



Desigualdades Válidas en Programación con Variables Enteras. Aplicación al Problema de la Mochila

Valid Inequalities in Integer Programming. Application to the Knapsack Problem

Trabajo Fin de Grado en Matemáticas
Universidad de Málaga

Autor: Lourdes Albacete Maza

Área: Estadística e Investigación Operativa. **Departamento:** Análisis Matemático, Estadística e Investigación Operativa y Matemática Aplicada.

Fecha de presentación: Septiembre de 2023

Tema: Optimización Entera. Desigualdades Válidas

Tipo: Trabajo de revisión bibliográfica

Modalidad: Individual

Número de páginas (sin incluir introducción, bibliografía ni anexos): 35

DECLARACIÓN DE ORIGINALIDAD DEL TFG

D./Dña. *Lourdes Albacete Maza*, con DNI [REDACTED] (*DNI, NIE o pasaporte*), estudiante del Grado en *Matemáticas* de la Facultad de Ciencias de la Universidad de Málaga, **DECLARO:**

Que he realizado el Trabajo Fin de Grado titulado “*Desigualdades Válidas en Programación con Variables Enteras. Aplicación al Problema de la Mochila*” y que lo presento para su evaluación. Dicho trabajo es original y todas las fuentes bibliográficas utilizadas para su realización han sido debidamente citadas en el mismo.

De no cumplir con este compromiso, soy consciente de que, de acuerdo con la normativa reguladora de los procesos de evaluación de los aprendizajes del estudiantado de la Universidad de Málaga de 23 de julio de 2019, esto podrá conllevar la calificación de suspenso en la asignatura, sin perjuicio de las responsabilidades disciplinarias en las que pudiera incurrir en caso de plagio.

Para que así conste, firmo la presente en Málaga, el *09/09/2023*

Fdo:.....

Índice general

Resumen	I
Abstract	I
1. Introducción	1
1.1. Formulación de problemas PEM, PE y PEB	2
1.2. Variables binarias como herramienta auxiliar: restricciones disyuntivas . . .	6
1.3. Resolución de PEM, PE y PEB	8
1.3.1. Formulación ideal	13
2. Desigualdades válidas	16
2.1. Definiciones y ejemplos	16
2.2. Desigualdades válidas para problemas enteros	18
2.2.1. Corte por redondeo (Integer Rounding Cut)	18
2.2.2. Cortes de Gomory fraccionales	21
2.2.3. Corte de integralidad en el objetivo	26
2.3. Desigualdades válidas para problemas PEM	27
2.3.1. Corte por Redondeo Entero Mixto (Mixed Integer Rounding)	28
2.3.2. Cortes de Gomory mixtos	30
3. Desigualdades válidas en problemas con cierta estructura. El problema de la mochila	35
3.1. Desigualdades de cubrimiento	35
3.2. Desigualdades de cubrimiento extendidas	37
3.3. Procedimiento “Sequential Lifting”	38
4. Experimentos computacionales	42
4.1. Experimento 1: Problema de la mochila con y sin desigualdades de cubrimiento	43
4.2. Experimento 2: Problema de la mochila con desigualdades de cubrimiento y con desigualdades de cubrimiento extendidas	44
5. Conclusiones	47
A. Tablas	48
Bibliografía	48

Desigualdades Válidas en Programación con Variables Enteras. Aplicación al Problema de la Mochila

Resumen

La programación lineal con variables enteras es una herramienta fundamental en la actualidad debido a su versatilidad en una amplia variedad de disciplinas. Sin embargo, la resolución de este tipo de problemas no es una tarea trivial, puesto que el carácter discreto de sus variables hace que el número de posibles soluciones crezca de forma exponencial. Es por ello que se hace fundamental el uso de herramientas que ayuden a encontrar la solución óptima del problema. En este trabajo nos centraremos en las llamadas desigualdades válidas.

En el Capítulo 1, se presentará la formulación matemática y posibles métodos para resolver tres problemas específicos de programación lineal con variables enteras: Programación Entera (PE); Programación Entera Mixta (PEM); y Programación Entera Binaria (PEB). Además, se introducirá el concepto de formulación ideal de un problema de programación lineal con variables enteras motivando así la idea de desigualdad válida.

El Capítulo 2 comienza con la definición rigurosa de desigualdad válida. A continuación, se presentarán cortes específicos para problemas (PE) y (PEM). En el caso entero puro nos centraremos en tres tipos de desigualdades: corte por redondeo, usualmente llamado *integer rounding cut* por su nombre en inglés; corte de Gomory fraccional; y corte de integralidad en el objetivo. Los cortes que presentaremos en el caso entero mixto serán el corte por redondeo entero mixto, que en inglés se conoce como *mixed integer cut*, y los cortes de Gomory mixtos.

En el Capítulo 3, determinaremos desigualdades válidas para un caso particular de problema (PEB), el problema de la mochila. El objetivo es encontrar desigualdades válidas que sean facetas de la envolvente convexa de la región factible de dicho problema.

En el Capítulo 4, llevaremos a cabo varios experimentos computacionales en los que analizaremos la eficacia de los cortes específicos para el problema de la mochila descritos en el Capítulo 3.

En el Capítulo 5, se resumirán las conclusiones de este trabajo.

Palabras clave:

PROGRAMACIÓN LINEAL ENTERA, DESIGUALDAD VÁLIDA, CORTE, FORMULACIÓN IDEAL, PROBLEMA DE LA MOCHILA.

Valid Inequalities in Integer Programming. Application to the Knapsack Problem

Abstract

Integer linear programming is a fundamental tool nowadays due to its versatility in a wide range of disciplines. However, solving these types of problems is not a trivial task since the discrete nature of its variables causes the number of possible solutions to grow exponentially. Therefore, the use of tools to help us find the optimal solution to the problem becomes essential. In this work, we will focus on the so-called valid inequalities.

Chapter 1 will present the mathematical formulation and possible methods to solve three specific problems of linear programming with integer variables: Integer Programming (IP), Mixed Integer Programming (MIP), and Binary Integer Programming (BIP). Additionally, we will introduce the concept of an ideal formulation of a linear programming problem with integer variables, thus motivating the idea of valid inequalities.

Chapter 2 begins with the rigorous definition of valid inequality. Next, specific cuts for IP and MIP problems will be presented. In the pure integer case, we will focus on three types of inequalities: integer rounding cut, fractional Gomory cut, and objective integrality cut. The cuts presented for the mixed integer case will be mixed integer cut and mixed Gomory cuts.

In Chapter 3, we will determine the valid inequalities for a particular case of the Binary Integer Programming problem, the knapsack problem. The objective is to find valid inequalities that are facets of the convex hull of the feasible region of this problem.

In Chapter 4 various computational experiments have been analyzed to test the effectiveness of the specific cuts for the knapsack problem described in Chapter 3.

Finally, in Chapter 5, the conclusions of this work have been summarized.

key words:

INTEGER LINEAR PROGRAMMING, VALID INEQUALITY, CUT, IDEAL FORMULATION, KNAPSACK PROBLEM.

Capítulo 1

Introducción

El desarrollo de la programación lineal ha sido uno de los avances científicos más importantes de mediados del siglo XX. Una de las razones de dicha relevancia es el hecho de ser una herramienta con numerables aplicaciones, desde minimizar el coste de la fabricación de cualquier producto hasta maximizar los beneficios de grandes empresas aprovechando sus recursos de forma óptima. Muchas son ya las compañías que se han beneficiado del uso de la programación lineal, [11].

Todo problema de programación lineal (PL) consiste en maximizar una función, a la que llamaremos función objetivo, en el que intervienen variables de decisión que están sujetas a ciertas restricciones. Tanto la función objetivo como las restricciones son funciones lineales. Los problemas de este tipo tienen la siguiente forma:

$$\begin{array}{ll} \max_x & c'x \\ \text{s.a.} & Ax \leq b \\ & x \geq 0 \end{array} \quad (\text{PL})$$

donde $x \in \mathbb{R}^n$ es el vector de variables de decisión, $A \in \mathcal{M}_{m \times n}(\mathbb{R})$ es la matriz de coeficientes de las restricciones, $b \in \mathbb{R}^m$ es el vector de términos independientes de dichas restricciones y $c \in \mathbb{R}^n$ es el vector de coeficientes de la función objetivo. Es decir, nos encontramos ante un problema de optimización con n variables y m restricciones. Además, el símbolo $'$ en c' denota el vector traspuesto c . Si no se indica lo contrario, por defecto asumiremos que todos los vectores son vectores columna.

Sin pérdida de generalidad, durante todo el trabajo, supondremos que los problemas de optimización que consideramos son de la forma maximizar una cierta función objetivo $f(x)$. Los problemas de minimización también se pueden expresar con esta opción simplemente afirmando que $\min f(x)$ es equivalente a $\max -f(x)$.

En muchos problemas de la vida real, las variables de decisión sólo tienen sentido si su valor es un número entero. Con frecuencia es necesario asignar personas, máquinas o vehículos, en cantidades enteras, a las actividades que influyen en la función objetivo que queremos estudiar, [11]. Por ejemplo, si tenemos que decidir el número de personas que intervienen en cierta actividad, debemos elegir entre una, dos, tres, ... personas. No tendría sentido elegir 1.5 personas.

Si en el problema lineal (PL), consideramos que las variables de decisión x toman valores enteros en lugar de valores reales, entonces diremos que estamos ante un problema de programación entera (PE). Un caso particular muy importante de programación entera son aquellos problemas que involucran solo dos posibles decisiones: Sí o No. Por ejemplo, ¿debería una determinada empresa contratar más personal? ¿debe emprenderse un proyecto específico? Con sólo dos posibilidades, este tipo de decisiones se pueden representar

mediante variables que sólo toman dos valores, 0 o 1. Las variables de este tipo se llaman binarias y a este tipo de problemas, problemas de programación entera binaria (PEB), [11].

Existen, además, una gran variedad de problemas donde las variables de decisión pueden tomar tanto valores reales como enteros. Por ejemplo, imaginemos que queremos abrir dos nuevos hospitales en Andalucía, de forma que toda la población esté cubierta, es decir, que todos los ciudadanos puedan asistir a alguno de los dos hospitales disponibles. Además, queremos que cada posible paciente vaya al hospital más cercano. Para ello, elegimos minimizar la suma de las distancias de los pacientes al correspondiente hospital. Por simplicidad, asumiremos que las posibles localizaciones son cada una de las ocho capitales andaluzas. Es decir, habría ocho variables enteras (binarias) asociadas a este caso que contestarían a la pregunta: ¿abro un hospital en la capital i , $i = 1, \dots, 8$? Las únicas posibles respuestas son Sí o No. Es decir, 1 o 0, respectivamente. Por otro lado, las variables continuas de este problema indicarán la fracción de la población que se asigna a cada hospital.

Este tipo de modelos, en los que las variables pueden tomar valores reales o enteros, se conocen como problemas de programación entera mixta (PEM). En la siguiente sección formularemos con detalle estos tres tipos de problemas PE, PEB y PEM.

1.1. Formulación de problemas PEM, PE y PEB

Comenzamos la sección dando la formulación de un problema de programación entera mixta (PEM). Estos se pueden expresar de la forma:

$$\begin{aligned} \max_{x,y} \quad & c'x + h'y \\ \text{s.a.} \quad & Ax + Gy \leq b \\ & x \in \mathbb{R}_+^n \\ & y \in \mathbb{Z}_+^p \end{aligned} \tag{PEM}$$

donde $A \in \mathcal{M}_{m \times n}(\mathbb{R})$ y $G \in \mathcal{M}_{m \times p}(\mathbb{R})$ son las matrices de coeficientes de las restricciones del problema, $b \in \mathbb{R}^m$ es el vector de términos independientes, $h \in \mathbb{R}^p$ y $c \in \mathbb{R}^n$ son los vectores de coeficientes de la función objetivo, el vector de variables de decisión con valores reales lo denotamos por $x \in \mathbb{R}_+^n$ e $y \in \mathbb{Z}_+^p$ es el vector de valores enteros, [24]. Nos centraremos en los problemas con función objetivo y restricciones lineales. Para el caso no lineal se recomienda consultar el capítulo 13 de [11].

Nótese que, si tanto las componentes del vector h como de la matriz G son todas nulas, el problema se reduce a un problema como el que aparece en (PL). De la misma forma, si el vector c y la matriz A son nulos entonces el problema se convierte en un problema PE. Luego, la formulación de un problema PE quedaría de la siguiente forma:

$$\begin{aligned} \max_y \quad & h'y \\ \text{s.a.} \quad & Gy \leq b \\ & y \in \mathbb{Z}_+^p \end{aligned} \tag{PE}$$

Para el caso de un problema PEB, como comentamos anteriormente, estos responden a preguntas con solo dos posibles respuestas. Así, la i -ésima variable binaria de decisión se puede representar como:

$$y_i = \begin{cases} 1, & \text{si la decisión } i \text{ es sí} \\ 0, & \text{si la decisión } i \text{ es no} \end{cases}$$

De esta forma, la formulación que usaremos para estos problemas es la siguiente, [24]:

$$\begin{aligned} & \underset{y}{\text{máx}} && h'y \\ \text{s.a.} &&& Gy \leq b \\ &&& y \in \{0, 1\}^p \end{aligned} \tag{PEB}$$

Observación 1.1. Como se puede apreciar, esta formulación es de un problema PEB puro. También existen problemas PEB mixtos donde solo algunas variables del problema son binarias.

A continuación, veamos un ejemplo de un problema de optimización conocido como problema de localización y transporte, [8]. Es un ejemplo de problema (PEM) donde, además, las variables enteras que aparecen son binarias:

Ejemplo 1.1. Una ciudad tiene diez distritos. Cada uno de ellos genera una determinada cantidad de basura (en toneladas). Para cada distrito $i = 1, \dots, 10$, llamaremos A_i a las toneladas de basura generada por el distrito i .

La basura generada debe ser transportada a vertederos entre un total de cinco candidatos posibles, cada vertedero j tiene un costo fijo de construcción (en millones de euros) que llamaremos h_j .

Adicionalmente, se ha estimado el coste de transportar una tonelada de basura desde un distrito cualquiera a cada uno de los vertederos. De esta forma, llamaremos t_{ij} al coste de transportar una tonelada de basura desde el distrito i al vertedero j . Definimos también el parámetro c_j como la capacidad de almacenamiento de basura en toneladas del vertedero j .

A continuación, definimos las variables de decisión. Para $j = 1, \dots, 5$, las variables binarias y_j determinarán si se construye el vertedero j . En particular,

$$y_j = \begin{cases} 1, & \text{si se construye el vertedero } j \\ 0, & \text{si no se construye el vertedero } j \end{cases}$$

Las variables continuas x_{ij} con $i = 1, \dots, 10$ indicarán las toneladas de basura transportadas desde el distrito i al depósito j .

Queremos minimizar el coste total de transportar toda la basura generada por los diez distritos y la función objetivo que modela dicho coste total se puede expresar de la siguiente forma:

$$\sum_{i=1}^{10} \sum_{j=1}^5 t_{ij} x_{ij} + \sum_{j=1}^5 h_j y_j.$$

El primer término $\sum_{i=1}^{10} \sum_{j=1}^5 t_{ij} x_{ij}$ hace referencia al coste de transporte y el segundo término

$\sum_{j=1}^5 h_j y_j$ indica el coste de construcción.

Por otro lado, se debe transportar la totalidad de la basura que genera cada zona. Esta restricción se puede modelar de la siguiente forma:

$$\sum_{j=1}^5 x_{ij} = A_i, \quad i = 1, \dots, 10.$$

Además, se debe respetar la capacidad de almacenamiento de basura para cada vertedero, utilizándolo sólo en el caso de que se decida su construcción. De esta forma, llegamos a la siguiente restricción:

$$\sum_{i=1}^{10} x_{ij} \leq c_j y_j, \quad j = 1, \dots, 5.$$

Por lo tanto, el problema quedaría de la siguiente forma:

$$\begin{aligned} \max_{x,y} \quad & - \sum_{i=1}^{10} \sum_{j=1}^5 t_{ij} x_{ij} - \sum_{j=1}^5 h_j y_j \\ \text{s.a.} \quad & \sum_{j=1}^5 x_{ij} = A_i, \quad i = 1, \dots, 10 \\ & \sum_{i=1}^{10} x_{ij} \leq c_j y_j, \quad j = 1, \dots, 5 \\ & x_{ij} \geq 0, \quad i = 1, \dots, 10, \quad j = 1, \dots, 5 \\ & y_j \in \{0, 1\}, \quad j = 1, \dots, 5 \end{aligned} \tag{1.1}$$

A continuación veremos también un ejemplo de problema de localización pero en este caso, la formulación que utilizaremos es la de un problema (PE), [21].

Ejemplo 1.2. Una empresa de automóviles dispone de tres fábricas, A, B y C. Por otro lado, la empresa cuenta con dos centros de distribución, D1 y D2. Las capacidades de producción de las tres fábricas durante un año son 1000, 1500 y 1200 vehículos, respectivamente. Además, las demandas en los centros de distribución D1 y D2 son de 2300 y 1400 vehículos, respectivamente.

Adicionalmente, se debe tener en cuenta que existe un coste de transporte distinto de cada fábrica a cada centro de distribución. De esta forma, llamaremos d_{ij} al coste de transporte de los vehículos desde la fábrica i al centro de distribución j .

Si queremos minimizar el coste de transporte de los vehículos satisfaciendo todas las restricciones de capacidad y demanda anteriormente descritas, ¿cuántos vehículos deben fabricarse en cada fábrica y a qué centro de distribución deberían mandarlos?

Empezamos definiendo las variables de decisión del problema. Definimos y_{ij} , para $i = 1, 2, 3$ y $j = 1, 2$, como el número de vehículos a transportar de la fábrica i hasta el centro de distribución j .

Las restricciones del problema en cuanto a la capacidad de producción de las fábricas se modelan de la siguiente forma:

$$y_{11} + y_{12} \leq 1000, \quad y_{21} + y_{22} \leq 1500, \quad y_{31} + y_{32} \leq 1200.$$

Por otro lado, las restricciones del problema en relación con la demanda de cada centro de distribución se pueden escribir como:

$$y_{11} + y_{21} + y_{31} \geq 2300, \quad y_{12} + y_{22} + y_{32} \geq 1400.$$

El objetivo principal del problema es minimizar el coste de transportar todos los vehículos demandados, teniendo en cuenta la fábrica de la que salen y el centro de distribución al que llegan. De esta forma, la función objetivo del problema es la siguiente:

$$\sum_{i=1}^3 \sum_{j=1}^2 d_{ij} y_{ij}.$$

Finalmente, el modelo se escribiría como sigue:

$$\begin{aligned}
 \underset{y}{\text{máx}} \quad & - \sum_{i=1}^3 \sum_{j=1}^2 d_{ij} y_{ij} \\
 \text{s.a.} \quad & y_{11} + y_{12} \leq 1000 \\
 & y_{21} + y_{22} \leq 1500 \\
 & y_{31} + y_{32} \leq 1200 \\
 & y_{11} + y_{21} + y_{31} \geq 2300 \\
 & y_{12} + y_{22} + y_{32} \geq 1400 \\
 & y_{ij} \in \mathbb{Z}_+, \quad i = 1, 2, 3, \quad j = 1, 2
 \end{aligned} \tag{1.2}$$

Para finalizar esta sección, veamos un ejemplo de un problema de programación utilizando la formulación (PEB) descrita anteriormente:

Ejemplo 1.3. Problema de la Mochila 0-1, [24]:

Asumiremos que tenemos una mochila con una capacidad limitada, b . Asumiremos también que disponemos de p objetos que podemos introducir en la mochila. Para cada objeto $i = 1, \dots, p$, denotamos por a_i al espacio que ocupa el objeto i en la mochila. Asimismo, c_i será el beneficio asociado al objeto i . Además, trabajamos con la hipótesis de que si el objeto i se introduce en la mochila, éste debe introducirse en su totalidad. Es decir, no podemos dividir el objeto en varias partes.

El objetivo del problema es elegir un conjunto de objetos que no exceda la capacidad máxima y maximice el beneficio total.

