulación:

$$
\begin{aligned}
&\delta \leq \pi_j^T (d - A_1 x_1) + b_2^T v_j \\
&\delta \leq \pi_j^T (d - A_1 x_1^j + A_1 x_1^j - A_1 x_1) + b_2^T v_j \\
&\delta \leq \pi_j^T (d - A_1 x_1^j) + b_2^T v_j + \pi_j^T A_1 (x_1^j - x_1)
\end{aligned}
$$

donde $g^j = \pi_j^T (d - A_1 x_1^j) + b_2^T v_j$ y x_1^j son los valores de la función objetivo del subproblema divisional y las variables de la primera división en la iteración j, respectivamente. Luego, el problema maestro divisional relajado y el subproblema divisional tienen la siguiente expresión:
Problema maestro divisional relajado

$$
\begin{aligned}
&max \quad c_1^T x_1 + \delta \\
&\text{s.a} \\
&\qquad B_1 x_1 \leq b_1 \\
&\qquad \delta \leq g^j + \pi_l^T A_1 (x_1^j - x_1) \quad l = 1, \ldots, j \\
&\qquad x_1 \geq 0
\end{aligned}
\tag{3.9}
$$

Subproblema divisional

$$
\begin{aligned}
g^j = \quad & max \quad c_2^T x_2 \\
& s.a \\
& A_2 x_2 \leq d - A_1 x_1^j \quad : \quad \pi \\
& B_2 x_2 \leq b_2 \qquad\quad\;\; : \quad v \\
& x_2 \geq 0.
\end{aligned}
\tag{3.10}
$$

3.3.1. Cortes de infactibilidad para el problema bi - divisional

Dado que en la práctica no se cumple que para todo $x_1 \in S_1$ el subproblema bi - divisional es factible, es necesario hacer una modificación en el problema maestro divisional para que elimine las propuestas que hacen infactible al subproblema bi - divisional. Esto se logra, como se vió en el Capítulo 2, al añadir al problema maestro cortes de infactibilidad. La construcción de estos cortes se describirá a continiación.

Recordemos que el lema de Farkas nos dice que existe $x \geq 0$, tal que $Ax = b$ si y sólo si $p^T A \geq 0$ implica $p^T b \geq 0$. Luego, al escribir el problema 3.10 en su forma estándar matricial

$$\begin{bmatrix} A_2 & I & 0 \\ B_2 & 0 & I \end{bmatrix} \begin{pmatrix} x_2 \\ h_1 \\ h_2 \end{pmatrix} = \begin{pmatrix} d - A_1 x_1 \\ b_2 \end{pmatrix}$$

este es factible si y sólo si

$$\begin{pmatrix} \pi & v \end{pmatrix} \begin{bmatrix} A_2 & I & 0 \\ B_2 & 0 & I \end{bmatrix} \geq 0 \Rightarrow \begin{pmatrix} \pi & v \end{pmatrix} \begin{pmatrix} d - A_1 x_1 \\ b_2 \end{pmatrix} \geq 0.$$

Si se expresa como problema lineal se obtiene

$$\begin{aligned} &min \quad \pi^T(d - A_1 x_1) + v^T b_2 \\ &\text{s.a} \\ &\qquad A_2^T \pi + B_2^T v \geq 0 \\ &\qquad \pi \geq 0 \\ &\qquad v \geq 0. \end{aligned} \tag{3.11}$$

Si la función objetivo del problema 3.11 alcanza un valor estrictamente negativo, entonces se genera un corte de infactibilidad que excluye los valores de x_1 que verifican

$$\pi^T(d - A_1 x_1) + v^T b_2 \leq 0$$

Sin embargo, el problema 3.11 puede ser no acotado; luego, se introduce una cota superior para las variables, con lo que se obtiene

$$\begin{aligned} &min \quad \pi^T(d - A_1 x_1) + v^T b_2 \\ &\text{s.a} \\ &\qquad A_2^T \pi + B_2^T v \geq 0 && : \quad x_2 \\ &\qquad -\pi \geq -1 && : \quad q_1 \\ &\qquad -v \geq -1 && : \quad q_2 \\ &\qquad \pi \geq 0 \\ &\qquad v \geq 0. \end{aligned} \tag{3.12}$$