Empezamos definiendo las variables binarias y_i que determinarán si introducimos el objeto i en la mochila o no. En particular, para $i = 1, \dots, p$,

$$y_i = \begin{cases} 1, & \text{si se elige el objeto } i \\ 0, & \text{si no se elige el objeto } i \end{cases}$$

La restricción que indica que no puede superarse la capacidad máxima b , puede modelarse de la siguiente forma:

$$\sum_{i=1}^p a_i y_i \leq b$$

Dicho de otro modo, la capacidad de los elementos incluidos en la mochila es $\sum_{i=1}^p a_i y_i$, y esta cantidad no puede sobrepasar la cantidad máxima, b . De esta forma, si incluimos el objeto i , entonces tendremos que $y_i = 1$, y en la suma anterior añadiríamos el término a_i . Por otro lado, si el objeto i no se incluye en la mochila, la variable binaria asociada es nula, i.e. $y_i = 0$, y en la expresión anterior no sumaríamos nada.

Por último, la función objetivo que modela el beneficio total, que queremos maximizar puede expresarse como:

$$\sum_{i=1}^p c_i y_i$$

Es decir, el problema quedaría de la siguiente forma:

$$\begin{aligned}
 \underset{y}{\text{máx}} \quad & \sum_{i=1}^p c_i y_i \\
 \text{s.a.} \quad & \sum_{i=1}^p a_i y_i \leq b \\
 & y_i \in \{0, 1\}, \quad i = 1, \dots, p.
 \end{aligned} \tag{1.3}$$

1.2. Variables binarias como herramienta auxiliar: restricciones disyuntivas

Existen muchas situaciones donde las variables enteras, o en particular, las variables binarias no forman parte del modelo original, sino que se utilizan como herramienta auxiliar. En esta sección, nos centraremos en un caso particular: las restricciones disyuntivas, [24]. Para otras alternativas, como el uso de las variables binarias en las funciones definidas a trozos, se recomienda consultar [24].

Dado un problema de programación lineal (PL) está claro que para que el problema sea factible deben satisfacerse todas las restricciones expresadas en la región factible del mismo, $\{x \geq 0 : Ax \leq b\}$.

Sin embargo, existen problemas de optimización en los que sólo se exige que se verifique k de m restricciones. Por simplicidad, asumiremos que $k = 1$, y $m = 2$. Es decir, estamos interesados en que se cumpla una restricción de las dos disponibles. También a modo ilustrativo, asumiremos que consideramos un problema lineal con variables continuas al que le hemos añadido restricciones disyuntivas lineales. De esta forma, el modelo quedaría así:

$$\begin{array}{ll} \underset{x}{\text{máx}} & c'x \end{array} \quad \text{(PD.1)}$$

$$\text{s.a.} \quad Ax \leq b \quad \text{(PD.2)}$$

$$x \geq 0 \quad \text{(PD.3)}$$

$$d'x \leq d_0 \text{ ó } h'x \leq h_0 \quad \text{(PD.4)}$$

donde la ecuación (PD.4) hace referencia a las restricciones disyuntivas.

Es decir, para que el problema (PD) sea factible, basta con que o bien se verifique $d'x \leq d_0$ o bien se cumpla $h'x \leq h_0$, además, por supuesto, debe de satisfacerse que x sea no negativa y que $Ax \leq b$.

El problema de optimización (PD) no puede resolverse de forma exacta tal y como está formulado actualmente. Es por ello que se recurre a una formulación alternativa pero equivalente en la que se introduce una nueva variable binaria, $y \in \{0, 1\}$ y una constante M suficientemente grande.

En particular, la restricción (PD.4) puede escribirse como:

$$\begin{aligned} d'x - d_0 &\leq M(1 - y) \\ h'x - h_0 &\leq My \end{aligned}$$

De esta forma, se tiene que si $y = 1$, entonces:

$$\begin{aligned} d'x - d_0 &\leq 0 \\ h'x - h_0 &\leq M \end{aligned}$$

Es decir, la restricción $d'x \leq d_0$ debe cumplirse, mientras que la condición $h'x - h_0 \leq M$ no aporta ninguna información, pues sólo está indicando que $h'x - h_0$ debe ser una cantidad menor o igual que un número muy grande.

Análogamente, si $y = 0$, entonces tenemos que:

$$\begin{aligned} d'x - d_0 &\leq M \\ h'x - h_0 &\leq 0 \end{aligned}$$

En otras palabras, en este caso, la restricción que debe satisfacerse es $h'x \leq h_0$.

Por lo tanto, usando esta técnica el problema original (PD) se transformaría en:

$$\begin{array}{ll}
 \text{máx}_{x,y} & c'x \\
 \text{s.a.} & Ax \leq b \\
 & x \geq 0 \\
 & d'x - d_0 \leq M(1 - y) \\
 & h'x - h_0 \leq My \\
 & y \in \{0, 1\}
 \end{array} \tag{PD*}$$

Cómo obtener el valor de M no es una cuestión trivial. De hecho, valores pequeños de M dan lugar a regiones factibles más ajustadas, que simplifican la resolución del problema. Sin embargo, si el valor de M es demasiado pequeño, entonces podríamos obtener problemas infactibles.

Por otro lado, los valores grandes de M disminuyen las probabilidades de tener problemas infactibles a costa de incrementar el tiempo computacional necesario para resolver el problema. En este trabajo, asumiremos conocido el valor de M . Para conocer más información sobre cómo acotar dicho valor M , se recomienda consultar, [20; 16; 13].

Veamos un ejemplo de un problema (PE), en particular un problema PEB mixto, donde se utilizan las variables binarias como herramienta auxiliar para modelar restricciones disyuntivas, [23]:

Ejemplo 1.4. Una empresa de coches está considerando la fabricación de tres tipos nuevos de vehículos: compactos, tamaño mediano y monovolúmenes. La empresa quiere conocer qué tipo de vehículos debe fabricar y cuántos para maximizar los beneficios, teniendo en cuenta que, según ha detectado una comisión de expertos, un nuevo modelo solo resulta económicamente viable si se fabrican al menos 1000 unidades. El material necesario, el tiempo de fabricación y los beneficios esperados (expresado en miles de euros), para cada tipo de vehículo, se especifican en la Tabla 1.1. Además, el total del material utilizado no puede superar las 6000 toneladas y el tiempo total que la empresa puede emplear en la fabricación de los vehículos es de 60000 horas.

	Compactos	Medianos	Monovolúmenes	Máximo permitido
Material (toneladas)	1.5	3	5	6000
Trabajo (horas)	30	25	40	60000
Beneficios (miles de €)	2	3	4	

Tabla 1.1: Datos del Ejemplo 1.4

Empezamos definiendo las variables $x_i \in \mathbb{Z}_+$, para $i = 1, 2, 3$:

x_1 = número de vehículos compactos fabricados

x_2 = número de vehículos medianos fabricados

x_3 = número de monovolúmenes fabricados

Dado que $x_i \in \mathbb{Z}_+, i = 1, 2, 3$, la restricción “*O se fabrican 1000 unidades del modelo elegido o no se fabrica ninguna*” puede modelarse de la siguiente forma:

$$x_i \geq 1000 \text{ ó } x_i \leq 0, \quad i = 1, 2, 3$$

Por lo tanto, el modelo original quedaría:

$$\begin{aligned} \max_{x,y} \quad & 2x_1 + 3x_2 + 4x_3 \\ \text{s.a.} \quad & 1.5x_1 + 3x_2 + 5x_3 \leq 6000 \\ & 30x_1 + 25x_2 + 40x_3 \leq 60000 \\ & x_i \geq 1000 \quad \text{ó} \quad x_i \leq 0 \\ & x_i \in \mathbb{Z}_+, \quad i = 1, 2, 3 \end{aligned}$$

Para poder resolver este problema, introducimos tres variables binarias $y_i \in \{0, 1\}$ $i = 1, 2, 3$, de forma que, para un valor M dado, tenemos:

$$\begin{aligned} 1000 - x_i &\leq M(1 - y_i) \\ x_i &\leq My_i \end{aligned}$$

Luego, el nuevo problema de optimización quedaría:

$$\begin{aligned} \max_{x,y} \quad & 2x_1 + 3x_2 + 4x_3 \\ \text{s.a.} \quad & 1.5x_1 + 3x_2 + 5x_3 \leq 6000 \\ & 30x_1 + 25x_2 + 40x_3 \leq 60000 \\ & 1000 - x_i \leq M(1 - y_i), \quad i = 1, 2, 3 \\ & x_i \leq My_i, \quad i = 1, 2, 3 \\ & x_i \in \mathbb{Z}_+, \quad i = 1, 2, 3 \\ & y_i \in \{0, 1\}, \quad i = 1, 2, 3 \end{aligned} \tag{1.5}$$

1.3. Resolución de PEM, PE y PEB

Resolver un problema de optimización con variables enteras, ya sea (PEM), (PE) o (PEB), puede parecer sencillo pues, a priori, la única diferencia que tienen estos con los de programación lineal (PL) es la restricción sobre integridad de las variables y, por lo tanto, se tienen muchas menos soluciones factibles que considerar. Sin embargo, incluso en el caso (PEB) donde tenemos variables binarias y, por lo tanto, un número finito de soluciones, no es posible garantizar que el problema se pueda resolver de forma sencilla. En particular, si el problema de optimización tiene p variables binarias, entonces tendríamos 2^p soluciones que considerar (aunque algunas de ellas se pueden descartar por no cumplir las demás restricciones). Es decir, tendríamos que resolver 2^p problemas (PL), fijando un valor de cada variable binaria, 0 o 1, previamente. Entonces, cada vez que p aumenta en una unidad, el número de soluciones se duplica. Por ejemplo, tal y como puede apreciarse en la Tabla 1.2, si $p = 8$ entonces se tendrían que considerar 256 soluciones; con $p = 16$, son más de cincuenta mil; con $p = 32$ tenemos más de mil millones, y así sucesivamente; por eso, resulta computacionalmente inviable, [11].

p	2^p
8	256
16	6.55×10^4
32	4.29×10^9

Tabla 1.2: Ilustración de cómo aumenta el número de problemas lineales a resolver en función del número de variables binarias del problema original.

Además, existen problemas (PEB) en los cuales se puede reducir considerablemente (por infactibilidad de las soluciones) el número de problemas (PL) que se deberían resolver una vez fijado el número de variables binarias. Sin embargo, aún así, las cifras son intratables. Un ejemplo de ello es el caso de el conocido como problema del viajante (o por sus siglas en inglés TSP, Traveling Salesman Problem), [7]. A priori, en este problema habría que resolver 2^p problemas lineales tal y como se ha comentado anteriormente, siendo p el número de variables binarias. En el caso del TSP con n nodos, tenemos además que $p = 2^n$, es decir, que el número de variables binarias, se puede expresar como una potencia del número de nodos en la red. Luego, en principio el número de problemas lineales que deberíamos de resolver, en función del número de nodos sería $2^{(2^n)}$. En la Tabla 1.3 se puede apreciar que por ejemplo si consideramos un TSP con 101 nodos, entonces habrá que resolver 1.07×10^{30588} problemas del tipo (PL). Afortunadamente, se ha probado que muchos de esos problemas son infactibles. De hecho, tal y como se puede ver en [24] basta resolver $(n-1)!$ problemas de la forma (PL). Esta cantidad, aunque es considerablemente menor que la anterior, también pasa por resolver un gran número de problemas lineales que para valores razonables de n se hace intratable. Como ejemplo, sólo hay que consultar la Tabla 1.3 donde para $n = 101$ habría que resolver $(n-1)! = 9.33 \times 10^{157}$ problemas (PL).

n	$p = 2^n$	$2^p = 2^{(2^n)}$	$(n-1)!$
3	8	256	2
4	16	6.55×10^4	6
5	32	4.29×10^9	24
50	1.13×10^{15}	3.55×10^{301}	6.08×10^{62}
101	1.27×10^{30}	1.07×10^{30588}	9.33×10^{157}
1001	1.07×10^{301}	$1.07 \times 10^{301.029}$	4.02×10^{2567}

Tabla 1.3: Ilustración de cómo aumenta el número de problemas lineales a resolver en función del número de nodos del problema TSP.

Por otro lado, a veces puede ser tentador resolver el problema (PE) usando las estrategias de resolución de programación lineal. Para ello será necesario definir el concepto de *relajación continua del problema* o *problema relajado*:

Definición 1.1. Sea un problema con variables enteras (PE), (PEM) o (PEB). Llamaremos problema relajado de cualquiera de estos tipos de problemas al modelo resultante de eliminar de su formulación la restricción de integrabilidad de las variables de los mismos. Es decir, sustituiremos la restricción $y_i \in \mathbb{Z}^p$ ó $y_i \in \{0, 1\}^p$, $\forall i$, por la restricción $y_i \in \mathbb{R}^p$ o $y_i \in [0, 1]^p$, respectivamente.

Una vez que hemos definido la relajación del problema, podríamos recurrir a la siguiente estrategia de resolución para problemas con variables enteras. Para ello, consideramos la relajación de (PE), (PEM) o (PEB). De esta forma, el problema original se transforma en un problema (PL).

Cuando tenemos el problema de optimización lineal, podemos encontrar el óptimo (usando por ejemplo el método símplex, cuyos detalles pueden encontrarse en el capítulo 4 de [11]) y redondear los valores no enteros obtenidos. Sin embargo, si usamos esta estrategia podemos encontrarnos con dos problemas asociados: *i*) redondear la solución óptima del problema (PL) resulta en una solución infactible para el problema original con variables enteras, [11]. De hecho, con frecuencia es difícil decidir en qué sentido redondear

para conservar la factibilidad, y *ii*) aún cuando la solución óptima redondeada sea factible, no existe garantía de que esta solución redondeada sea la solución óptima del problema original con variables enteras.

Veamos a continuación los Ejemplos 1.5 y 1.6 donde ilustraremos, respectivamente, los problemas *i*) y *ii*) anteriormente mencionados.

Ejemplo 1.5. Consideramos el siguiente problema (PE):

$$\begin{aligned} \max_y \quad & y_1 + 0.64y_2 \\ \text{s.a.} \quad & 50y_1 + 31y_2 \leq 250 \\ & 3y_1 - 2y_2 \geq -4 \\ & y_1, y_2 \in \mathbb{Z}_+ \end{aligned} \tag{1.6}$$

La solución de este problema relajado es $(\frac{376}{193}, \frac{950}{193}) \simeq (1.94, 4.92)$ que redondeada al entero más cercano se convierte en $(2, 5)$. Es fácil comprobar que el punto $(2, 5)$ no es una solución factible al no cumplir la primera restricción del problema puesto que $50 \cdot 2 + 31 \cdot 5 = 255 \not\leq 250$, [24]. Podemos ver este ejemplo ilustrado en la Figura 1.1a. La Figura 1.1b muestra una imagen detallada de la zona donde se produce la infactibilidad. La zona en gris representa la región factible del problema relajado y el punto $(2, 5)$ anteriormente mencionado está representado con una cruz azul.

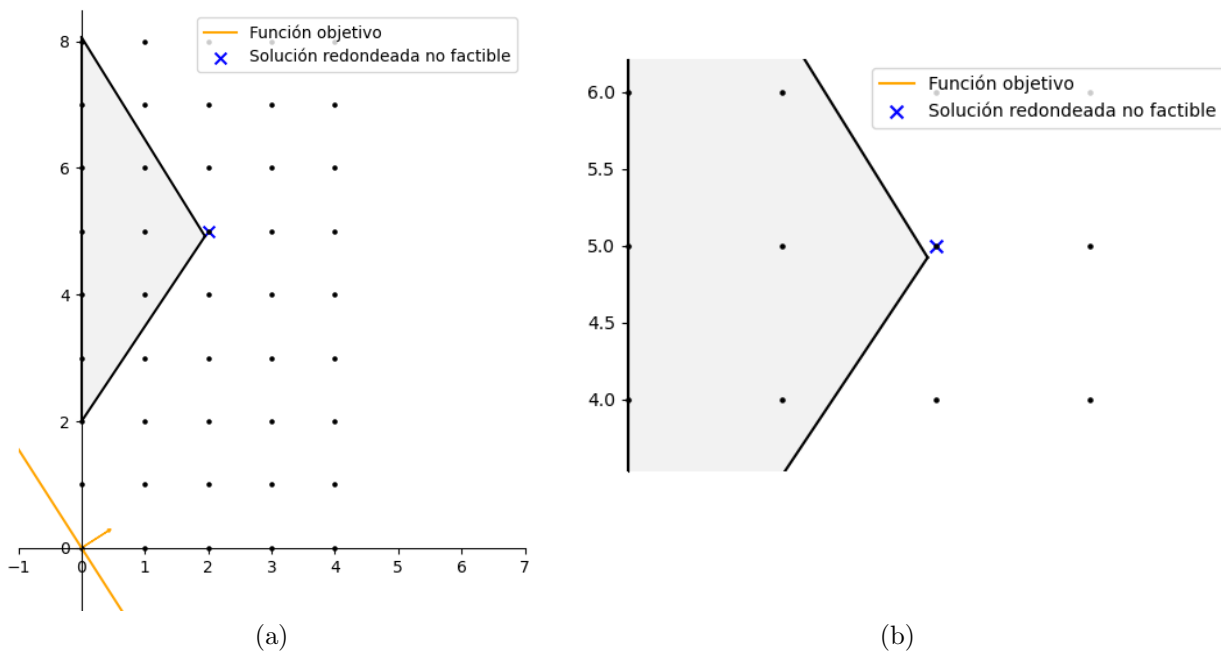


Figura 1.1: Redondear la solución óptima del problema relajado puede dar lugar a un punto infactible. En (a) podemos observar una imagen general de este fenómeno, mientras que (b) representa más detalles del mismo.

Ejemplo 1.6. Considérese el siguiente problema (PE):

$$\begin{aligned} \underset{y}{\text{máx}} \quad & y_1 + 5y_2 \\ \text{s.a.} \quad & y_1 + 10y_2 \leq 20 \\ & y_1 \leq 2 \\ & y_1, y_2 \in \mathbb{Z}_+ \end{aligned} \tag{1.7}$$

La solución de este problema relajado es $(2, \frac{9}{5}) = (2, 1.8)$. La variable y con valor de $\frac{9}{5}$ se redondearía en este caso al punto entero factible más cercano. La solución entera que se obtendría sería $(2, 1)$. Aún siendo esta solución factible, está muy lejos de la solución óptima $(0, 2)$ como se puede observar en la Figura 1.2. Los puntos azules $(0, 2)$, $(2, \frac{9}{5})$ y $(2, 1)$ representan la solución óptima del problema (PE) original, la del problema relajado y la solución obtenida por redondeo, respectivamente. Además la flecha indica la dirección de máximo crecimiento de la función objetivo, [11].

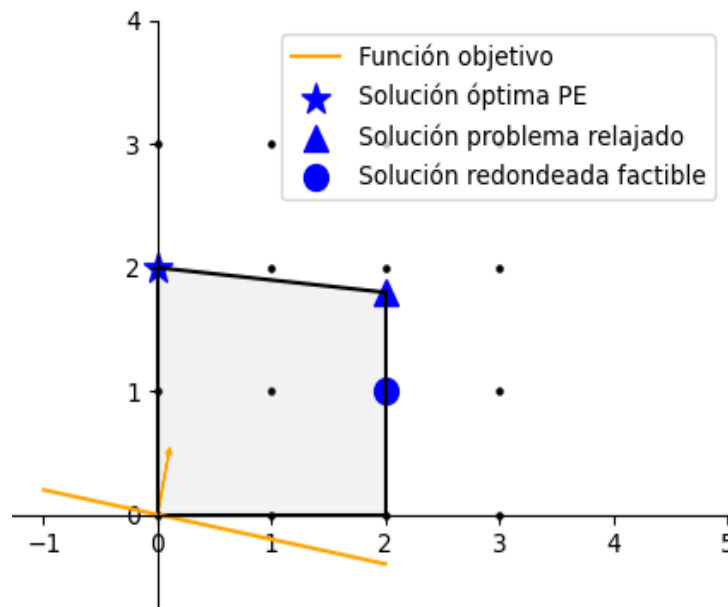


Figura 1.2: Redondear la solución óptima del problema relajado puede dar lugar a una solución subóptima.