El dual del problema 3.12, que se denominará de *suma de infactibilidades*, es:

$$\begin{aligned} &max \quad -(\kappa^T q_1 + \kappa^T q_2) \\ &\text{s.a} \\ &\qquad A_2 x_2 - q_1 \leq d - A_1 x_1 && : \quad \pi \\ &\qquad B_2 x_2 - q_2 \leq b_2 && : \quad v \\ &\qquad x_2, q_1, q_2 \geq 0 \end{aligned} \tag{3.13}$$

donde κ es un vector de unos de dimensión apropiada.
Observemos que $-(\kappa^T q_1 + \kappa^T q_2) \leq 0$, luego el subproblema bi - divisional es factible para un

valor de x_1 si en el óptimo $-(\kappa^T q_1 + \kappa^T q_2) = 0$. En este caso se genera un corte de optimalidad y si $-(\kappa^T q_1 + \kappa^T q_2) < 0$, entonces se genera un corte de infactibilidad.
El problema maestro relajado bi - divisional considerando ambos cortes, de optimalidad e infactibilidad, se define como:

$$\begin{array}{ll} max & c_1^T x_1 + \delta \\ \text{s.a} & \\ & B_1 x_1 \leq b_1 \\ & \gamma^l \delta \leq g^j + \pi_l^T A_1 (x_1^j - x_1) \quad l = 1, \ldots, j \\ & x_1 \geq 0 \end{array} \tag{3.14}$$

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

3.3.2. Algoritmo

El siguiente algoritmo opera de la misma manera que lo hace el algoritmo de Benders, pero para problemas bi - divisionales.

Algoritmo 2 Para encontrar la solución óptima de un problema bi - divisional

Entrada: Un problema bi - divisional

Salida: La solución óptima del problema bi - divisional

1. **Inicialización.** $j = 0$, $\delta^j = \infty$, $\varepsilon = 10^{-4}$.

2. **Resolución del problema maestro divisional.**

$$\begin{array}{ll} max & c_1^T x_1 + \delta \\ \text{s.a} & \\ & B_1 x_1 \leq b_1 \\ & \gamma^l \delta \leq g^j + \pi_l^T A_1 (x_1^j - x_1) \quad l = 1, \ldots, j \\ & x_1 \geq 0 \end{array}$$

   Obtener la solución (x_1^j, δ^j). Mientras no se haya generado ningún corte de optimalidad o infactibilidad, se hace δ igual cero.

3. **Resolución del subproblema divisional de suma de infactibilidades.**

$$\begin{array}{rll} g^j = & max & -(\kappa^T q_1 + \kappa^T q_2) \\ & \text{s.a} & \\ & & A_2 x_2 - q_1 \leq d - A_1 x_1 \quad : \quad \pi \\ & & B_2 x_2 - q_2 \leq b_2 \qquad\quad\; : \quad v \\ & & x_2, q_1, q_2 \geq 0 \end{array}$$

   Si $g^j < 0$, obtener π_j, formar un corte de infactibilidad y si $g^j = 0$, ir al paso 4.

4. **Resolución del subproblema divisional.**

$$\begin{array}{rl} g^j = & max \quad c_2^T x_2 \\ & s.a \\ & A_2 x_2 \leq d - A_1 x_1^j \quad : \quad \pi \\ & B_2 x_2 \leq b_2 \qquad\quad\; : \quad v \\ & x_2 \geq 0 \end{array}$$

   Obtener x_2^j y actualizar $c_2^T x_2^j$.