Un algoritmo alternativo, y muy utilizado que ha demostrado ser eficiente para resolver un problema (PE), (PEM) y (PEB) es el algoritmo de ramificación y acotación (también conocido por su nombre en inglés, *Branch-and-Bound*), [11]. La idea básica en la que se apoya este algoritmo es *divide y vencerás*. En particular, se construye un árbol donde iterativamente se resuelve el problema original. En el nodo raíz, se resuelve el problema relajado y, a continuación, se procede a buscar una solución óptima mediante la ramificación iterativa, es decir, imponiendo restricciones lineales que dividen la región factible del problema en subconjuntos y organizando los subproblemas resultantes como nodos en un árbol, [9]. El procedimiento termina cuando cada submodelo ha producido una solución no factible, o se ha demostrado que no contiene una solución mejor que la que ya está disponible. La mejor solución encontrada durante el procedimiento es una solución óptima, [24].

Veamos brevemente con el siguiente ejemplo cómo se realizan las primeras iteraciones del algoritmo de ramificación y acotación, [3].

Ejemplo 1.7. Consideramos el siguiente problema (PE):

$$\begin{aligned}
 \max_y \quad & 5y_1 + 8y_2 \\
 \text{s.a.} \quad & y_1 + y_2 \leq 6 \\
 & 5y_1 + 9y_2 \leq 45 \\
 & y_1, y_2 \in \mathbb{Z}_+
 \end{aligned} \tag{1.8}$$

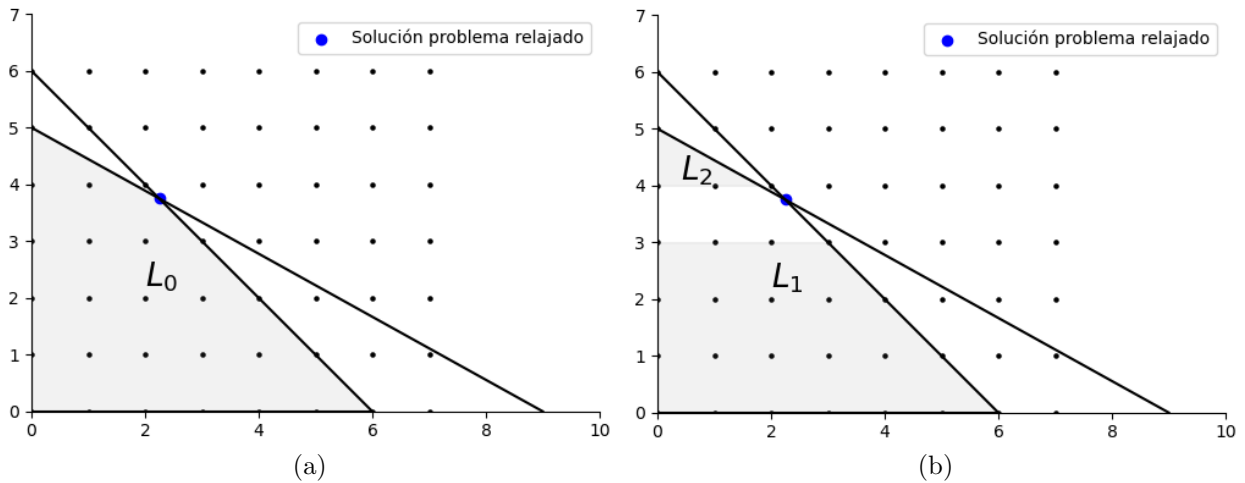


Figura 1.3: En (a) se observa el conjunto L_0 y en (b) aparece L_0 dividido en dos subconjuntos L_2 , en la parte superior, y L_1 , en la parte inferior.

La región gris de la Figura 1.3a forman la región factible del problema relajado original (1.8). Como se puede apreciar en la Figura 1.3a, hemos llamado L_0 a la región factible del problema original (1.8) relajado cuya solución óptima es $(\frac{9}{4}, \frac{15}{4}) = (2.25, 3.75)$, que está representado con un punto azul, y con valor en la función objetivo 41.25.

Ambas variables y_1 e y_2 deben ser enteras en la solución óptima. Para lograr la integridad de las variables, podemos dividir la región factible original del problema relajado en dos subregiones. Sabemos que y_2 , que tiene un valor de 3.75 en la solución óptima del problema relajado, debe ser un número entero menor o igual que 3 o un valor entero mayor o igual que 4. Por lo tanto, la primera división que se hará vendrá dada por las restricciones $y_2 \leq 3$ e $y_2 \geq 4$. Esto da lugar a las regiones L_1 y L_2 de la Figura 1.3b, respectivamente. Obsérvese que, al hacer las divisiones, hemos excluido la antigua solución del problema relajado. Una posible alternativa habría sido hacer las divisiones en función de la variable y_1 , en lugar de con la variable y_2 . Dado que el valor óptimo de y_1 en el problema relajado es 2.25, tendríamos dos nuevas regiones determinadas por las desigualdades $y_1 \leq 2$ e $y_1 \geq 3$, [3].

Determinar sobre qué variable se debe ramificar no es una cuestión trivial. Para conocer más información sobre este tema, se recomienda consultar [2; 12].

Una vez tenemos las subregiones L_1 y L_2 , resolvemos el problema en cada región como hemos hecho anteriormente en L_0 . Por ejemplo, en el caso de L_1 , tendríamos que resolver la relajación del problema (1.8) a la que añadiríamos la restricción $y_2 \geq 4$.

Este proceso se repetirá de forma que resolveremos todos los subproblemas posibles descartando aquellos que lleguen a soluciones no factibles, así como los que la solución encontrada sea peor que la anterior, hasta que encontremos la solución óptima del problema original con variables enteras, [3].

Nótese que el ejemplo anterior es un ejemplo de utilización del algoritmo *Branch and Bound* en un problema (PE). Para problemas (PEM) se recomienda consultar [11].

Todas las alternativas mencionadas anteriormente son metodologías con las que resolver un problema (PE), (PEM) o (PEB). Sin embargo, en una situación ideal en la que todos los puntos extremos de la región factible fuesen valores enteros, bastaría con resolver el problema relajado una única vez utilizando estrategias de programación lineal como el simplex. De esta forma, obtendríamos que la solución óptima del problema relajado está en uno de los vértices de la región factible, que además sabemos que tiene las componentes enteras. Es decir, la solución óptima del problema relajado es equivalente a la del problema original. Sin embargo, encontrar una reformulación equivalente al problema original en la que todos los vértices son enteros no es sencillo. En la siguiente sección, se explicarán algunos conceptos para conseguirlo.

1.3.1. Formulación ideal

Comenzaremos la sección explicando los conceptos de poliedro y formulación. También explicaremos cuándo una formulación es mejor que otra. Estas definiciones serán clave para entender cómo encontrar una reformulación del problema original donde los vértices son enteros. Para más detalles sobre estos conceptos, se recomienda consultar [24].

Definición 1.2. Llamamos poliedro P a un subconjunto de \mathbb{R}^n determinado por un conjunto finito de restricciones lineales. Es decir,

$$P = \{x \in \mathbb{R}^n : Ax \leq b\}.$$

Definición 1.3. Un poliedro $P \subseteq \mathbb{R}^{n+p}$ es una formulación para un conjunto $X \subseteq \mathbb{Z}^p \times \mathbb{R}^n$ si y sólo si $X = P \cap (\mathbb{Z}^p \times \mathbb{R}^n)$.

Definición 1.4. Dado un poliedro P , x es un punto extremo de P si $x_1, x_2 \in P$ con $x = \lambda x_1 + (1 - \lambda)x_2$, $0 \leq \lambda \leq 1$ implica que $x = x_1 = x_2$. Nótese que un punto extremo corresponde a la idea geométrica de vértice.

El siguiente resultado nos indica que la solución óptima de un problema (PL) está siempre en un vértice. La demostración se puede encontrar en el capítulo 2 de [19].

Teorema 1.1. Consideremos un problema de programación lineal de la forma (PL). Si el problema tiene solución óptima finita, entonces el poliedro que define su región factible tiene un punto extremo que es solución óptima del problema.

En la Figura 1.4, se ilustran los distintos conceptos definidos anteriormente. En primer lugar, consideramos el conjunto $X = \{(1, 1), (2, 1), (3, 1), (1, 2), (2, 2), (3, 2), (2, 3)\}$ que podría constituir la región factible de cierto problema de optimización de tipo (PE). Los puntos factibles se encuentran señalados con cruces azules en la Figura 1.4. A continuación, definimos dos poliedros P_1 y P_2 dados por:

$$P_1 = \left\{ (x, y) \in \mathbb{R}^2 : \frac{3}{10} + \frac{7}{4}x \geq y, \frac{77}{10} - \frac{27}{16}x \geq y, \frac{3}{7}x - \frac{4}{7} \geq y, \frac{17}{10} - \frac{13}{16}x \geq y \right\}$$

$$P_2 = \left\{ (x, y) \in \mathbb{R}^2 : 6x - 4 \geq y, y = \frac{16}{5}, \frac{128}{25} - \frac{10x}{13} \geq y, 6x - 20 \geq y, y = \frac{4}{5} \right\}$$

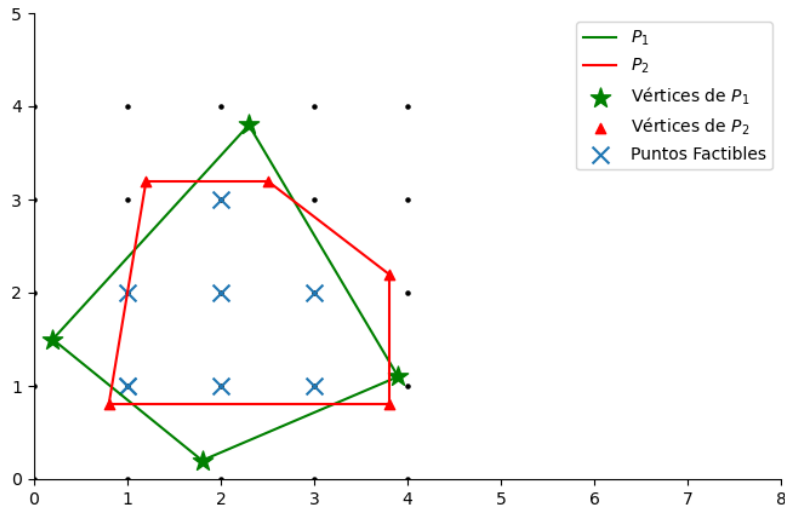


Figura 1.4: Formulaciones P_1 y P_2 para el conjunto X .

y dibujados con línea sólida verde y roja, respectivamente. Además, estos dos poliedros constituyen formulaciones del problema original, pues $X = P_1 \cap \mathbb{Z}^2$ y $X = P_2 \cap \mathbb{Z}^2$. Por último, los puntos señalados con estrellas y triángulos son, respectivamente, los vértices de los poliedros P_1 y P_2 .

En la Figura 1.4, se observan dos formulaciones distintas para el mismo problema de optimización. Así pues, nos podemos plantear la siguiente pregunta: Dadas dos formulaciones distintas P_1 y P_2 para X , ¿podemos decir que una es mejor que otra?

La geometría nos ayuda a encontrar la respuesta. Si nos fijamos en la Figura 1.5 podemos observar que se ha añadido una formulación más, P_3 , a las dos que aparecen en la Figura 1.4. La formulación P_3 es conocida como la formulación ideal, pues si resolvemos un problema (PL) en P_3 , la solución óptima va a ser un vértice (*punto extremo*), tal y como nos dice el Teorema 1.1. Dado que todos los vértices de P_3 son puntos con componentes enteras, entonces el problema (PE) original estaría resuelto. A continuación daremos una definición rigurosa para saber cuándo una formulación es mejor que otra.

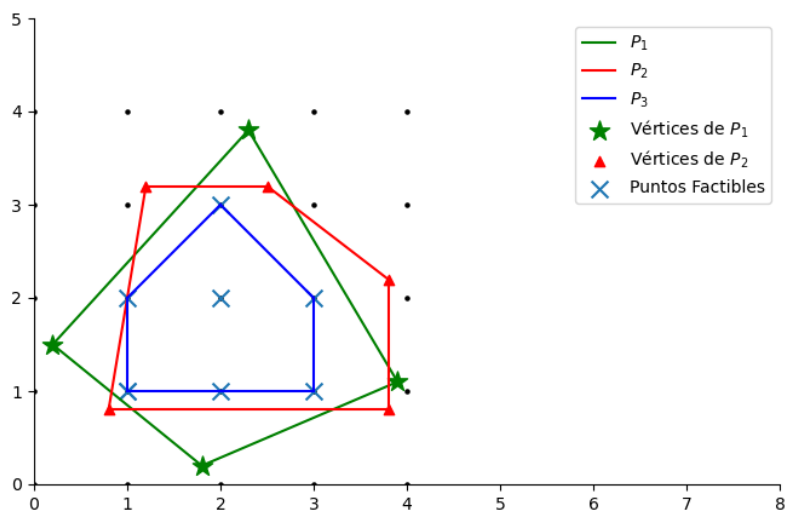


Figura 1.5: Formulaciones P_1 , P_2 y P_3 para el conjunto X .

Definición 1.5. Dado un conjunto $X \subseteq \mathbb{Z}^p$, y dos formulaciones P_1 y P_2 de X , decimos que P_1 es una formulación mejor que P_2 si $P_1 \subset P_2$. Es decir, si el conjunto de puntos definido por P_1 está contenido en el conjunto de puntos generados por P_2 .

Además, a la anteriormente citada como formulación ideal se le conoce con el nombre de envolvente convexa. Formalmente:

Definición 1.6. Dado un conjunto $X \subseteq \mathbb{R}^n$, la envolvente convexa de X , denotada $\text{conv}(X)$, se define como: $\text{conv}(X) = \{x : x = \sum_{i=1}^t \lambda_i x^i, \sum_{i=1}^t \lambda_i = 1, \lambda_i \geq 0, i = 1, \dots, t\}$ para todos los subconjuntos finitos $\{x^1, \dots, x^t\}$ de X .

A continuación, veremos un resultado que afirma que la envolvente convexa es un poliedro.

Proposición 1.1. Si $X = \{x \in \mathbb{Z}^p : Ax \leq b\}$. Entonces, la envolvente convexa de X es un poliedro, [24].

Entonces, dado que si $X \subset \mathbb{Z}^p$ se tiene que $X \subseteq \text{conv}(X) \subseteq P$, para toda P formulación de X , podemos decir que $\text{conv}(X)$ es la mejor formulación posible para X , la ideal, [24].

Por lo tanto, el problema original (PE) dado por:

$$\begin{array}{ll} \underset{x}{\text{máx}} & c'x \\ \text{s.a.} & x \in X \end{array}$$

puede reformularse, en base a los resultados anteriores como un problema equivalente de programación lineal, cuyos vértices son enteros y que tiene la siguiente forma:

$$\begin{array}{ll} \underset{x}{\text{máx}} & c'x \\ \text{s.a.} & x \in \text{conv}(X) \end{array}$$

Sin embargo, en la práctica, no es posible, en general, encontrar la envolvente convexa de X , puesto que equivale a añadir un número exponencial de nuevas restricciones, [24]. Es por ello, que lo que se propone, en su lugar, es aproximar dicha envolvente convexa a través de algunas restricciones conocidas con el nombre de *desigualdades válidas*. Los detalles sobre este concepto, así como su construcción, se verán en el próximo capítulo.

Capítulo 2

Desigualdades válidas

2.1. Definiciones y ejemplos

El objetivo de este capítulo es describir distintas estrategias para aproximar la envolvente convexa de la región factible de un problema (PE), (PEB) o (PEM). El concepto fundamental que ya hemos introducido en el capítulo anterior de manera informal para reducir dicha región factible es el de desigualdad válida. En esta sección formalizaremos esta idea y veremos algunos ejemplos sencillos.

Definición 2.1. Sea $\pi \in \mathbb{R}^n$ y $\pi_0 \in \mathbb{R}$. Una desigualdad $\pi'x \leq \pi_0$ es una desigualdad válida para $X \subseteq \mathbb{R}^n$ si $\pi'x \leq \pi_0$ para todo $x \in X$, [24].

Es decir, una desigualdad será válida si todos los puntos factibles verifican dicha desigualdad. En el siguiente ejemplo, mostraremos algunas restricciones que son desigualdades válidas y otras que no lo son.

Ejemplo 2.1. Consideramos el conjunto $X = \{(x, y) : x \leq 9999y, 0 \leq x \leq 5, y \in \{0, 1\}\}$, representado en la Figura 2.1 con una línea sólida y el punto (0,0) en azul oscuro. Podemos asumir que X constituye la región factible de un problema mixto binario. En ese caso, la zona de color gris describiría la región factible del problema de optimización relajado.

En primer lugar, notaremos que la desigualdad $y \geq \frac{x}{5}$ dibujada con línea discontinua azul es válida, puesto que todos los puntos factibles satisfacen dicha restricción. Además, las restricciones $y \geq \frac{x}{7}$ e $y \leq \frac{x}{5} + 2$ representadas con línea discontinua verde y negra, respectivamente, también son desigualdades válidas por la misma razón mencionada anteriormente. Por otro lado, la restricción, dibujada con línea discontinua roja, y dada por $y \geq \frac{x}{3}$ no es una desigualdad válida para X , ya que como puede observarse, por ejemplo, el punto (5,1) que pertenece a X , no satisface dicha restricción, [24].

Como hemos explicado anteriormente, el objetivo de introducir desigualdades válidas en un problema de optimización con variables enteras es reducir la región factible del problema relajado, y simplificar así su resolución. Es por ello que, de entre todas las posibles desigualdades válidas que podemos incluir, sólo nos interesa tener en cuenta aquellas que eliminen soluciones fraccionarias de tipo (PEM) del problema relajado sin eliminar soluciones enteras. En el caso de problemas mixtos nos referimos solo a eliminar soluciones fraccionarias en aquellas componentes que toman valores enteros en el problema original.

Esta idea nos permite definir el concepto de corte:

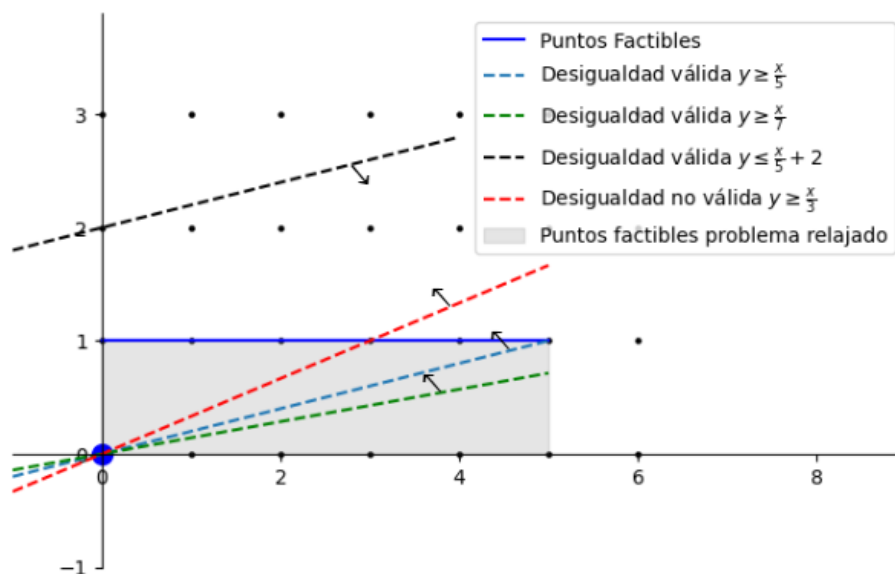


Figura 2.1: Ilustración del Ejemplo 2.1

Definición 2.2. Un plano de corte (o corte) de un problema con variables enteras es una desigualdad válida que reduce la región factible del problema (PL) relajado sin eliminar ninguna solución entera del problema original, [4].

En particular, si nos fijamos en la Figura 2.1 observamos que la restricción $y \leq \frac{x}{5} + 2$ es una desigualdad válida, pero no es un corte, puesto que la región factible del problema relajado se mantiene constante. Sin embargo, las restricciones $y \geq \frac{x}{5}$ e $y \geq \frac{x}{7}$ sí que son cortes puesto que reducen el número de soluciones fraccionarias.