5. **Criterio de paro.** Si $\delta^j - c_2^j x_2^j \leq \varepsilon$ el algoritmo termina. En otro caso, obtener π_j y g^j, formar un corte de optimalidad e ir al paso 2.

En cada iteración se modifica la región factible del problema bi - divisional relajado (aumenta en uno el número de restricciones) y el valor óptimo del subproblema bi - divisional. El tamaño del problema lineal PL-2 era $(m + m_1 + m_2) \times (n_1 + n_2)$. El tamaño del problema maestro bi - divisional relajado es $(m_1 + j) \times (n_1 + 1)$ y el del subproblema bi - divisional es $(m + m_2) \times n_2$.

3.3.3. Convergencia

En esta sección se demostrará que el algoritmo anterior alcanza el óptimo en un número finito de pasos.

Lema 2. *Si $(\tilde{x}_1, \tilde{x}_2)$ es factible en el problema PL-2 y (x_1^*, δ^*) es óptima en el problema maestro bi - divisional completo,*

$$c_1^T \tilde{x}_1 + c_2^T \tilde{x}_2 \leq c_1^T x_1^* + \delta^*.$$

Demostración.

Sea $(\tilde{x}_1, \tilde{x}_2)$ una solución factible del problema PL-2, entonces $\tilde{x}_2$ es una solución factible del subproblema divisional al poner $\tilde{x}_1$ en lugar de x_1. Por el Teorema 2 de dualidad

$$c_2^T \tilde{x}_2 \leq \text{mín}\left\{(d - A_1\tilde{x}_1)^T \pi_l + b_2^T v_l : l = 1, \ldots, \vartheta\right\}.$$

De esta desigualdad se tiene

$$c_1^T \tilde{x}_1 + c_2^T \tilde{x}_2 \leq c_1^T \tilde{x}_1 + \text{mín}\left\{(d - A_1\tilde{x}_1)^T \pi_l + b_2^T v_l : l = 1, \ldots, \vartheta\right\}.$$

Dado que (x_1^*, δ^*) es solución óptima

$$c_1^T \tilde{x}_1 + \text{mín}\left\{(d - A_1\tilde{x}_1)^T \pi_l + b_2^T v_l : l = 1, \ldots, \vartheta\right\} \leq c_1^T x_1^* + \delta^*$$

de las dos ultimas desigualdades se sigue el lema. □

Lema 3. *Si $x_1^j = x_1^h$ con $j < h$, entonces $\delta^h = c_2^T x_2^h$.*

Demostración.

Supongamos que

$$\delta^j - \varepsilon > c_2^T x_2^j \quad y \quad \delta^h - \varepsilon > c_2^T x_2^h.$$

En la iteración j el corte que se introduce es

$$\begin{aligned}
\delta &\leq g^j + \pi_j^T A_1(x_1^j - x_1) \\
\delta &\leq c_2^T x_2^j + \pi_j^T A_1(x_1^j - x_1) \\
\delta &\leq c_2^T x_2^h + \pi_j^T A_1(x_1^j - x_1)
\end{aligned}$$

pero la solución (x_1^h, δ^h) debe satisfacer la última desigualdad, es decir,

$$\delta^h \leq c_2^T x_2^h + \pi_j^T A_1(x_1^j - x_1^h) = c_2^T x_2^h.$$

Así, tenemos que
$$\delta^h - \varepsilon > c_2^T x_2^h \quad y \quad \delta^h \leq c_2^T x_2^h$$
lo cual es una contradicción. Luego, el lema se verifica. $\square$

El siguiente lema garantiza que en cada iteración la base óptima del subproblema es diferente:

Lema 4. *Si $j \neq h$ y $\delta^h - \varepsilon > c_2^T x_2^h$, entonces $(\pi_j, v_j) \neq (\pi_h, v_h)$.*

Demostración.

Sin pérdida de generalidad, podemos suponer que $j < h$. Supongamos que $(\pi_j, v_j) = (\pi_h, v_h)$, entonces el corte que se introduce al problema maestro divisional relajado en la iteración j es igual al corte que se introduce en la iteración h. Luego, el problema maestro divisional relajado en la iteración $h+1$ reportará la misma información que en la iteración h. Por el Lema 3, $c_2^T x_2^h = \delta^h$, pero por hipótesis $\delta^h - \varepsilon > c_2^T x_2^h$. Entonces, $(\pi_j, v_j) \neq (\pi_h, v_h)$. $\square$

Observación 3.6. *Como la cardinalidad del conjunto Σ es finita y por el Lema 4 la base del subproblema bi - divisional en cada iteración es diferente; entonces $\delta^j = c_2^T x_2^j$ para algún $j < \infty$.*

Esto asegura que el criterio de paro se alcanza en un número finito de pasos.

Teorema 9. *Si en la iteración j, $\delta^j = c_2^T x_2^j$, entonces (x_1^j, x_2^j) es una solución óptima del problema PL-2.*

Demostración.

Sea $(\tilde{x}_1, \tilde{x}_2)$ una solución factible del problema PL-2 y (x_1^*, δ^*) una solución óptima del problema maestro bi - divisional completo. Por el Lema 2 y porque la región factible del problema relajado en la iteración j contiene a la región factible del problema maestro bi - divisional completo, se tiene
$$c_1^T \tilde{x}_1 + c_2^T \tilde{x}_2 \leq c_1^T x_1^* + \delta^* \leq c_1^T x_1^j + \delta^j \leq c_1^T x_1^j + c_2^T x_2^j,$$
lo cual prueba el teorema. $\square$

Ejemplo 3.3. *Consideremos el siguiente problema bi - divisional con dos restricciones acopladas*
$$\begin{aligned} max \quad 2x + y & \\ x & \leq 4 \\ y & \leq 6 \\ x + y & \leq 5 \\ 2x + 3y & \leq 12 \\ & x, y \geq 0 \end{aligned}$$
cuya solución es $(x, y) = (4, 1)$.

Iniciamos con una solución inicial $(x, y) = (0, 4)$, x variable de la primera división e y variable de la segunda división.

Paso 4: El primer subproblema es

$$\begin{aligned} g^1 = \quad & max \quad y \\ & y \leq 5 - 0 = 5 \\ & 3y \leq 12 - 2(0) = 12 \\ & y \leq 6 \\ & y \geq 0, \end{aligned}$$

que al resolverlo da lugar a $y^1 = 4$, $g^1 = 4$, $u_1 = (0, -\frac{1}{3})^T$ y $v_1 = 0$.

Paso 2: El problema maestro es

$$\begin{aligned} & max \quad 2x + \delta \\ & x \leq 4 \\ & \theta - \tfrac{2}{3}x \leq 4 \\ & x \geq 0, \end{aligned}$$

cuya solición óptima es $x^2 = 4$, $\delta^2 = \frac{20}{3}$.

Paso 4: Formulando de nuevo el subproblema para $x = 4$, resulta

$$\begin{aligned} g^2(x) = \quad & max \quad y \\ & y \leq 1 \\ & 3y \leq 4 \\ & y \leq 6 \\ & y \geq 0, \end{aligned}$$

que al resolverlo da lugar a $y^2 = 1$, $g^2 = 1$, $u_2 = (-1, 0)^T$ y $v_2 = 0$.

Paso 2: Formulando nuevamente el problema maestro para $g^2 = 1$ y $u_2 = (-1, 0)^T$

$$\begin{aligned} & max \quad 2x + \delta \\ & x \leq 4 \\ & \delta - \tfrac{2}{3}x \leq 4 \\ & \delta - x \leq 5 \\ & x \geq 0, \end{aligned}$$

que al resolverlo se obtiene la solución óptima $x^3 = 4$, $\delta^3 = 1$. Una vez formulado y resuelto de nuevo el subproblema se obtiene $g^3 = c_2^T y^3 = (1)(1) = 1 = \delta^3$ y, por lo tanto, converge el algoritmo. Hemos encontrado el óptimo del problema original.