Durante todo el trabajo, y tal y como suele hacerse en la literatura, asumiremos que todas las desigualdades válidas que describimos, serán además cortes. Es por ello que usaremos ambas palabras como sinónimos.

Por otro lado, observamos que en la Figura 2.1 la desigualdad $y \geq \frac{x}{5}$ es *mejor* que la desigualdad $y \geq \frac{x}{7}$ puesto que reduce más la región factible del problema relajado, y en consecuencia la resolución del problema original con variables enteras será más sencilla. De hecho, la restricción $y \geq \frac{x}{5}$ forma parte de la envolvente convexa del poliedro definido por la región factible del problema relajado.

Esto nos lleva a definir cuándo una desigualdad válida es mejor que otra:

Definición 2.3. Sean $\pi, \pi_0, \gamma, \gamma_0 \in \mathbb{R}^n$. Las desigualdades válidas $\pi'x \leq \pi_0$ y $\gamma'x \leq \gamma_0$ se dice que son *equivalentes* si $(\gamma, \gamma_0) = \lambda(\pi, \pi_0)$ para algún $\lambda > 0$. Si no son equivalentes y existe $\mu > 0$ tal que $\gamma \geq \mu\pi$ y $\gamma_0 \leq \mu\pi_0$, entonces $\{\gamma'x \leq \gamma_0\} \subset \{\pi'x \leq \pi_0\}$. En este caso decimos que $\gamma'x \leq \gamma_0$ *domina* o que es *más fuerte* que $\pi'x \leq \pi_0$ o que $\pi'x \leq \pi_0$ está *dominada por* o que es *más débil* que $\gamma'x \leq \gamma_0$. En este sentido, una desigualdad válida es *maximal* si no está dominada por ninguna otra desigualdad válida, [14].

Ateniéndonos a la definición anterior y fijándonos en la Figura 2.1, podemos decir que la desigualdad $y \geq \frac{x}{5}$ domina a la desigualdad válida $y \geq \frac{x}{7}$ o que $y \geq \frac{x}{7}$ es más débil que $y \geq \frac{x}{5}$. Es más, como se puede observar, $y \geq \frac{x}{5}$ es maximal.

Además, la siguiente proposición nos permite afirmar que las desigualdades válidas maximales son las más importantes de todas las posibles, puesto que permite definir la envolvente convexa a través de las llamadas *facetas*.

Antes de enunciar el resultado teórico, daremos la definición de faceta, y de cara de un poliedro necesarias para poder entender el primer concepto.

Definición 2.4. Sean $\pi, \pi_0 \in \mathbb{R}^n$. Si $\pi'x \leq \pi_0$ es una desigualdad válida para un poliedro P y $F = \{x \in P : \pi'x = \pi_0\}$, decimos que F es una cara de P , [14].

Definición 2.5. Decimos que una cara F de un poliedro P es una faceta de P si $\dim(F) = \dim(P) - 1$, [14].

Proposición 2.1. Cualquier desigualdad válida maximal de $X \subseteq \mathbb{R}^n$ define una cara no vacía de $\text{conv}(X)$, y el conjunto formado por todas la desigualdades válidas maximales contiene todas las facetas que definen $\text{conv}(X)$, [14].

A continuación, describiremos algunas desigualdades válidas para problemas enteros.

2.2. Desigualdades válidas para problemas enteros

Durante toda esta sección, asumiremos que conocemos la región factible de un problema (PE).

2.2.1. Corte por redondeo (Integer Rounding Cut)

El primer corte que definiremos para problemas (PE) está basado en un resultado teórico sencillo e intuitivo:

Proposición 2.2. Sea $b \in \mathbb{R}$ y sea $Y = \{y \in \mathbb{Z} : y \leq b\}$, entonces la desigualdad $y \leq \lfloor b \rfloor$ es válida para Y , [24].

Demostración. Procederemos por reducción al absurdo. Supongamos que la desigualdad $y \leq \lfloor b \rfloor$ no es válida para Y . Entonces, existe $y \in Y$ tal que $\lfloor b \rfloor < y \leq b$. Esto es imposible, puesto que no existe ningún número entero que lo verifique. Por lo tanto, $y \leq \lfloor b \rfloor$ es válida para Y . \square

Gracias a ese resultado, podemos describir el conocido como “**Procedimiento Chvátal-Gomory (C-G)**”. A continuación vamos a describir dicho procedimiento para construir un corte sobre el conjunto Y .

Consideramos el conjunto $Y = P \cap \mathbb{Z}^p$ donde $P = \{y \in \mathbb{R}_+^p : Ay \leq b\}$ es un poliedro con $b \in \mathbb{R}^m$ y $A \in \mathcal{M}_{m \times p}(\mathbb{R})$ con columnas $\{a_1, a_2, \dots, a_p\}$. Sea $u \in \mathbb{R}_+^m$ un vector dado, el “Procedimiento Chvátal-Gomory (C-G)” se realiza de la siguiente forma: primero, se construye una nueva desigualdad. Para ello, se multiplica cada restricción definida por P por las componentes de u para posteriormente sumarlas. Es decir, en el primer paso construimos la desigualdad:

$$\sum_{j=1}^p u' a_j y_j \leq u' b. \quad (2.1)$$

Nótese que (2.1) es una desigualdad válida para el conjunto Y puesto que $u \geq 0$ y

$$\sum_{j=1}^p a_j y_j \leq b.$$

A continuación, en el segundo paso, redondeamos hacia abajo al entero más cercano los coeficientes $u'a_j$ que aparecen en la parte izquierda de (2.1) obteniéndose:

$$\sum_{j=1}^p \lfloor u'a_j \rfloor y_j \leq u'b. \quad (2.2)$$

De esta forma, como $y \geq 0$, la restricción (2.2) obtenida tras dicho redondeo es una desigualdad válida para Y .

Por último, fijándonos en (2.2), observamos que como y es un vector con valores enteros, en consecuencia, $\sum_{j=1}^p \lfloor u'a_j \rfloor y_j$ es entero. Aplicando la Proposición 2.2 llegamos a la siguiente desigualdad válida para Y :

$$\sum_{j=1}^p \lfloor u'a_j \rfloor y_j \leq \lfloor u'b \rfloor. \quad (2.3)$$

El pseudocódigo de este procedimiento puede verse en el Algoritmo 1:

Algoritmo 1 Procedimiento Chvátal-Gomory

Entrada: $A = (a_1 | \dots | a_p) \in \mathcal{M}_{m \times p}(\mathbb{R})$, $b \in \mathbb{R}^m$, $u \in \mathbb{R}_+^m$

- 1: $\sum_{j=1}^p u'a_j y_j \leq u'b$
- 2: $\sum_{j=1}^p \lfloor u'a_j \rfloor y_j \leq u'b$
- 3: $\sum_{j=1}^p \lfloor u'a_j \rfloor y_j \leq \lfloor u'b \rfloor$

Salida: Desigualdad $\sum_{j=1}^p \lfloor u'a_j \rfloor y_j \leq \lfloor u'b \rfloor$

A continuación, vamos a ilustrar este proceso con un ejemplo, [14].

Ejemplo 2.2. Sea $Y = P \cap \mathbb{Z}^p$ el conjunto de los puntos discretos en P , donde P es un poliedro que viene dado por las siguientes restricciones:

$$\begin{aligned} -y_1 + 2y_2 &\leq 4 \\ 5y_1 + y_2 &\leq 20 \\ -2y_1 - 2y_2 &\leq -7 \\ y &\geq 0 \end{aligned} \quad (2.4)$$

En la Figura 2.2 se pueden observar los puntos de Y marcados con cruces azules y el poliedro, P , que lo delimita dado por las restricciones de (2.4).

Empezamos haciendo el paso 1 con una combinación de las tres desigualdades con pesos no negativos $u_1 = (\frac{4}{11}, \frac{3}{11}, 0)$. Es decir, $\frac{4}{11} \times (-y_1 + 2y_2) + \frac{3}{11} \times (5y_1 + y_2) + 0 \times (-2y_1 - 2y_2) \leq \frac{4}{11} \times 4 + \frac{3}{11} \times 20 + 0 \times -7$. De esta forma obtenemos la desigualdad válida (2.5) para Y , que viene ilustrada en la Figura 2.2 como una línea discontinua negra:

$$y_1 + y_2 \leq \frac{76}{11}. \quad (2.5)$$

Conforme se explica en el paso 2, como $y \geq 0$, los coeficientes de la parte izquierda de la desigualdad válida obtenida se pueden redondear hacia abajo al entero más cercano llegando en este caso a la misma desigualdad:

$$y_1 + y_2 \leq \frac{76}{11}. \tag{2.6}$$

Una vez que la parte de la izquierda es entera para todos los puntos de Y , podemos hacer lo mismo con la parte de la derecha basándonos en la Proposición 2.2:

$$y_1 + y_2 \leq \left\lfloor \frac{76}{11} \right\rfloor = 6. \tag{2.7}$$

La desigualdad (2.7) obtenida mediante el procedimiento C-G del Algoritmo 1 está representada en la Figura 2.2 con un línea discontinua roja.

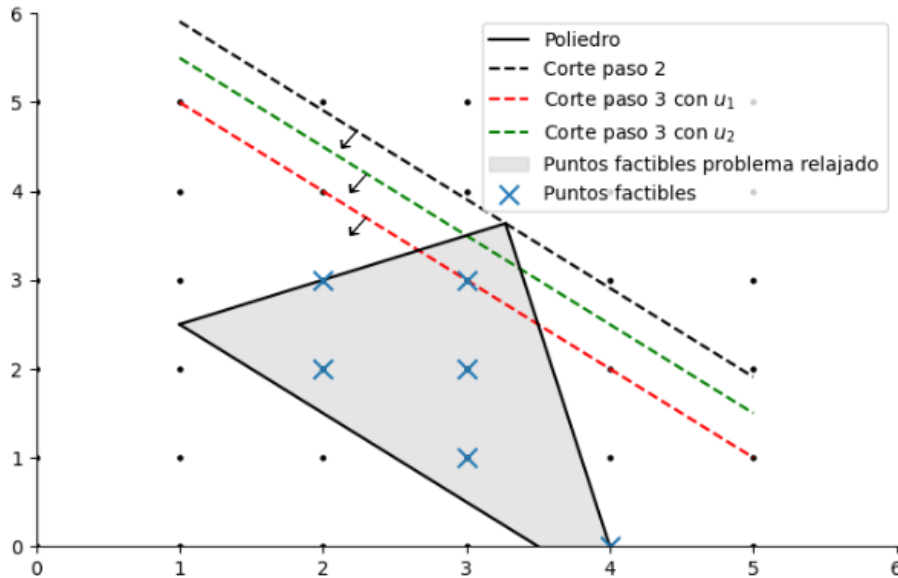


Figura 2.2: Ilustración del Ejemplo 2.2

Este procedimiento es suficiente para generar todas las desigualdades válidas de un problema (PE), como afirma el siguiente teorema cuya demostración no detallaremos en este trabajo pero se puede consultar en el capítulo 8 de [24].

Teorema 2.1. Todas las desigualdades válidas para Y se pueden obtener utilizando el procedimiento Chavátal-Gomory un número finito de veces, [24].

Nótese que este procedimiento es recursivo y requiere el uso de las desigualdades obtenidas en los pasos anteriores para generar nuevas.

Por otro lado, por el Teorema 2.1 sabemos que, en teoría, podemos generar cualquier desigualdad válida usando este procedimiento, pero en la práctica esto no es cierto. El problema se debe a una convergencia lenta del algoritmo ya que las desigualdades obtenidas no son muy fuertes en general. Normalmente, nunca se obtiene una faceta de la envolvente convexa utilizando este procedimiento, esto se debe a que el redondeo realizado en el paso 3 del Algoritmo 1 “empuja” la desigualdad generada en el paso 2 pero, en

general, no llega tan lejos como para que la recta que define el hiperplano coincida con un punto de la región factible del problema entero. El límite de alcance con dicho empuje es que el hiperplano coincida con un punto entero, que puede ser o no de la región factible del problema, [14].

En el Ejemplo 2.2 podemos ver un caso en el que en el último paso del procedimiento la desigualdad dada en (2.7) “empuja” al hiperplano generado por la desigualdad del paso 2, dada por la ecuación (2.2), hasta que la recta que lo define coincida con un punto de la región factible, Y , [14]. Sin embargo, nótese que si, por ejemplo, tomásemos el vector $u_2 = (\frac{8}{11}, \frac{6}{11}, 0)$, entonces en el paso 2 obtendríamos la desigualdad $2y_1 + 2y_2 \leq \frac{152}{11}$. Posteriormente, en el paso 3, se tendría la desigualdad $2y_1 + 2y_2 \leq 13$, que viene representada con una línea discontinua verde en la Figura 2.2. Obsérvese que, en este caso, ningún punto factible de \mathbb{Z}^2 cae sobre la recta $2y_1 + 2y_2 = 13$, [14].

2.2.2. Cortes de Gomory fraccionales

Los cortes de Gomory fraccionales son una de las estrategias más utilizadas para generar desigualdades válidas en problemas cuyas variables de decisión toman valores enteros. La idea general es sencilla. Una vez se ha resuelto la relajación continua del problema original (PE) con el algoritmo del simplex [19] y se ha obtenido una solución óptima con componentes fraccionarias, se construye una desigualdad válida o corte a partir de la tabla óptima del simplex. Esta nueva desigualdad eliminará la solución óptima fraccionaria pero mantendrá todas las soluciones factibles con valores enteros del problema original.

En cada iteración del algoritmo, la región factible se va haciendo más pequeña y, por lo tanto, la aproximación a la envolvente convexa del problema original con variables enteras cada vez es mejor, de acuerdo a la Definición 1.5.

Este procedimiento puede repetirse hasta que todas las componentes del problema relajado sean enteras.

Para el desarrollo de esta sección seguiremos [19]. Formalmente, supongamos que tenemos un problema de optimización entera de la forma (PE) cuya región factible viene dada por

$$Y = \{y \in \mathbb{Z}_+^p : Ay \leq b\} \quad (2.8)$$

donde $A \in \mathcal{M}_{m \times p}(\mathbb{Z})$ y $b \in \mathbb{Z}^m$.

Nótese, que sin pérdida de generalidad, podemos asumir que los coeficientes de A y b son números enteros. En caso contrario, es decir, si estos coeficientes fueran números racionales, bastaría con multiplicar cada restricción por el mínimo común múltiplo de dichos valores para tener coeficientes enteros. Por ejemplo, la restricción

$$\frac{1}{3}y_1 + \frac{1}{6}y_2 \leq \frac{2}{5}$$

con valores fraccionarias, podemos escribirla como:

$$10y_1 + 5y_2 \leq 12$$

al multiplicar por el mínimo común múltiplo de 3, 6 y 5, que es 30.

Además, dado que vamos a utilizar el algoritmo del simplex, supondremos que la región factible del problema original está escrita en su forma estándar, es decir, que (2.8) se escribe como:

$$Y = \{y \in \mathbb{Z}_+^{p+m} : (A, I)y = b\} \quad (2.9)$$

donde $y = (y_1, \dots, y_p, y_{p+1}, \dots, y_{p+m})$ es el vector de variables de decisión, en el que las p primeras variables y_1, \dots, y_p son las variables de decisión del problema original mientras que las m últimas componentes y_{p+1}, \dots, y_{p+m} son las llamadas variables de holgura, [19]. Además, $I \in \mathcal{M}_{m \times m}(\mathbb{Z})$ representa la matriz identidad de orden m .

Supongamos ahora que resolvemos la relajación continua de un problema (PE) con región factible (2.9) utilizando el método del simplex. Recordemos que en el método del simplex distinguimos entre variables de decisión básicas y no básicas.

Denotamos por $J_B \subset \{1, \dots, p+m\}$ al conjunto de índices de las variables básicas. Por otro lado, $J_{NB} = \{1, \dots, p+m\} \setminus J_B$ será el conjunto de índices de las variables no básicas. Supondremos, además que la solución óptima del problema relajado viene dada por y^* , y que la tabla óptima asociada se puede resumir en:

$$b^* | I \ A^*, \quad (2.10)$$

siendo $b^* \in \mathbb{R}^m$ el vector de términos independientes obtenido tras las transformaciones del simplex, $I \in \mathcal{M}_{m \times m}(\mathbb{Z})$ la matriz identidad asociada a las variables básicas dadas por los índices en J_B , y A^* la matriz de restricciones resultante tras el algoritmo del simplex que está asociada a las variables no básicas cuyos índices están en J_{NB} .

Por otro lado, sabemos que y^* se puede escribir como $y^* = (y_B^*, y_{NB}^*)$ donde y_B^* e y_{NB}^* son, respectivamente, los vectores con las componentes básicas y no básicas de J_B y J_{NB} . Además, en el óptimo se tiene que $y_B^* = b^*$ e $y_{NB}^* = 0$.

Si y^* tiene todas sus componentes enteras, o lo que es lo mismo, si las componentes de b^* tomaran valores enteros, entonces habríamos acabado, puesto que la solución óptima del problema relajado coincide con el óptimo del problema original (PE).

Por el contrario, supongamos que existe $r \in J_B$ tal que $y_r^* \notin \mathbb{Z}$. Nuestro objetivo será construir una desigualdad válida, llamada corte de Gomory, que elimine dicha solución fraccionaria sin eliminar ninguna solución entera factible.

Consideramos, por lo tanto, la única restricción de la tabla óptima del simplex en la que aparece y_r :

$$y_r + \sum_{j \in J_{NB}} a_{rj}^* y_j = b_r^*. \quad (2.11)$$

Nótese que $b_r^* \notin \mathbb{Z}$ pues $y_r^* = b_r^*$, ya que $y_j^* = 0$, para todo $j \in J_{NB}$ e $y_r^* \notin \mathbb{Z}$ por hipótesis.

A continuación, expresamos a_{rj}^* en función de su parte entera y su parte fraccionaria. Luego, tenemos que

$$a_{rj}^* = \lfloor a_{rj}^* \rfloor + f_{rj}$$

siendo, $0 \leq f_{rj} < 1$.

Por lo que podemos escribir (2.11) como:

$$y_r + \sum_{j \in J_{NB}} (\lfloor a_{rj}^* \rfloor + f_{rj}) y_j = b_r^*$$

o equivalentemente,

$$y_r + \sum_{j \in J_{NB}} \lfloor a_{rj}^* \rfloor y_j = b_r^* - \sum_{j \in J_{NB}} f_{rj} y_j. \quad (2.12)$$

Nótese que, fijado $r \in J_B$, se tiene que $f_{rj} \geq 0$, para todo $j \in J_{NB}$, e $y_j \geq 0$, para todo $j \in J_{NB}$ por hipótesis. Luego, se tiene que

$$\sum_{j \in J_{NB}} f_{rj} y_j \geq 0$$

y, en consecuencia,

$$b_r^* - \sum_{j \in J_{NB}} f_{rj} y_j \leq b_r^*. \quad (2.13)$$

Además, observamos que el lado izquierdo de la ecuación (2.12) debe ser un valor entero, si nos centramos en el problema original, pues $y_r \in \mathbb{Z}$, $y_j \in \mathbb{Z}$, para todo $j \in J_{NB}$ y $\lfloor a_{rj} \rfloor \in \mathbb{Z}$ por definición, para todo $j \in J_{NB}$. Luego, el lado derecho de (2.12), dado por $b_r^* - \sum_{j \in J_{NB}} f_{rj} y_j$ también debe ser un número entero. De esta forma, la desigualdad (2.13) podemos escribirla como:

$$b_r^* - \sum_{j \in J_{NB}} f_{rj} y_j \leq \lfloor b_r^* \rfloor$$

o equivalentemente,

$$\sum_{j \in J_{NB}} f_{rj} y_j \geq b_r^* - \lfloor b_r^* \rfloor$$

que, escrito en términos de los coeficientes de la tabla óptima del simplex quedaría

$$\sum_{j \in J_{NB}} (a_{rj}^* - \lfloor a_{rj}^* \rfloor) y_j \geq b_r^* - \lfloor b_r^* \rfloor. \quad (2.14)$$

La desigualdad (2.14) es conocida como Corte de Gomory fraccional.

Tal y como se ha comentado, el corte de Gomory fraccional dado en (2.14) viene expresado en función de los coeficientes de las restricciones de la tabla óptima del simplex. Sin embargo, es posible obtenerlos en función de los coeficientes originales. Para ello debemos definir la matriz B . En particular, denotaremos por B a la matriz de dimensiones $m \times m$ cuyas columnas son las columnas asociadas a las variables básicas en la solución óptima de la matriz de restricciones del problema original expresada en su forma estándar, es decir, de la matriz $(A|I)$. Sabiendo esto, tenemos que la columna a_j^* se puede expresar como:

$$a_j^* = B^{-1} a_j$$

siendo B^{-1} la inversa de la matriz B . De igual forma tenemos que el vector b^* quedará así:

$$b^* = B^{-1} b.$$

Con estas expresiones, podemos enunciar el siguiente resultado cuya demostración se puede obtener a partir de [14].

Teorema 2.2. [24]. Sea $r \in J_B$ un índice de las variables básicas obtenidas en la tabla óptima del simplex tras resolver el problema relajado de (PE). Sea B la matriz de coeficientes asociada a la solución óptima definida anteriormente. Sea β^r la fila r -ésima de la matriz B^{-1} , y sea $q_i = \beta_i^r - \lfloor \beta_i^r \rfloor$, $i = 1, \dots, m$. Entonces, el corte fraccional de Gomory (2.14) se puede escribir como un corte de Chvátal-Gomory

$$\sum_{j=1}^p \lfloor u' a_j \rfloor y_j \leq \lfloor u' b \rfloor, \quad (2.15)$$

tomando $u = q$, y resultando

$$\sum_{j=1}^p \lfloor q' a_j \rfloor y_j \leq \lfloor q' b \rfloor. \quad (2.16)$$

Vamos a ilustrar lo anterior con un ejemplo, [22].

Ejemplo 2.3. Consideramos el siguiente problema (PE):

$$\begin{aligned} \min_y \quad & 3y_1 + 4y_2 \\ \text{s.a.} \quad & 3y_1 + y_2 \geq 4 \\ & y_1 + 2y_2 \geq 4 \\ & y \in \mathbb{Z}_+^2 \end{aligned} \tag{2.17}$$

La relajación del problema (2.17) escrito con su formulación estándar quedaría:

$$\begin{aligned} \min_y \quad & 3y_1 + 4y_2 \\ \text{s.a.} \quad & -3y_1 - y_2 + y_3 = -4 \\ & -y_1 - 2y_2 + y_4 = -4 \\ & y \geq 0 \end{aligned} \tag{2.18}$$

donde (y_1, y_2) es el vector de las variables originales del problema e (y_3, y_4) es el vector de las llamadas variables de holgura.

A continuación, pasamos a resolver el problema (2.18) mediante el algoritmo del simplex. La Tabla 2.1 representa la tabla óptima asociada al problema (2.18). Por lo tanto, la solución óptima del problema (2.18) es $y^* = (\frac{4}{5}, \frac{8}{5}, 0, 0)$ con valor de $\frac{44}{5}$ en la función objetivo.

	3	4	0	0
$\frac{4}{5}$	1	0	$-\frac{2}{5}$	$\frac{1}{5}$
$\frac{8}{5}$	0	1	$\frac{1}{5}$	$-\frac{3}{5}$
$\frac{44}{5}$	0	0	$-\frac{2}{5}$	$-\frac{9}{5}$

Tabla 2.1: Tabla óptima del algoritmo del simplex asociada al problema (2.18).

La solución óptima y^* tiene dos componentes fraccionarias, por lo tanto no es solución del problema entero (2.17). Seleccionamos la segunda componente $y_2^* = \frac{8}{5}$ para construir el corte de Gomory. Nótese que también podríamos haber elegido la primera componente, quedando un corte alternativo.

Tomamos la única restricción de la tabla óptima del simplex en la que aparece y_2 :

$$y_2 + \frac{1}{5}y_3 - \frac{3}{5}y_4 = \frac{8}{5} \tag{2.19}$$

Ahora vamos a expresar los coeficientes de las variables de holgura en función redondeo por abajo de las mismas y su parte fraccionaria. Entonces, escribimos $\frac{1}{5} = 0 + \frac{1}{5}$ y $-\frac{3}{5} = -1 + \frac{2}{5}$. De esta forma, podemos escribir (2.19) como:

$$y_2 + \left(0 + \frac{1}{5}\right)y_3 + \left(-1 + \frac{2}{5}\right)y_4 = \frac{8}{5} \tag{2.20}$$

o lo que es lo mismo,

$$y_2 + 0y_3 - y_4 = \frac{8}{5} - \left(\frac{1}{5}y_3 + \frac{2}{5}y_4\right). \tag{2.21}$$

Por otro lado, está claro que, como $y_3, y_4 \geq 0$, se tiene

$$\frac{8}{5} - \left(\frac{1}{5}y_3 + \frac{2}{5}y_4\right) \leq \frac{8}{5}. \tag{2.22}$$

Siguiendo con la línea de razonamiento descrita anteriormente, sabemos que si nos centramos en el problema original (2.17), se tiene que $y_2, y_3, y_4 \in \mathbb{Z}$, luego la parte izquierda de (2.21) es entera y en consecuencia, la parte derecha $\frac{8}{5} - \left(\frac{1}{5}y_3 + \frac{2}{5}y_4\right)$ también lo es. Por lo tanto, haciendo uso de la Proposición 2.2 llegamos a la desigualdad:

$$\frac{8}{5} - \left(\frac{1}{5}y_3 + \frac{2}{5}y_4\right) \leq \left\lfloor \frac{8}{5} \right\rfloor = 1, \quad (2.23)$$

o equivalentemente,

$$\frac{1}{5}y_3 + \frac{2}{5}y_4 \geq \frac{8}{5} - 1 = \frac{3}{5}. \quad (2.24)$$

Es decir, hemos construido un corte de Gomory dado por la desigualdad (2.24).

A continuación, resolveremos el problema (2.18) introduciendo dicho corte para ver que, efectivamente, elimina soluciones fraccionarias. Para ello, partimos de la tabla óptima del simplex que aparece en la Tabla 2.1. Introducimos la nueva restricción (2.24) y resolvemos continuando con el algoritmo. Como se puede observar en la nueva tabla óptima obtenida que aparece en la Tabla 2.2, la solución óptima de dicho problema es $y^* = (2, 1, 3, 0, 0)$, con valor de 10 en la función objetivo. Dado que ya no existen componentes fraccionarias en la solución óptima, no es necesario continuar el proceso, puesto que hemos encontrado la solución óptima del problema original (2.17): $(y_1^*, y_2^*) = (2, 1)$.

	3	4	0	0	0
2	1	0	0	1	-2
1	0	1	0	-1	1
3	0	0	1	2	-5
10	0	0	0	-1	-2

Tabla 2.2: Tabla óptima del algoritmo del simplex asociada al problema (2.18) incluyendo el corte de Gomory (2.24).

Vamos a escribir (2.24) en función de las variables básicas y_1 e y_2 tal y como se explica en el Teorema 2.2. En este caso, la matriz B^{-1} es

$$\begin{pmatrix} -\frac{2}{5} & \frac{1}{5} \\ \frac{1}{5} & -\frac{3}{5} \end{pmatrix}$$

Por lo tanto, de acuerdo con el Teorema 2.2, el vector de pesos de Gomory q tendrá por componentes:

$$q_1 = \frac{1}{5} - 0 = \frac{1}{5}, \quad q_2 = -\frac{3}{5} - (-1) = \frac{2}{5}.$$

Es decir, $q' = \left(\frac{1}{5}, \frac{2}{5}\right)$. Luego, sabiendo que $a'_1 = (-3, -1)$, $a'_2 = (-1, -2)$ y $b' = (-4, -4)$ la restricción en las nuevas variables será:

$$\lfloor q'a_1 \rfloor y_1 + \lfloor q'a_2 \rfloor y_2 = -y_1 - y_2 \leq \lfloor q'b \rfloor = -3,$$

o equivalentemente,

$$y_1 + y_2 \geq 3. \quad (2.25)$$

En la Figura 2.3, se puede apreciar que (2.25), que viene representada con una línea discontinua roja, es un corte, ya que elimina puntos fraccionarios de la región factible del

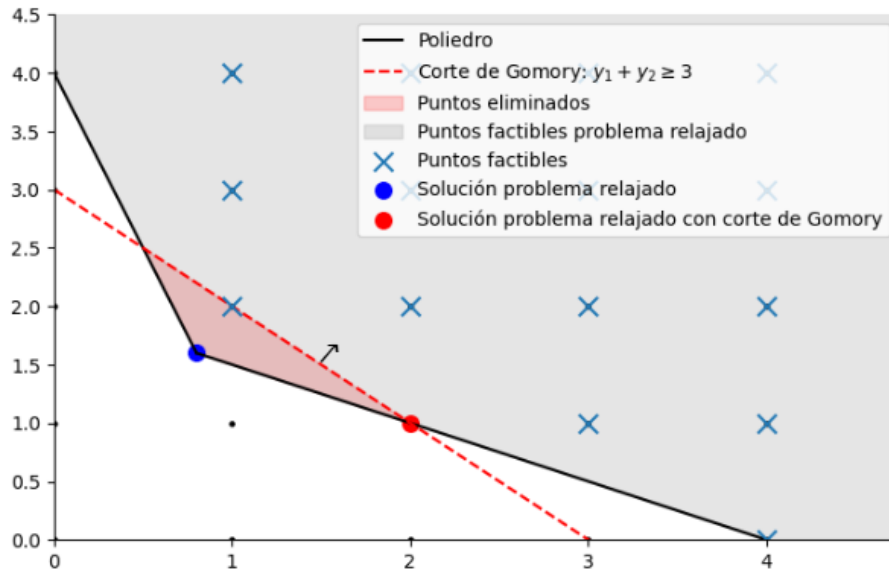


Figura 2.3: Ilustración del Ejemplo 2.3

problema relajado. Los puntos de la región factible del problema original vienen representados con cruces azules mientras que la región factible del problema relajado aparece en gris. Además, el punto azul representa la solución óptima del problema relajado, recordemos que es $(\frac{4}{5}, \frac{8}{5})$, y la solución óptima obtenida gracias a introducir el corte de Gomory en el problema, $(2,1)$, viene representada por un punto rojo.

2.2.3. Corte de integralidad en el objetivo

A lo largo del proceso de ramificación y acotación para problemas de la forma (PE), explicado en el Capítulo 1, se obtienen distintas soluciones. De hecho, sabemos que las cotas superiores del valor óptimo en la función objetivo se van actualizando. Estas cotas superiores se pueden utilizar para obtener cortes. Si $c \in \mathbb{Z}^p$ es el vector de coeficientes de la función objetivo, formado por valores enteros, $y \in \mathbb{Z}^p$ y $z^* \in \mathbb{R}$ es la mejor cota superior encontrada para dicho valor objetivo, podemos obtener el llamado corte de integralidad en el objetivo, [4]:

$$c'y \leq \lfloor z^* \rfloor \tag{2.26}$$

Ejemplo 2.4. Consideremos el problema (PE) definido en el Ejemplo 1.7. En este caso $c' = (5, 8)$ y, efectivamente, $c \in \mathbb{Z}^2$. La función objetivo del problema relajado tiene un valor de 41.25 en la solución óptima. De esta forma se obtiene el corte de integralidad en el objetivo descrito en (2.26) como:

$$5y_1 + 8y_2 \leq 41. \tag{2.27}$$

En la Figura 2.4, se puede observar que, efectivamente (2.27) es un corte, puesto que reduce la región factible del problema relajado original.

Todas las desigualdades construidas para problemas (PE) también son válidas para problemas (PEB). En el Capítulo 3, daremos ejemplos de desigualdades válidas para un tipo de problema (PEB) con cierta estructura, el problema de la mochila [24].

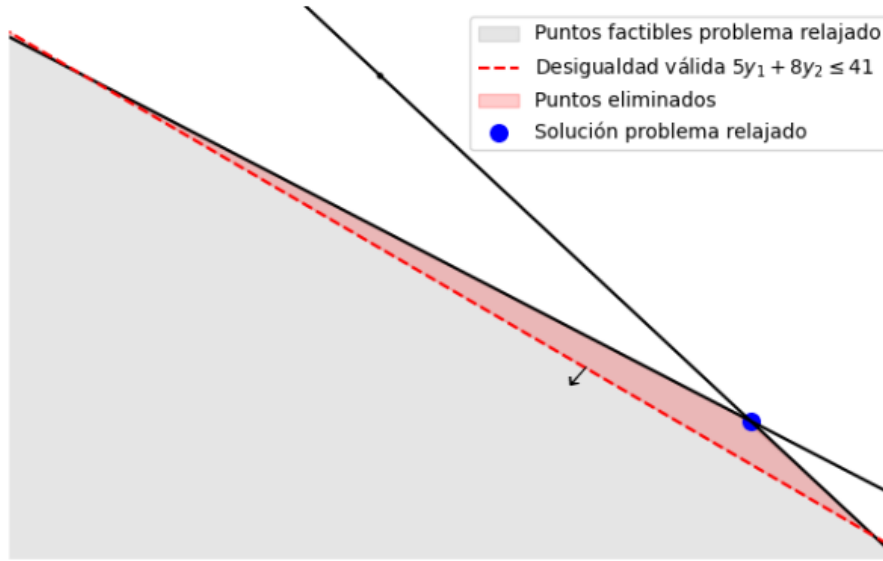


Figura 2.4: Ilustración del Ejemplo 2.4

2.3. Desigualdades válidas para problemas PEM

En la sección 2.2 se han analizado las desigualdades válidas para problemas con variables enteras. De hecho, hemos visto que cuando $y \leq b$, $y \in \mathbb{Z}$, la desigualdad $y \leq \lfloor b \rfloor$ es suficiente para generar todas las desigualdades para un problema (PE). En esta sección, estudiaremos distintas desigualdades válidas para problemas que tienen tanto variables de decisión continuas como enteras. En particular, comenzaremos examinando si existe una desigualdad básica similar a la estudiada en la Proposición 2.2 pero para problemas (PEM).

Proposición 2.3. Consideremos el conjunto $X = \{(x, y) \in \mathbb{R}_+ \times \mathbb{Z} : y \leq b + x\}$. Sea $f = b - \lfloor b \rfloor > 0$. Entonces la desigualdad

$$y \leq \lfloor b \rfloor + \frac{x}{1 - f} \tag{2.28}$$

es válida para X , [5].

Demostración. Vamos a probar que la desigualdad (2.28) se satisface para todo punto $(x, y) \in X$. Nótese que, como $y \in \mathbb{Z}$, puede ocurrir que o bien $y \leq \lfloor b \rfloor$ o bien $y \geq \lfloor b \rfloor + 1$.

Supongamos que estamos en el caso $y \leq \lfloor b \rfloor$. Sabemos además, que $x \geq 0$, o lo que es lo mismo, $-x \leq 0$. Luego, dado que $f = b - \lfloor b \rfloor < 1$, se tiene que $1 - f > 0$, al igual que $\frac{1}{1-f} > 0$. Así pues $-\frac{x}{1-f} \leq 0$. Por lo tanto, sumando la desigualdad $y \leq \lfloor b \rfloor$ y $-\frac{x}{1-f}$ tenemos que:

$$y - \frac{x}{1 - f} \leq \lfloor b \rfloor, \tag{2.29}$$

o equivalentemente,

$$y \leq \lfloor b \rfloor + \frac{x}{1 - f}, \tag{2.30}$$

que coincide con la expresión (2.28).

Si por el contrario, suponemos que estamos en el caso $y \geq [b] + 1$. Como se ha demostrado anteriormente, $\frac{1}{1-f} > 0$, por lo tanto, $\frac{f}{1-f} > 0$, puesto que $f > 0$ por hipótesis. Así pues, obtenemos la desigualdad:

$$\frac{yf}{1-f} \geq \frac{([b] + 1)f}{1-f}. \quad (2.31)$$

Por otro lado, multiplicando la desigualdad que tenemos como hipótesis, $y - x \leq b$, por $\frac{1}{1-f} > 0$ obtenemos:

$$\frac{y-x}{1-f} \leq \frac{b}{1-f}. \quad (2.32)$$

Sumando (2.31) y (2.32) llegamos a la desigualdad válida:

$$\frac{f[b] + f + y - x}{1-f} \leq \frac{yf + b}{1-f}. \quad (2.33)$$

Sustituyendo el segundo sumando del lado izquierdo de (2.33) por $f = b - [b]$, tenemos:

$$\frac{f[b] + (b - [b]) + y - x}{1-f} \leq \frac{yf + b}{1-f}. \quad (2.34)$$

A continuación, extraemos factor común $[b]$ en la parte de la izquierda de la desigualdad y obtenemos:

$$\frac{(f-1)[b] + b + y - x}{1-f} \leq \frac{yf + b}{1-f}. \quad (2.35)$$

En el siguiente paso, reordenamos términos y extraemos y , llegando a:

$$\frac{(f-1)[b] + y(1-f)}{1-f} \leq \frac{x}{1-f}, \quad (2.36)$$

o lo que es equivalente,

$$-[b] + y \leq \frac{x}{1-f}, \quad (2.37)$$

que escrito de otra forma, se obtiene:

$$y \leq [b] + \frac{x}{1-f}, \quad (2.38)$$

llegando así a la desigualdad (2.28) que buscábamos. \square

Nótese que cuando la variable continua x se anula obtenemos la desigualdad de la que hablabamos al principio de la sección $y \leq [b]$, que viene recogida en la Proposición 2.2. Además, conviene resaltar que en la demostración no importa si $y \in \mathbb{Z}$ o si $y \in \mathbb{Z}_+$, mientras que $x \in \mathbb{R}_+$ es una hipótesis imprescindible.

2.3.1. Corte por Redondeo Entero Mixto (Mixed Integer Rounding)

En esta sección, vamos a construir una desigualdad válida a la que llamaremos desigualdad por redondeo entero mixto o, como se conoce en inglés, *Mixed Integer Rounding (MIR)*. Esta será una generalización en $\mathbb{R}_+^n \times \mathbb{Z}_+^p$ de la desigualdad que aparece en la Proposición 2.3.

Para empezar, vamos a obtener una pequeña variante de la desigualdad (2.28). Consideramos el conjunto:

$$X^* = \{(x, y) \in \mathbb{R}_+ \times \mathbb{Z}_+^2 : g_1 y_1 + g_2 y_2 \leq b + x\},$$

donde $g_1, g_2 \in \mathbb{R}$ y $b \notin \mathbb{Z}$.

Proposición 2.4. Sea $f = b - \lfloor b \rfloor > 0$ y $f_i = g_i - \lfloor g_i \rfloor \geq 0$ para $i = 1, 2$. Entonces:

$$\lfloor g_1 \rfloor y_1 + \left(\lfloor g_2 \rfloor + \frac{f_2 - f}{1 - f} \right) y_2 \leq \lfloor b \rfloor + \frac{x}{1 - f} \quad (2.39)$$

es válida para X^* , [24].