Ejemplo 3.4. *Consideremos el siguiente problema divisional con dos restricciones de acople*

$$\begin{array}{rrcll} max & 9x_1 & + & 5x_2+6x_3 & \\ s.a & & & & \\ & 7x_1 & + & 3x_2+4x_3 & \leq 30 \\ & x_1 & + & x_2+2x_3 & \leq 10 \\ & 3x_1 & & & \leq 12 \\ & & & 6x_2+13x_3 & \leq 60 \\ & & & 3x_2+8x_3 & \leq 25 \\ & & & x_1,x_2,x_3 & \geq 0 \end{array} \tag{3.15}$$

cuya solución óptima es $(x_1,x_2,x_3)=(0.72,8.32,0)$.

Iniciamos con una solución inicial $(x_1,x_2,x_3)=(0,2,1)$, x_1 variable de la primera división y x_2,x_3 variables de la segunda división.

Paso 4: El primer subproblema es

$$\begin{array}{rll} g^1= & max & 5x_2+6x_3 \\ \text{s.a} & & \\ & 3x_2+4x_3\leq 60 & \\ & x_2+2x_3\leq 10 & \\ & 6x_2+13x_3\leq 60 & \\ & 3x_2+8x_3\leq 25 & \\ & x_2,x_3\geq 0, & \end{array} \tag{3.16}$$

que al resolverlo da lugar a $x_2^1=\frac{5}{6}, x_3^1=0$, $g^1=41.6$, $u_1=(0,0)^T$ y $v_1=(0,\frac{5}{3})^T$.

Paso 2: El problema maestro es

$$\begin{array}{rl} max & 9x_1+\delta \\ \text{s.a} & \\ & 3x_1\leq 15 \\ & \delta\leq 41.6 \\ & x_1\geq 0, \end{array} \tag{3.17}$$

cuya solución óptima es $x_1^2=4$ y $\delta^2=41.6$.

Paso 4: Formulando de nuevo el subproblema para $x=4$

$$\begin{array}{rll} g^2= & max & 5x_2+6x_3 \\ \text{s.a} & & \\ & 3x_2+4x_3\leq 2 & \\ & x_2+2x_3\leq 6 & \\ & 6x_2+13x_3\leq 60 & \\ & 3x_2+8x_3\leq 25 & \\ & x_2,x_3\geq 0, & \end{array} \tag{3.18}$$

que al resolverlo da lugar a $x_2^2 = \frac{2}{3}, x_3^2 = 0$, $g^2 = \frac{10}{3}$, $u_2 = (\frac{5}{3}, 0)^T$ y $v_2 = (0,0)^T$.

Paso 2: Formulando de nuevo el problema maestro para $g^2 = \frac{10}{3}$ y $u_2 = (\frac{5}{3}, 0)^T$

$$\begin{aligned} max \quad & 9x_1 + \delta \\ \text{s.a} \quad & \\ & 3x_1 \leq 15 \\ & \delta \leq 41.6 \\ & \tfrac{35}{3}x_1 + \delta \leq 50 \\ & x_1 \geq 0, \end{aligned} \tag{3.19}$$

cuya solución óptima es $x_1^3 = 0.72$ y $\delta^3 = 41.6$.

Paso 4: Formulando nuevamente el subproblema para $x_1 = 0.72$

$$\begin{aligned} g^3 = \quad & max \quad 5x_2 + 6x_3 \\ \text{s.a} \quad & \\ & 3x_2 + 4x_3 \leq 24.96 \\ & x_2 + 2x_3 \leq 9.28 \\ & 6x_2 + 13x_3 \leq 60 \\ & 3x_2 + 8x_3 \leq 25 \\ & x_2, x_3 \geq 0, \end{aligned} \tag{3.20}$$

con solución óptima $x_2^3 = 8.32, x_3^2 = 0$ y valor óptimo es $g^3 = 41.6$.
Dado que $\delta^3 - g^3 = 41.6 - 41.6 = 0$, el algoritmo converge, y por lo tanto, hemos encontrado el óptimo del problema original.