Demostración. Si $(x, y) \in X^*$, entonces satisface

$$\lfloor g_1 \rfloor y_1 + \lceil g_2 \rceil y_2 \leq b + x + (1 - f_2) y_2,$$

ya que $g_2 = \lceil g_2 \rceil - (1 - f_2)$ e $y_1 \geq 0$ y, por lo tanto, $\lfloor g_1 \rfloor y_1 \leq g_1 y_1$. Ahora, como $\lfloor g_1 \rfloor y_1 + \lceil g_2 \rceil y_2 \in \mathbb{Z}$ y $x + (1 - f_2) y_2 \in \mathbb{R}_+$ puesto que $y_2 \geq 0$, podemos aplicar la Proposición 2.3 obteniendo así la siguiente desigualdad:

$$\lfloor g_1 \rfloor y_1 + \lceil g_2 \rceil y_2 \leq \lfloor b \rfloor + \frac{x + (1 - f_2) y_2}{1 - f}.$$

A continuación, utilizando que $1 = \lceil g_2 \rceil - \lfloor g_2 \rfloor$, llegamos a:

$$\lfloor g_1 \rfloor y_1 + (1 + \lfloor g_2 \rfloor) y_2 \leq \lfloor b \rfloor + \frac{x + (1 - f_2) y_2}{1 - f}.$$

Luego, escribiendo $1 = \frac{1-f}{1-f}$ y sumando $-\frac{(1-f_2)}{1-f} y_2$ a ambos lados de la desigualdad obtenemos:

$$\lfloor g_1 \rfloor y_1 + \left(\frac{1-f}{1-f} + \lfloor g_2 \rfloor - \frac{1-f_2}{1-f} \right) y_2 \leq \lfloor b \rfloor + \frac{x}{1-f},$$

o equivalentemente,

$$\lfloor g_1 \rfloor y_1 + \left(\lfloor g_2 \rfloor + \frac{f_2 - f}{1 - f} \right) y_2 \leq \lfloor b \rfloor + \frac{x}{1 - f}. \quad (2.40)$$

La desigualdad (2.40) es la expresión a la que queríamos llegar. □

En el siguiente teorema se expone la generalización en $\mathbb{R}_+^n \times \mathbb{Z}_+^p$ de la Proposición 2.3 y la Proposición 2.4. La demostración del teorema se puede consultar en [6].

Teorema 2.3. Consideramos un conjunto entero mixto definido por una única desigualdad $X = \{(x, y) \in \mathbb{R}_+^n \times \mathbb{Z}_+^p : ax + gy \leq b\}$. Sea $f = b - \lfloor b \rfloor > 0$ y $f_j = g_j - \lfloor g_j \rfloor \geq 0$. Entonces la desigualdad

$$\sum_{j=1}^p \left(\lfloor g_j \rfloor + \frac{(f_j - f)^+}{1 - f} \right) y_j + \frac{1}{1 - f} \sum_{j: a_j < 0} a_j x_j \leq \lfloor b \rfloor, \quad (2.41)$$

donde $(f_j - f)^+ = \max(0, f_j - f)$, es una desigualdad válida para $\text{conv}(X)$, [6].

Llamaremos desigualdad Mixed Integer Rounding (MIR) a (2.41), [24].

2.3.2. Cortes de Gomory mixtos

Los cortes de Gomory mixtos son la generalización en $\mathbb{R}_+^n \times \mathbb{Z}_+^p$ de los cortes de Gomory fraccionales vistos anteriormente en la Sección 2.2.2. De esta forma, al igual que en los cortes de Gomory fraccionales, la idea es partir de la solución óptima de la relajación continua del problema original (PEM) con el algoritmo del simplex [19]. Si esta solución tiene componentes fraccionarias en las variables que toman valores enteros en el problema original, se construye una desigualdad válida a partir de la tabla óptima del simplex siguiendo un procedimiento similar al que se ha descrito en la Sección 2.2.2 con el que conseguiremos la integrabilidad. Se repetirá el procedimiento hasta conseguir que todas las variables del problema relajado que en el problema original tenían la restricción de integrabilidad tomen valores enteros en el problema relajado.

Para construir el corte de Gomory mixto, al igual que se hizo con los cortes de Gomory fraccionales, seguiremos [19]. En este caso, consideramos un problema de optimización entero mixto de la forma (PEM). Recordamos que la región factible de dicho problema viene dada por

$$X = \{z = (x, y) \in \mathbb{R}_+^n \times \mathbb{Z}_+^p : Ax + Gy \leq b\} \quad (2.42)$$

donde $A \in \mathcal{M}_{m \times n}(\mathbb{R})$ y $G \in \mathcal{M}_{m \times p}(\mathbb{R})$. Nótese que z es un vector con $n + p$ componentes, de forma que $(z_1, \dots, z_n) = (x_1, \dots, x_n) \in \mathbb{R}_+^n$, mientras que $(z_{n+1}, \dots, z_{n+p}) = (y_1, \dots, y_p) \in \mathbb{Z}_+^p$.

Al igual que se hizo en la Sección 2.2.2, llamaremos $J_B \subset \{1, \dots, n + p + m\}$ al conjunto de índices de las variables básicas y, análogamente, llamaremos $J_{NB} = \{1, \dots, n + p + m\} \setminus J_B$ al conjunto de índices de las variables no básicas. Consideramos además la relajación continua del problema (PEM) en forma estándar y su solución óptima asociada que llamaremos z^* . La región factible del problema original (2.42) en forma estándar se escribirá como:

$$X = \{z \in \mathbb{R}_+^n \times \mathbb{Z}_+^p \times \mathbb{R}_+^m : (A, G, I)z = b\}, \quad (2.43)$$

donde el vector (z_1, \dots, z_{n+p}) es el vector de variables de decisión del problema original y $(z_{n+p+1}, \dots, z_{n+p+m})$ son las llamadas variables de holgura del problema. Además, $I \in \mathcal{M}_{m \times m}(\mathbb{Z})$ es la matriz identidad de orden m .

Por otro lado, consideramos la última tabla óptima del simplex asociada a la solución anteriormente mencionada z^* , que se puede resumir en:

$$b^* | I \ H^*, \quad (2.44)$$

siendo $b^* \in \mathbb{R}^m$ el vector de términos independientes que se obtiene tras las transformaciones del algoritmo, $I \in \mathcal{M}_{m \times m}(\mathbb{Z})$ la matriz identidad asociada a las variables básicas dadas por los índices en J_B , y $H^* = (A, G)^*$ es la matriz de restricciones obtenida tras las transformaciones del simplex, asociada a las variables no básicas cuyos índices están en J_{NB} .

Sea z_r con $r \in \{n + 1, \dots, n + p\}$ una variable de decisión que toma valores enteros en un problema (PEM). Supongamos que z_r^* no es entera en dicha solución óptima. Así, z_r^* no es nula y por ello, es una variable básica. Consideremos ahora la única restricción en la tabla óptima del simplex en la que aparece z_r :

$$z_r = b_r^* - \sum_{j \in J_{NB}} h_{rj}^* z_j. \quad (2.45)$$

Claramente, b_r^* no es entera puesto que en el óptimo $z_j^* = 0$, para todo $j \in J_{NB}$ y, por lo tanto, $z_r^* = b_r^*$. Por ello, podemos escribir $b_r^* = \lfloor b_r^* \rfloor + f_r$, con $0 < f_r < 1$. De esta forma,

$$z_r - \lfloor b_r^* \rfloor = f_r - \sum_{j \in J_{NB}} h_j^* z_j. \quad (2.46)$$

Como z_r es una variable entera, satisface o bien $z_r \leq \lfloor b_r^* \rfloor$, o bien $z_r \geq \lfloor b_r^* \rfloor + 1$. Dicho de otra forma, se tiene que $z_r - \lfloor b_r^* \rfloor \leq 0$ o $z_r - \lfloor b_r^* \rfloor \geq 1$. Utilizando (2.46), esta condición es equivalente a

$$\text{o bien } \sum_{j \in J_{NB}} h_{rj}^* z_j \geq f_r, \text{ o bien } \sum_{j \in J_{NB}} h_{rj}^* z_j \leq f_r - 1. \quad (2.47)$$

Nótese que la restricción (2.47) es una restricción disyuntiva, de las que explicamos en el Capítulo 1.

A continuación, dividiremos ambos lados de la primera desigualdad en (2.47) entre $f_r > 0$, y ambos lados de la segunda restricción entre $1 - f_r > 0$. Esto nos lleva a una restricción disyuntiva equivalente a (2.47):

$$\text{o bien } \sum_{j \in J_{NB}} \frac{h_{rj}^*}{f_r} z_j \geq 1, \text{ o bien } - \sum_{j \in J_{NB}} \frac{h_{rj}^*}{1 - f_r} z_j \geq 1. \quad (2.48)$$

Por otro lado, fijado $j \in J_{NB}$, se tiene que $\max \left\{ \frac{h_{rj}^*}{f_r}, -\frac{h_{rj}^*}{1-f_r} \right\} \geq \frac{h_{rj}^*}{f_r}$ y $\max \left\{ \frac{h_{rj}^*}{f_r}, -\frac{h_{rj}^*}{1-f_r} \right\} \geq -\frac{h_{rj}^*}{1-f_r}$. Entonces si tomamos la suma a ambos lados llegamos a que

$$\sum_{j \in J_{NB}} \max \left\{ \frac{h_{rj}^*}{f_r}, -\frac{h_{rj}^*}{1-f_r} \right\} \geq \sum_{j \in J_{NB}} \frac{h_{rj}^*}{f_r} \quad (2.49)$$

y,

$$\sum_{j \in J_{NB}} \max \left\{ \frac{h_{rj}^*}{f_r}, -\frac{h_{rj}^*}{1-f_r} \right\} \geq \sum_{j \in J_{NB}} -\frac{h_{rj}^*}{1-f_r}. \quad (2.50)$$

Por lo tanto, como la expresión de la izquierda en (2.49) y (2.50) es mayor o igual que las expresiones de la derecha en ambas desigualdades, tendremos que el máximo entre $\sum_{j \in J_{NB}} \frac{h_{rj}^*}{f_r}$ y $\sum_{j \in J_{NB}} -\frac{h_{rj}^*}{1-f_r}$ será menor o igual que dicha expresión obteniendo así la siguiente desigualdad:

$$\sum_{j \in J_{NB}} \max \left\{ \frac{h_{rj}^*}{f_r}, -\frac{h_{rj}^*}{1-f_r} \right\} \geq \max \left\{ \sum_{j \in J_{NB}} \left(\frac{h_{rj}^*}{f_r} \right), \sum_{j \in J_{NB}} \left(-\frac{h_{rj}^*}{1-f_r} \right) \right\} \quad (2.51)$$

Como las variables no básicas toman valores no negativos, (2.51) implica que:

$$\sum_{j \in J_{NB}} \max \left\{ \frac{h_{rj}^*}{f_r}, -\frac{h_{rj}^*}{1-f_r} \right\} z_j \geq \max \left\{ \sum_{j \in J_{NB}} \left(\frac{h_{rj}^*}{f_r} \right) z_j, \sum_{j \in J_{NB}} \left(-\frac{h_{rj}^*}{1-f_r} \right) z_j \right\} \geq 1,$$

donde la última desigualdad se deriva del hecho de que al menos uno de los dos sumatorios $\sum_{j \in J_{NB}} \left(\frac{h_{rj}^*}{f_r} \right) z_j$ y $\sum_{j \in J_{NB}} \left(-\frac{h_{rj}^*}{1-f_r} \right) z_j$ es mayor o igual que 1. Por lo tanto, (2.48) implica la restricción:

$$\sum_{j \in J_{NB}} \max \left\{ \frac{h_{rj}^*}{f_r}, -\frac{h_{rj}^*}{1-f_r} \right\} z_j \geq 1. \quad (2.52)$$

Determinar el máximo entre $\frac{h_{rj}^*}{f_r}$ y $-\frac{h_{rj}^*}{1-f_r}$ es sencillo, porque para cada $j \in J_{NB}$, los valores de estas expresiones tienen signos opuestos, que dependen solo del signo de h_{rj}^* puesto que $0 < f_r < 1$. Para ser precisos, definimos:

$$J^+ = \{j \in J_{NB} : h_{rj}^* \geq 0\} \quad \text{y} \quad J^- = \{j \in J_{NB} : h_{rj}^* < 0\}.$$

Nótese que $J_{NB} = J^+ \cup J^-$. Entonces tenemos:

$$\text{máx} \left\{ \frac{h_{rj}^*}{f_r}, -\frac{h_{rj}^*}{1-f_r} \right\} = \begin{cases} \frac{h_{rj}^*}{f_r}, & \text{si } j \in J^+ \\ -\frac{h_{rj}^*}{1-f_r}, & \text{si } j \in J^- \end{cases}$$

Así, podemos escribir (2.52) como:

$$\sum_{j \in J^+} \frac{h_{rj}^*}{f_r} z_j - \sum_{j \in J^-} \frac{h_{rj}^*}{1-f_r} z_j \geq 1. \quad (2.53)$$

Llamaremos a la desigualdad (2.53) corte de Gomory mixto y representa la condición necesaria para que z_r tome un valor entero.

Por último, queremos expresar (2.53) en función de las variables originales del problema, para ello tenemos que expresar cada variable z_j con $j \in J_{NB}$ en función de las variables originales utilizando la primera tabla del algoritmo del simplex. Entonces, cada variable no básica se podrá escribir de la siguiente forma, para cada $i = 1, \dots, m$:

$$z_{n+p+i} = b_1 - \sum_{j=1}^n a_{ij} z_j - \sum_{j=1}^p g_{ij} z_j. \quad (2.54)$$

A continuación, mostraremos un ejemplo ilustrativo de como aplicar el corte de Gomory mixto.

Ejemplo 2.5. Partiremos del Ejemplo 2.3, y supondremos que y_1 es la única variable toma valores enteros en el problema original. Es decir, que en el problema original la restricción de integrabilidad de las variables solo se aplica a y_1 y no a y_2 convirtiendo así el problema (2.17) de la forma (PE) en un problema de la forma (PEM). Además, para seguir la notación que hemos adoptado en esta sección, llamaremos $z = (z_1, \dots, z_4) = (y_1, \dots, y_4)$. Recordamos que la tabla óptima del simplex del problema relajado en forma estándar es:

	3	4	0	0
$\frac{4}{5}$	1	0	$-\frac{2}{5}$	$\frac{1}{5}$
$\frac{8}{5}$	0	1	$\frac{1}{5}$	$-\frac{3}{5}$
$\frac{44}{5}$	0	0	$-\frac{2}{5}$	$-\frac{9}{5}$

Tabla 2.3: Tabla óptima del algoritmo del simplex asociada al problema (2.18).

Así, la restricción correspondiente a la variable básica z_1 es:

$$z_1 - \frac{2}{5} z_3 + \frac{1}{5} z_4 = \frac{4}{5}. \quad (2.55)$$

Por lo tanto, $J^+ = \{4\}$ y $J^- = \{3\}$. Por otro lado, $b_1^* = \frac{4}{5}$ y, por lo tanto, $f_1 = b_1^* - \lfloor b_1^* \rfloor = b_1^* = \frac{4}{5}$. Siguiendo el razonamiento descrito anteriormente, el corte de Gomory mixto que se obtiene es:

$$\frac{h_{14}^*}{f_1} z_4 - \frac{h_{13}^*}{1 - f_1} z_3 = \frac{1}{4} z_4 + 2z_3 \geq 1. \tag{2.56}$$

Recordamos que el corte de Gomory (2.56) es la condición necesaria para que la variable z_1 tome un valor entero. Introducimos la nueva restricción (2.56) en la Tabla 2.3 y continuamos con el algoritmo del simplex. De este modo, obtenemos una nueva tabla óptima que viene representada en la Tabla 2.4.

	3	4	0	0	0
1	1	0	0	$\frac{1}{4}$	$-\frac{1}{5}$
$\frac{3}{2}$	0	1	0	$-\frac{5}{8}$	$\frac{1}{10}$
$\frac{1}{2}$	0	0	1	$\frac{1}{8}$	$-\frac{1}{2}$
9	0	0	0	$-\frac{7}{4}$	$-\frac{1}{5}$

Tabla 2.4: Tabla óptima del algoritmo del simplex una vez añadida la restricción (2.56).

Por lo tanto, la solución óptima asociada al problema relajado una vez introducida la restricción (2.56) es $z^* = (1, \frac{3}{2}, \frac{1}{2}, 0, 0)$ con valor 9 en la función objetivo. Como se puede observar, la variable z_1 toma un valor entero en la solución óptima del problema relajado y, por lo tanto, hemos encontrado la solución óptima del problema original de la forma (PEM): $(z_1^*, z_2^*) = (1, \frac{3}{2})$.

Por último, vamos a escribir el corte de Gomory (2.56) en términos de las variables de decisión originales z_1 y z_2 . Como sabemos que $z_3 = -4 + 3z_1 + z_2$ y $z_4 = -4 + z_1 + 2z_2$, sustituyendo ambas expresiones en (2.56), llegamos a la desigualdad válida:

$$\frac{25}{4} z_1 + \frac{5}{2} z_2 \geq 10. \tag{2.57}$$

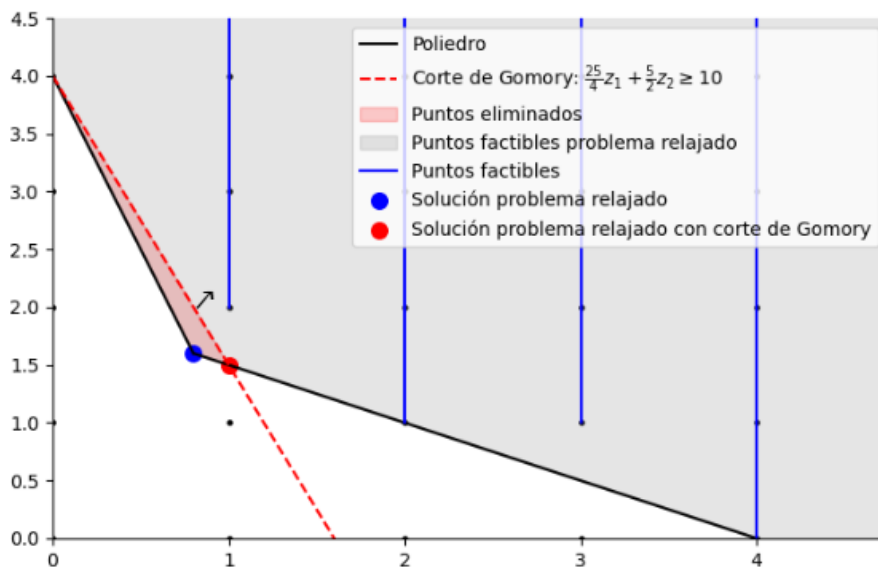


Figura 2.5: Ilustración del Ejemplo 2.5

En la Figura 2.5, se observa que la restricción (2.57), representada con una línea discontinua roja, es un corte. Como se puede apreciar, esta desigualdad elimina puntos

fraccionarios que no pertenecen a la región factible del problema original, representada con líneas azules, al tratarse de un problema de la forma (PEM). La región gris de la imagen es la región factible del problema relajado, mientras que la región roja indica los puntos eliminados tras incluir el corte de Gomory. Además, el punto $(\frac{4}{5}, \frac{8}{5})$, que aparece en azul, representa la solución óptima del problema relajado y la solución óptima obtenida gracias a introducir el corte de Gomory en el problema, $(1, \frac{3}{2})$, viene representada por un punto rojo.

Capítulo 3

Desigualdades válidas en problemas con cierta estructura. El problema de la mochila

En el Capítulo 2 hemos presentado una teoría general sobre desigualdades válidas para problemas de tipo (PE) y (PEM) y algunas técnicas para construir dichas desigualdades. Sin embargo, estas técnicas generales pueden ser ineficientes a la hora de desarrollar facetas de la envolvente convexa de la región factible de un problema de optimización con variables enteras, [14]. En este capítulo, estudiaremos desigualdades para un problema particular de tipo (PEB), el problema de la mochila [24]. Como sabemos, las desigualdades estudiadas en la Sección 2.2 son válidas para este problema particular. No obstante, utilizando la estructura del problema podemos desarrollar desigualdades más fuertes y eficientes. A continuación, detallaremos algunas de ellas.

3.1. Desigualdades de cubrimiento

Comenzamos esta sección, recordando el problema de la mochila que aparece en el Ejemplo 1.3 del Capítulo 1. Asumiremos que tenemos una mochila con una capacidad limitada que denotaremos como $b > 0$ y disponemos de p objetos que podemos introducir en la mochila. Para cada objeto $j = 1, \dots, p$, llamaremos $a_j > 0$ al espacio que ocupa en la mochila y $c_j > 0$ representará el beneficio que nos proporciona. Las variables de decisión, y_j , solo tomarán dos valores; 0, para denotar que no introduciremos el objeto j en la mochila, y 1, para el caso contrario. El objetivo del problema es elegir un conjunto de objetos que no excedan la capacidad máxima, b , y que a su vez, maximicen el beneficio total.

De esta forma, la formulación del problema se puede expresar como:

$$\begin{aligned} \text{máx}_y \quad & \sum_{j=1}^p c_j y_j \\ \text{s.a.} \quad & \sum_{j=1}^p a_j y_j \leq b \\ & y_j \in \{0, 1\}, \quad j = 1, \dots, p. \end{aligned} \tag{3.1}$$

Por lo tanto, la región factible del problema de la mochila es:

$$Y := \left\{ y \in \{0, 1\}^p : \sum_{j=1}^p a_j y_j \leq b \right\}, \quad (3.2)$$

donde $b > 0$ y $a_j > 0$ para $j \in P = \{1, \dots, p\}$.

A lo largo de este capítulo representaremos elementos del conjunto $\{0, 1\}^p$ utilizando vectores característicos de forma que si $R \subseteq P$, construimos el vector $y^R \in \{0, 1\}^p$ definiendo sus componentes como $y_j^R = 1$, si $j \in R$ e $y_j^R = 0$, en caso contrario. A continuación, definiremos los conceptos de cubrimiento y cubrimiento minimal para el conjunto (3.2).

Definición 3.1. Un conjunto $C \subseteq P$ es un cubrimiento para Y si $\sum_{j \in C} a_j > b$. Diremos que un cubrimiento es minimal si $\sum_{j \in C \setminus \{k\}} a_j \leq b$ para todo $k \in C$, [24].

Nótese que, si $C \subseteq P$ y C es un cubrimiento, entonces se verifica que $y^C \notin Y$, y viceversa.

Ejemplo 3.1. Consideramos el siguiente conjunto de la forma (3.2):

$$Y = \{y \in \{0, 1\}^7 : 11y_1 + 6y_2 + 6y_3 + 5y_4 + 5y_5 + 4y_6 + y_7 \leq 19\}. \quad (3.3)$$

En este caso, $P = \{1, 2, 3, 4, 5, 6, 7\}$. El conjunto $C_1 = \{3, 4, 5, 6\}$ es un cubrimiento para Y , puesto que $a_3 + a_4 + a_5 + a_6 = 6 + 5 + 5 + 4 = 20 > 19 = b$ y el conjunto $C_2 = \{1, 2, 3, 4\}$ también lo es ya que $a_1 + a_2 + a_3 + a_4 = 11 + 6 + 6 + 5 = 28 > 19 = b$. Además, C_1 es un cubrimiento minimal ya que $a_4 + a_5 + a_6 = 5 + 5 + 4 = 14 < 19 = b$, $a_3 + a_5 + a_6 = 6 + 5 + 4 = 15 < 19 = b$, $a_3 + a_4 + a_6 = 6 + 5 + 4 = 15 < 19 = b$ y $a_3 + a_4 + a_5 = 6 + 5 + 5 = 16 < 19 = b$. Sin embargo, C_2 no es un cubrimiento minimal al no cumplirse que $\sum_{j \in C_2 \setminus \{k\}} a_j \leq b$ para todo $k \in C_2$ puesto que, en particular,

$$\sum_{j \in C_2 \setminus \{4\}} a_j = 23 \not\leq 19 = b.$$

El siguiente resultado nos muestra que si disponemos de un cubrimiento C , entonces cualquier subconjunto que contenga a C también es un cubrimiento.

Proposición 3.1. Sea $C \subseteq P$ un cubrimiento para Y y $R \subseteq P$ un conjunto tal que $C \subseteq R$. Entonces, R también es un cubrimiento para Y , [24].

Demostración. Consideramos el vector y^R . Por hipótesis, tenemos que $C \subseteq R$ y, como $a_j > 0$ para todo $j \in P$, llegamos a $\sum_{j \in R} a_j \geq \sum_{j \in C} a_j$. Utilizando que C es un cubrimiento

para Y , es decir, que se verifica $\sum_{j \in C} a_j > b$, y lo anterior, llegamos a que $\sum_{j=1}^p a_j y_j^R = \sum_{j \in R} a_j \geq \sum_{j \in C} a_j > b$ y entonces $y^R \notin Y$, o lo que es lo mismo, R es un cubrimiento para Y . \square

Proposición 3.2. Si $C \subseteq P$ es un cubrimiento para Y , la desigualdad

$$\sum_{j \in C} y_j \leq |C| - 1, \quad (3.4)$$

llamada desigualdad de cubrimiento, es válida para Y , [24].

Demostración. Sea $R \subseteq P$ y consideremos el vector $y^R \in \{0, 1\}^p$. Probaremos que si y^R no satisface la desigualdad (3.4), entonces $y^R \notin Y$. Supongamos entonces que y^R no verifica la desigualdad de cubrimiento, es decir, que y^R cumple $\sum_{j \in C} y_j^R > |C| - 1$, o equivalentemente, $\sum_{j \in C} y_j^R \geq |C|$. Entonces, es fácil ver que $\sum_{j \in C} y_j^R \leq |C|$ por definición del vector y^R y, por lo tanto, tenemos que $|C \cap R| = \sum_{j \in C} y_j^R = |C|$ y así llegamos a que $C \subseteq R$. De esta forma, podemos aplicar la Proposición 3.1 resultando así que $y^R \notin Y$, como queríamos probar. \square

Es fácil ver que si un cubrimiento no es minimal, entonces la correspondiente desigualdad de cubrimiento es redundante puesto que es la suma de una desigualdad de cubrimiento minimal y algunas restricciones con cotas superiores, [4].

Ejemplo 3.2. Las desigualdades de cubrimiento asociadas a los cubrimientos, C_1 y C_2 , del Ejemplo 3.1 se escriben de la siguiente forma, respectivamente:

$$y_3 + y_4 + y_5 + y_6 \leq 3, \tag{3.5}$$

y

$$y_1 + y_2 + y_3 + y_4 \leq 3. \tag{3.6}$$

A continuación, vamos a ver un ejemplo de cubrimiento y cubrimiento minimal para dimensión 2, es decir, para $p = 2$.

Ejemplo 3.3. Consideremos la siguiente región factible del problema de la mochila:

$$Y = \{y \in \{0, 1\}^2 : 3y_1 + 2y_2 \leq 4\}.$$

Un cubrimiento para Y sería $C = \{1, 2\}$ ya que $a_1 + a_2 = 3 + 2 = 5 > 4 = b$ y, como además, $a_1 = 3 \leq 4 = b$ y $a_2 = 2 \leq 4 = b$, C es un cubrimiento minimal. La desigualdad de cubrimiento asociada es $y_1 + y_2 \leq |C| - 1 = 2 - 1 = 1$. Si observamos la Figura 3.1 podemos ver que la desigualdad, representada con una línea discontinua roja, es válida ya que elimina soluciones fraccionarias sin eliminar soluciones factibles del problema original. La región sombreada de color gris representa el conjunto de puntos factibles del problema relajado y, la región en color rojo, los puntos eliminados de dicha región gris. Además, es fácil ver que la desigualdad $y_1 + y_2 \leq 1$ corresponde a una faceta de $\text{conv}(Y)$.

3.2. Desigualdades de cubrimiento extendidas

A continuación, veremos otra desigualdad de cubrimiento para el conjunto Y , la desigualdad de cubrimiento extendida. Este tipo de desigualdad hace más fuertes a las desigualdades de cubrimiento anteriormente vistas, en el sentido de que eliminan más soluciones fraccionarias de la región factible del problema relajado, [24].

Proposición 3.3. Si $C \subseteq P$ es un cubrimiento minimal para Y y llamamos $E(C) = C \cup \{j \in P \setminus C : a_j \geq a_i \text{ para todo } i \in C\}$, entonces, la desigualdad

$$\sum_{j \in E(C)} y_j \leq |C| - 1, \tag{3.7}$$

llamada desigualdad de cubrimiento extendida, es válida para Y , [14].

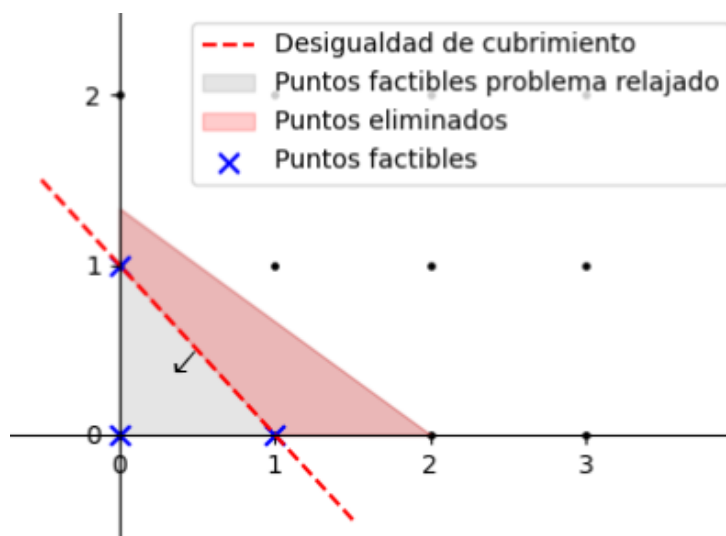


Figura 3.1: Ilustración del Ejemplo 3.3

La demostración de este resultado se puede consultar en [14]. En el siguiente ejemplo, se puede observar una desigualdad de cubrimiento extendida.

Ejemplo 3.4. En el Ejemplo 3.1 se ha dado un cubrimiento minimal para el conjunto (3.3), recordemos $C_1 = \{3, 4, 5, 6\}$. La extensión asociada a C_1 es $E(C_1) = C_1 \cup \{1, 2\} = \{1, 2, 3, 4, 5, 6\}$ puesto que se verifica que $a_1 = 11 \geq a_i$ para todo $i \in C_1$ y $a_2 = 6 \geq a_i$ para todo $i \in C_1$. Entonces, la desigualdad de cubrimiento extendida asociada se escribirá como:

$$y_1 + y_2 + y_3 + y_4 + y_5 + y_6 \leq 3. \quad (3.8)$$

Así, la desigualdad de cubrimiento (3.5) está dominada por la desigualdad válida (3.8), es decir, la restricción (3.8) es una restricción más fuerte que (3.5).

3.3. Procedimiento “Sequential Lifting”

En la sección anterior, hemos “fortalecido” las desigualdades de cubrimiento construyendo las llamadas desigualdades de cubrimiento extendidas. De esta forma, queremos encontrar un procedimiento que nos permita encontrar una desigualdad que sea lo más fuerte posible, es decir, que defina una faceta de $\text{conv}(Y)$. Veamos un ejemplo ilustrativo, [24]:

Ejemplo 3.5. Consideramos la desigualdad de cubrimiento asociada a C_1 vista en el Ejemplo 3.1, $y_3 + y_4 + y_5 + y_6 \leq 3$. Claramente, cuando $y_1 = y_2 = y_7 = 0$, la desigualdad anterior es válida para el conjunto $Y_1 = \{y \in \{0, 1\}^4 : 6y_3 + 5y_4 + 5y_5 + 4y_6 \leq 19\}$. Si por el contrario, hacemos que $y_2 = y_7 = 0$, queremos encontrar los valores de α_1 que hacen que la desigualdad

$$\alpha_1 y_1 + y_3 + y_4 + y_5 + y_6 \leq 3, \quad (3.9)$$

sea válida para el conjunto $Y_2 = \{y \in \{0, 1\}^5 : 11y_1 + 6y_3 + 5y_4 + 5y_5 + 4y_6 \leq 19\}$.

Cuando $y_1 = 0$, la desigualdad (3.9) es válida para cualquier valor de α_1 .

Cuando $y_1 = 1$, (3.9) será válida para Y_2 si y sólo si (3.9) se verifica para todo punto $y \in \{0, 1\}^4$ satisfaciendo $6y_3 + 5y_4 + 5y_5 + 4y_6 \leq 19 - 11$, o equivalentemente, si y sólo si $\alpha_1 + \epsilon \leq 3$, donde $\epsilon = \max\{y_3 + y_4 + y_5 + y_6 : 6y_3 + 5y_4 + 5y_5 + 4y_6 \leq 8, y \in \{0, 1\}^4\}$.

Resolviendo el problema, obtenemos que el valor máximo es $\epsilon = 1$ y se alcanza en el punto $y = (0, 0, 0, 1)$. Por lo tanto, $\alpha_1 \leq 2$. Resumiendo, la desigualdad (3.9) es válida para Y_2 para todo $\alpha_1 \leq 2$, y para $\alpha_1 = 2$ tenemos la desigualdad más fuerte.

En general, el problema es encontrar el mejor valor de α_j con $j \in P \setminus C$, donde C es un cubrimiento minimal para Y , tal que la desigualdad

$$\sum_{j \in C} y_j + \sum_{j \in P \setminus C} \alpha_j y_j \leq |C| - 1 \quad (3.10)$$

sea válida para Y , [24]. La desigualdad (3.10) es un “fortalecimiento” de la desigualdad de cubrimiento minimal $\sum_{j \in C} y_j \leq |C| - 1$. A este proceso, con el que fortalecemos las desigualdades de cubrimiento minimales, lo llamaremos levantamiento secuencial o, como se conoce en inglés, *sequential lifting*, [4]. A la desigualdad obtenida de la forma (3.10) la llamaremos desigualdad de cubrimiento levantada, que en inglés se conoce como *lifted cover inequality*.

A continuación, describiremos los pasos para construir una desigualdad de cubrimiento levantada, [24].

Consideramos el conjunto $\{j_1, \dots, j_r\}$ que será una ordenación de los elementos de $P \setminus C$. Tal y como su propio nombre indica, se trata de un algoritmo secuencial con un total de r iteraciones.

Consideramos una iteración cualquiera $t \in \{1, \dots, r\}$, y partimos de la última desigualdad obtenida, es decir, la desigualdad más fuerte obtenida hasta el momento que tendrá la siguiente forma:

$$\sum_{i=1}^{t-1} \alpha_{j_i} y_{j_i} + \sum_{j \in C} y_j \leq |C| - 1. \quad (3.11)$$

Para calcular el valor más grande de α_{j_t} que hace que la desigualdad

$$\alpha_{j_t} y_{j_t} + \sum_{i=1}^{t-1} \alpha_{j_i} y_{j_i} + \sum_{j \in C} y_j \leq |C| - 1 \quad (3.12)$$

sea válida para Y , debemos seguir un razonamiento similar al seguido en el Ejemplo 3.5. La idea principal es encontrar el vector y que cuando $y_{j_t} = 1$ satisfaga la restricción de capacidad

$$\sum_{i=1}^{t-1} a_{j_i} y_{j_i} + \sum_{j \in C} a_j y_j \leq b - a_{j_t} \quad (3.13)$$

y maximice el lado izquierdo de (3.11). Nótese que cuando $y_{j_t} = 0$, la restricción (3.12) siempre se cumple, pues asumimos conocido (3.11).

En consecuencia, esto nos lleva a resolver el siguiente problema de optimización:

$$\begin{aligned} \epsilon_t = \max_y \quad & \sum_{i=1}^{t-1} \alpha_{j_i} y_{j_i} + \sum_{j \in C} y_j \\ \text{s.a.} \quad & \sum_{i=1}^{t-1} a_{j_i} y_{j_i} + \sum_{j \in C} a_j y_j \leq b - a_{j_t} \\ & y \in \{0, 1\}^{|C|+t-1}. \end{aligned} \quad (3.14)$$

El problema (3.14) es un problema de tipo (PEB) que se puede resolver con los métodos estándares de programación lineal entera como el de ramificación y acotación visto en la Sección 1.3.

Si denotamos por ϵ_t^* al valor objetivo óptimo del problema (3.14), tendríamos que $\alpha_{jt} = |C| - 1 - \epsilon_t^*$.

Este procedimiento se repetirá para la iteración $t + 1$. Nótese que cuando $t = 1$, el procedimiento descrito es análogo pero eliminando el sumando

$$\sum_{i=1}^{t-1} \alpha_{ji} y_{ji} \quad (3.15)$$

donde aparezca.

El pseudocódigo del procedimiento “sequential lifting” puede verse en el Algoritmo 2:

Algoritmo 2 Sequential Lifting iteración $t \in \{1, \dots, r\}$

Entrada: $t, \alpha_{j_1}, \dots, \alpha_{j_{t-1}}$

1: Resolver el problema (3.14), que da como valor objetivo ϵ_t^*

2: $\alpha_{jt} = |C| - 1 - \epsilon_t^*$

Salida: α_{jt}

Veamos, a continuación, un ejemplo ilustrativo de este procedimiento, [24]:

Ejemplo 3.6. Consideramos el cubrimiento minimal obtenido en el Ejemplo 3.1, recordemos $C_1 = \{3, 4, 5, 6\}$. De este modo, $P \setminus C = \{1, 2, 7\}$ y, siguiendo el procedimiento anteriormente descrito, $j_1 = 1, j_2 = 2$ y $j_3 = 7$. En el Ejemplo 3.5 calculamos el coeficiente $\alpha_1 = 2$. Continuamos, por lo tanto, con la variable y_2 .

La desigualdad más fuerte calculada hasta el momento es $2y_1 + y_3 + y_4 + y_5 + y_6 \leq 3$. Queremos encontrar el valor más grande de α_2 para el cual la desigualdad $\alpha_2 y_2 + 2y_1 + y_3 + y_4 + y_5 + y_6 \leq 3$ es válida para Y . Para ello, resolvemos el siguiente problema de la forma (PEB):

$$\begin{aligned} \epsilon_2 = \max_y \quad & 2y_1 + y_3 + y_4 + y_5 + y_6 \\ \text{s.a.} \quad & 11y_1 + 6y_3 + 5y_4 + 5y_5 + 4y_6 \leq 19 - a_2 = 19 - 6 = 13 \\ & y_j \in \{0, 1\}, \quad j = 1, 3, 4, 5, 6. \end{aligned} \quad (3.16)$$

Resolviendo el problema obtenemos el máximo $\epsilon_2^* = 2$ en el punto $(1, 0, 0, 0, 0)$ y, por lo tanto, $\alpha_2 = |C| - 1 - \epsilon_2^* = 4 - 1 - 2 = 1$. Análogamente, calculamos $\epsilon_7^* = 3$ y así, $\alpha_7 = |C| - 1 - \epsilon_7^* = 4 - 1 - 3 = 0$. Por lo tanto, la desigualdad de cubrimiento levantada para Y es:

$$2y_1 + y_2 + y_3 + y_4 + y_5 + y_6 \leq 3. \quad (3.17)$$

Para finalizar este capítulo, veamos la siguiente proposición, cuya demostración aparece en [5], con la que podemos afirmar que las desigualdades de cubrimiento levantadas obtenidas con el proceso anteriormente descrito son facetas de $\text{conv}(Y)$, [5].

Proposición 3.4. Considerese un conjunto $S \subseteq \{0, 1\}^p$ tal que $S \cap \{y : y_p = 1\} \neq \emptyset$, y sea $\sum_{j=1}^{p-1} \alpha_j y_j \leq \beta$ una desigualdad válida para $S \cap \{y : y_p = 0\}$. Entonces,

$$\alpha_p = \beta - \max \left\{ \sum_{j=1}^{p-1} \alpha_j y_j : y \in S, y_p = 1 \right\} \quad (3.18)$$

es el coeficiente más grande que hace que $\sum_{j=1}^{p-1} \alpha_j y_j + \alpha_p y_p \leq \beta$ sea válida para S . Además, si $\sum_{j=1}^{p-1} \alpha_j y_j \leq \beta$ define una faceta d -dimensional de $\text{conv}(S) \cap \{y_p = 0\}$, entonces $\sum_{j=1}^p \alpha_j y_j \leq \beta$ define una faceta de $\text{conv}(S)$ de dimensión al menos $d + 1$.

Capítulo 4

Experimentos computacionales

Este capítulo está dedicado a los experimentos computacionales. Analizaremos de forma empírica la validez de los cortes estudiados en el Capítulo 3 dedicados a las desigualdades válidas del problema de la mochila (3.1). Con el objetivo de que este capítulo sea autocontenido, recordaremos de nuevo la notación utilizada. Representaremos el espacio que ocupa un objeto j en la mochila como a_j , p será el número de objetos que podemos introducir en la mochila y llamaremos b a la capacidad total que admite nuestra mochila. Además, notaremos como c_j al beneficio asociado al objeto j . Recordemos también que el objetivo principal del problema (3.1) es elegir un conjunto de objetos que no exceda la capacidad máxima b , y maximice el beneficio total.