.

Capítulo 4

Conclusiones

Al modelar problemas de la ciencia económica e ingeniería, aparecen frecuentemente problemas de programación lineal cuya matriz de coeficientes de restricciones tiene una estructura especial, por lo que los algoritmos que resuelven estos problemas por medio de la interacción de una secuencia de problemas más sencillos merecen ser estudiados.

A continuación presentamos los resultados más relevantes que hemos obtenidos en esta investigación, al aplicar la transferencia del recurso común entre los subproblemas para la optimización de un problema con estructura de bloques y una restricción de acople y al adaptar el algoritmo de Benders a problemas bi - divisional.

- Se diseñó un algoritmo que trabaja con el recurso común y las variables duales (precios sombra) de los subproblemas para encontrar el óptimo del problema cero, vía transferencias del subproblema con menor valor de su variable dual al subproblema con mayor valor en su variable dual.

- Se adaptó el algoritmo de Benders para solucionar un problema bi - divisional.

- Se demostró que los algoritmos que se diseñaron convergen a la solución óptima en número finito de iteraciones.

De las conclusiones anteriores pueden tenerse las siguientes líneas de investigación para el futuro:

1. Diseñar un algoritmo para resolver un problema multi- divisional vía transferencia de los recursos disponibles entre los subproblemas.

2. Adaptar el algoritmo de Benders generalizado para resolver un problema multi- divisional.

.

Referencias

[1] J.F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische mathematik*, 4(1):238–252, 1962.

[2] W.F. Bialas and M.H. Karwan. Mathematical methods for multilevel planning. *The Operations Research Program Department of Industrial Engineering State University of New York At Buffalo. Research Report*, (79-2), 1979.

[3] S. P. Bradley. *Applied Mathematical Programming*. Addison-Wesley Publishing Company, 1977.

[4] A. Charnes, RW Clower, and KO Kortanek. Effective control through coherent decentralization with preemptive goals. *Econometrica: Journal of the Econometric Society*, pages 294–320, 1967.

[5] G.B. Dantzig. *Linear programming and extensions*. Princeton Univ Pr, 1998.

[6] G.B. Dantzig and P. Wolfe. The decomposition algorithm for linear programs. *Econometrica: Journal of the Econometric Society*, pages 767–778, 1961.

[7] A.M. Geoffrion. Elements of large-scale mathematical programming: Part i: Concepts. *Management Science*, pages 652–675, 1970.

[8] A.M. Geoffrion. Generalized benders decomposition. *Journal of optimization theory and applications*, 10(4):237–260, 1972.

[9] RP Harvey. The decomposition principle for linear programs. *International Journal of Computer Mathematics*, 1(1-4):20–35, 1975.

[10] F.S. Hillier, G.J. Lieberman, and M.G. Osuna. *Introducción a la Investigación de Operaciones*, volume 1. McGraw-Hill, 1991.

[11] P. Jennergren. A price schedules decomposition algorithm for linear programming problems. *Econometrica: Journal of the Econometric Society*, pages 965–980, 1973.

[12] A. Kate. Decomposition of linear programs by direct distribution. *Econometrica: Journal of the Econometric Society*, pages 883–898, 1972.

[13] BN Pshenichniy. Linear optimal control problems. *SIAM Journal on Control*, 4:577, 1966.

[14] H.A. Taha. *Investigación de operaciones.* Pearson Educación, 2004.

[15] MN Thapa and GB Dantzig. Linear programming 1: Introduction, 1997.

[16] R.M. Van Slyke and R. Wets. L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics*, pages 638–663, 1969.