Para el análisis numérico realizado en este capítulo hemos seleccionado 25 instancias del problema de la mochila denotadas en la referencia [17] como *Strong correlated*. Tal y como aparece en [17], estas instancias son perfectas para demostrar la fortaleza de las desigualdades válidas, puesto que son instancias cuya solución óptima del problema relajado está muy lejos de la solución óptima del problema con variables binarias. Es decir, con este tipo de instancias podremos ver que efectivamente las desigualdades válidas vistas en el Capítulo 3 son cortes que reducen la región factible del problema relajado y mejoran la resolución del problema original con variables enteras.

De entre todas las instancias disponibles, se han elegido aquellas en las que hay $p = 100$ elementos, los coeficientes a_i están entre 1 y 1000 y, los coeficientes c_i son de la forma $c_i = a_i + 100$. Por lo tanto, consideramos los datos de los parámetros c_i , a_i y b de 25 problemas de la forma (3.1) con $p = 100$ variables. Todas las instancias pueden descargarse en [15]. En particular, los datos utilizados de las 25 instancias pueden verse en [1].

Para los experimentos computacionales, hemos utilizado Python 3.11.3, [18]. Como herramienta para resolver los problemas de optimización se ha usado el *solver* de optimización Gurobi 9.1, [10] a través de la librería de Python, `gurobipy`. Todos los experimentos se han ejecutado en un ordenador Windows 10 PRO con 4 GB de RAM.

Nótese que el *solver* Gurobi incluye por defecto varios cortes que reducen la región factible del problema a considerar. Por esta razón, y con el objetivo de evaluar correctamente el efecto de las desigualdades válidas en la resolución del problema de la mochila, hemos desactivado los siguientes parámetros de Gurobi relacionados con la introducción automática de cortes en procedimiento de ramificación y acotación: “Cuts” y “Presolve”. Además, hemos establecido un tiempo límite de 60 segundos para la resolución de cada problema. El resto de parámetros de Gurobi se han dejado en su opción por defecto.

Todos los detalles sobre la implementación de este capítulo pueden verse en [1].

A continuación, pasaremos a analizar en dos experimentos distintos el efecto de las desigualdades válidas en el problema de la mochila.

4.1. Experimento 1: Problema de la mochila con y sin desigualdades de cubrimiento

En esta sección, estudiaremos las mejoras computacionales que se producen entre resolver el problema de la mochila usando su formulación original, y aplicando las desigualdades válidas de cubrimiento más sencillas vistas en (3.4). Estas mejoras las mediremos en términos del número de nodos explorados en el árbol creado durante el procedimiento de ramificación y acotación. Para ello, hemos resuelto, por un lado las 25 instancias seleccionadas usando la formulación (3.1). Por otro lado, también se han resuelto 25 instancias del problema de la mochila (3.1) al que se le han añadido la restricción (3.4) para distintos cubrimientos.

Las Tablas A.1 y A.2 muestran, respectivamente, los resultados obtenidos tras resolver los primeros 25 problemas con la formulación original, y los 25 siguientes donde se han añadido las restricciones de cubrimiento (3.4).

En dichas tablas puede observarse el identificador de la instancia resuelta, el número de nodos, el tiempo de resolución (en segundos), el gap obtenido y el valor objetivo encontrado en el tiempo límite de 60 segundos.

Nótese que para resolver la formulación reforzada del problema de la mochila donde se añaden las desigualdades válidas del tipo (3.4), es necesario primero determinar qué subconjuntos de productos de entre los $p = 100$ disponibles son efectivamente cubrimientos, de acuerdo a la Definición 3.1.

Una opción sería enumerar todos los posibles conjuntos de q elementos con q variando de 1 a 100. Es decir, que para q fijo tendríamos $\binom{100}{q}$ conjuntos que podrán ser cubrimiento. Verificando para todo q si cada uno de esos $\binom{100}{q}$ conjuntos es cubrimiento o no, tendríamos todos los cubrimientos posibles para los datos dados. Para valores pequeños de p (y por tanto, de q) esta tarea es asequible. Sin embargo, si p es suficientemente grande, como es nuestro caso, esta tarea tiene un alto coste computacional.

Es por ello, que hemos propuesto una aproximación donde obtenemos algunos de los cubrimientos posibles. De hecho, por construcción los cubrimientos obtenidos son minimales (ver Definición 3.1). Pasemos a analizar cómo hemos construido dichos cubrimientos.

Para cada una de las 25 instancias, consideramos el conjunto de las capacidades de los productos a_i , $i = 1, \dots, 100$. Por simplicidad, hemos decidido crear un máximo de $U = 40$ cubrimientos. Inicialmente, el cubrimiento C^u está vacío, $u = 1, \dots, U$. Comenzamos añadiendo a C^u el producto u . Si $a_u > b$, hemos acabado. Si no, es decir, si $a_u \leq b$, se van añadiendo consecutivamente los elementos $u+1$, $u+2$, $u+3$, etc, siempre comprobando en cada iteración si la suma de las capacidades de los elementos añadidos hasta el momento supera o no la capacidad máxima b .

Tal y como hemos mencionado anteriormente, por construcción, los cubrimientos obtenidos son minimales.

Pasamos ahora a comentar los resultados obtenidos. Comenzamos con la Figura 4.1. En ella, observamos un histograma en el que se representa la frecuencia absoluta respecto al tiempo de ejecución en segundos en el que se han resuelto los 25 problemas de tipo (3.1), sin desigualdades de cubrimiento. Podemos ver que la mayoría de las instancias se resuelven en menos de 20 segundos. A estas instancias las llamaremos *instancias fáciles*.

Por otro lado, aquellas instancias cuyo tiempo de ejecución está entre 50 y 60 segundos las denotaremos por *instancias difíciles*.

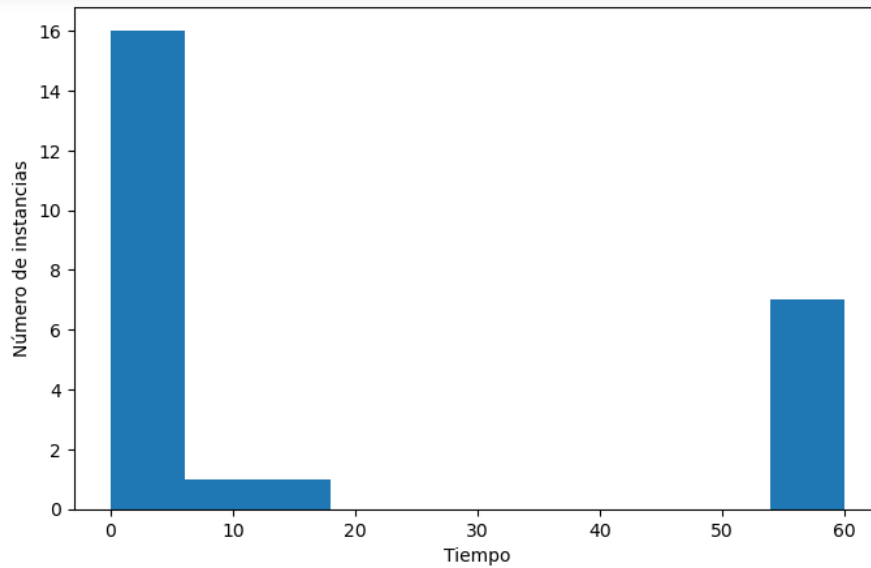


Figura 4.1: Histograma del tiempo de resolución de los 25 problemas de la forma (3.1).

En base a esto, hemos decidido estudiar si las desigualdades de cubrimiento mejoran estos problemas “difíciles” de resolver, en términos de número de nodos recorridos por el algoritmo de ramificación y acotación.

En la Figura 4.2, se muestran dos diagramas de cajas y bigotes, en los que se puede observar la distribución de los datos referentes al número de nodos recorridos por el algoritmo de ramificación y acotación en los problemas que tardan más de 50 segundos en resolverse. Para que la comparación entre ambas formulaciones sea justa, se ha comprobado que en todas las instancias, el valor objetivo final coincide y, por lo tanto, tiene sentido que comparemos el número de nodos obtenidos en ambas formulaciones. En la Figura 4.2a corresponde a los datos asociados al problema con la formulación original (3.1). En el diagrama que aparece en la Figura 4.2b, se muestra la distribución de las *instancias difíciles* referentes a los nodos, asociados a los mismos problemas pero con las desigualdades de cubrimiento (3.4) en su formulación. Es fácil ver que el número de nodos recorridos al usar la formulación con las desigualdades de cubrimiento es considerablemente menor que si se usa la formulación original. Por lo tanto, hemos comprobado de forma empírica que las desigualdades válidas de cubrimiento ayudan a resolver el problema de la mochila.

4.2. Experimento 2: Problema de la mochila con desigualdades de cubrimiento y con desigualdades de cubrimiento extendidas

En la Sección 3.2, definimos un tipo de desigualdades de cubrimiento más fuertes que las anteriormente vistas, las desigualdades de cubrimiento extendidas. En esta sección, mediante un experimento computacional, estudiaremos si existen mejoras significativas en la resolución del problema de la mochila en función de si se añaden las restricciones

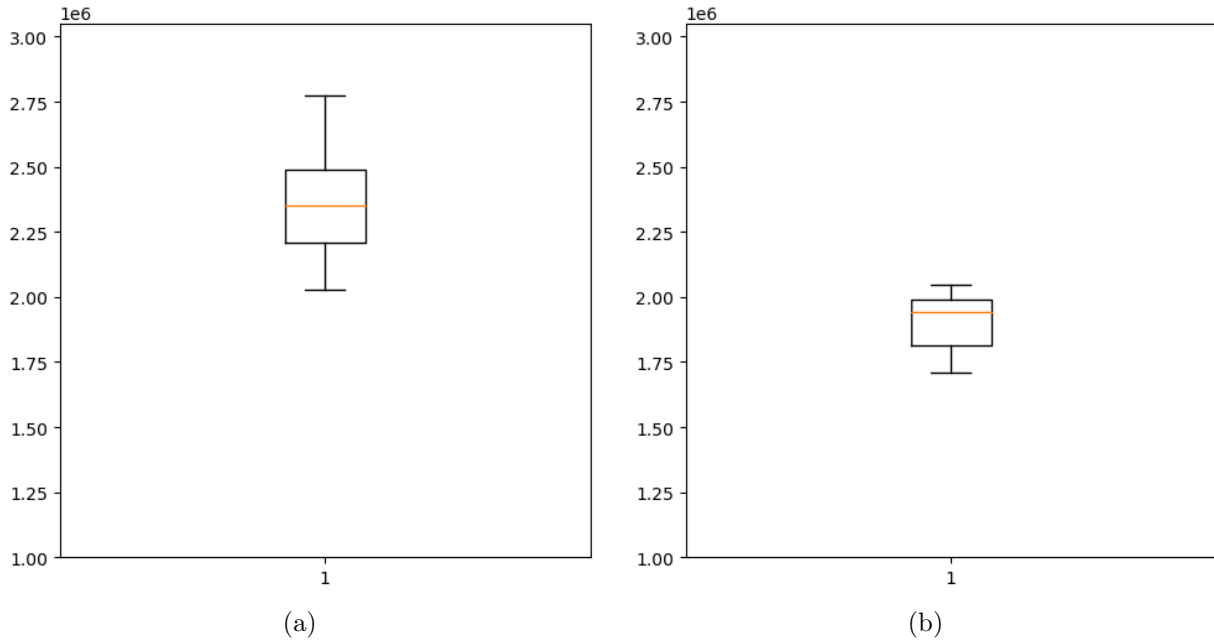


Figura 4.2: Distribución del número de nodos explorados entre los problemas sin desigualdades de cubrimiento en su formulación, es decir, con la formulación (3.1), (a) y con desigualdades de cubrimiento (3.4), (b).

de cubrimiento (3.4) de la Sección 4.1 o las restricciones asociadas a las desigualdades de cubrimiento extendidas (3.7).

Al igual que se hizo en la Sección 4.1, nuestro experimento consiste en resolver el problema con desigualdades de cubrimiento extendidas y comparar los resultados, en término de nodos explorados por el algoritmo, con los obtenidos en la sección anterior en los problemas resueltos con cortes de cubrimiento en su formulación.

Es por lo tanto necesario, construir los cubrimientos extendidos. Para ello, partiremos de los cubrimientos construidos anteriormente. Fijado un cubrimiento C , añadiremos aquellos productos j tales que $a_j \geq a_i$, para todo $i \in C$, tal y como indica la Proposición 3.3.

Para cada uno de estos cubrimientos extendidos, crearemos una restricción del tipo (3.7) que añadiremos a la formulación (3.1).

Este proceso se repetirá en cada una de las 25 instancias estudiadas. Una vez que se han resuelto estas 25 instancias, pasamos a analizar los resultados obtenidos, comparándolos con las 25 instancias donde las desigualdades válidas que se han añadido son del tipo (3.4). Los resultados obtenidos de esta nueva resolución de los problemas con desigualdades de cubrimiento extendidas aparecen en la Tabla A.3.

Por simplicidad, hemos comparado las instancias “difíciles” en las que el algoritmo de ramificación y acotación tarda más de 50 segundos en ejecutar las instancias con la formulación original. En la Figura 4.3, se puede observar la comparación de la distribución de los datos referentes al número de nodos explorados por el algoritmo mediante dos diagramas de cajas y bigotes. La Figura 4.3a corresponde a los datos de los problemas con desigualdades de cubrimiento (3.4) en la formulación (3.1) y, el diagrama de la Figura 4.3b corresponde a los problemas con desigualdades de cubrimiento extendidas (3.7).

De esta forma, observando los resultados que se muestran en la Figura 4.3, podemos

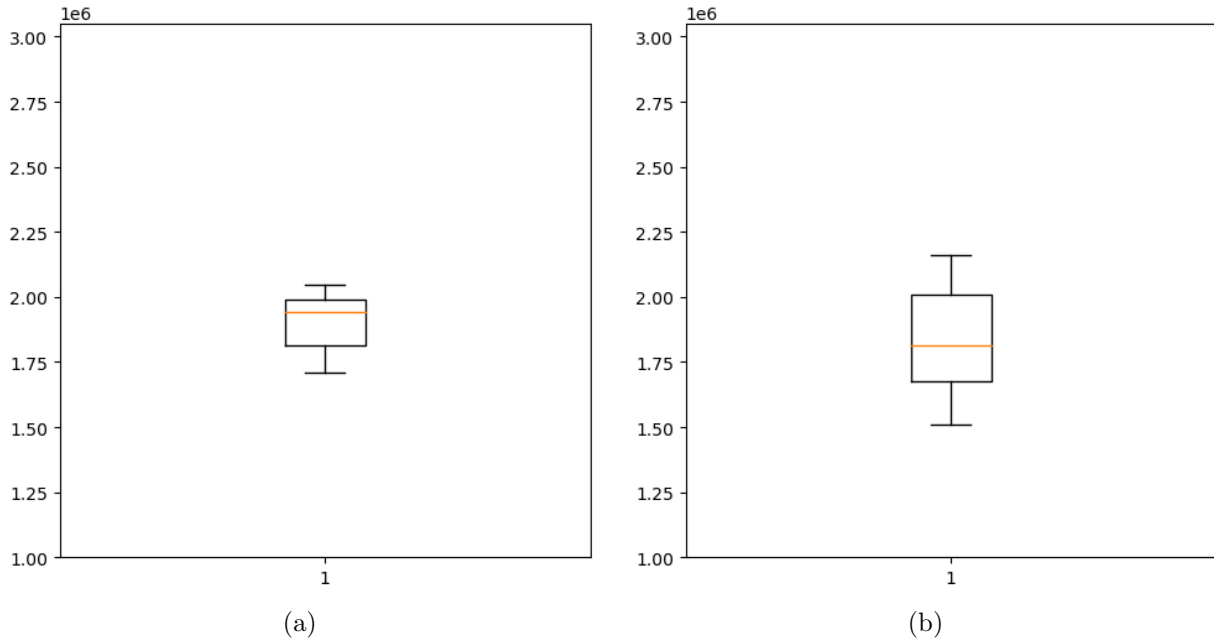


Figura 4.3: Distribución del número de nodos explorados entre los problemas con desigualdades de cubrimiento (3.4) en su formulación, (a) y con desigualdades de cubrimiento extendidas (3.7), (b).

concluir que el número de nodos explorados por el algoritmo es, en general, menor en los problemas con desigualdades de cubrimiento extendidas (3.7) que en los problemas con desigualdades de cubrimiento (3.4). Por ejemplo, la mediana del diagrama de la derecha es menor que en el diagrama de la izquierda. Luego, en general, podemos concluir que las desigualdades de cubrimiento extendidas (3.7) benefician a la resolución del problema de la mochila frente al uso de desigualdades de cubrimiento (3.4).

No obstante, nos gustaría observar que el bigote superior de la Figura 4.3b tiene un valor más grande que el bigote superior de la Figura 4.3a. Es decir, a priori, puede parecer que en dicha instancia los resultados empeoran con la formulación de cubrimiento extendida. Dicha instancia es la que tiene el identificador 4 en las Tablas A.2 y A.3. En la Tabla A.2, vemos que el número de nodos de esa instancia es de 2.163.017, mientras que en la Tabla A.3 el número de nodos explorado es 1.941.617. Es decir, el número de nodos aumentó cuando añadimos las desigualdades de cubrimiento extendidas en vez de las de cubrimiento simples. Sin embargo, queremos resaltar que el gap en el caso de los cubrimientos extendidos es mejor que en los cubrimientos simples. Es decir, la cota del problema relajado durante la ramificación y acotación es mejor para el problema con desigualdades de cubrimiento extendidas. Por lo tanto, en este caso también hemos comprobado de forma empírica que las desigualdades de cubrimiento extendidas (3.7) son mejores que las de cubrimiento (3.4) para el problema de la mochila.

Capítulo 5

Conclusiones

El objetivo principal de este trabajo ha sido motivar y profundizar en la teoría matemática que sustenta una herramienta fundamental para la resolución de problemas de programación lineal con variables enteras, que son las desigualdades válidas o cortes. Además, se han realizado varios experimentos computacionales para un problema específico, el problema de la mochila. Con estos experimentos podemos observar la eficiencia y gran utilidad de dichas desigualdades en la práctica.

En la parte teórica de este trabajo, hemos formalizado el concepto de desigualdad válida. Además, hemos desarrollado varias desigualdades válidas para tres tipos de problemas de programación lineal en enteros, que son (PE), (PEB) y (PEM). Además, hemos utilizado la estructura de un problema de tipo (PEB) particular como es el problema de la mochila, para desarrollar desigualdades válidas más fuertes y eficientes.

Por otro lado, hemos dedicado el último capítulo del trabajo a la parte práctica en la que llevamos a cabo distintos experimentos computacionales para analizar de forma empírica la validez y eficacia de los cortes estudiados para el problema de la mochila.

Apéndice A

Tablas

En este apéndice, se recogen las tablas de resultados obtenidas en los experimentos del Capítulo 4.

Id	nodos	tiempo	gap	objetivo
0	9132	0.60899	0	35617
1	73966	1.88999	0	32291
2	22798	0.94599	0	34181
3	13955	1.17100	8.89917e-05	33711
4	2771602	60.0159	0.00022	30981
5	43	0.06599	9.37196e-05	31841
6	10858	0.51999	0	32272
7	5001	0.29600	0	30405
8	61683	1.84100	3.33255e-05	30007
9	64044	2.28800	0	29523
10	2202498	60.06399	0.00152	30847
11	4659	0.56299	9.80007e-05	30612
12	35377	1.69000	0	29986
13	2351524	60.04499	0.00115	31222
14	2454028	60.02899	0.00131	29573
15	18	0.06200	0	32416
16	23	0.06200	3.37086e-05	29666
17	491411	12.55200	8.95121e-05	33515
18	2026479	60.04600	0.00191	31310
19	13280	0.92999	0	29700
20	2210685	60.00499	0.0017	29177
21	16774	0.88899	0	32470
22	21166	1.13599	0	34263
23	562269	12.24399	9.36651e-05	32029
24	2519743	60.01400	0.00180	33284

Tabla A.1: Datos correspondientes a los problemas sin desigualdades de cubrimiento en su formulación, es decir, a la formulación original del problema de la mochila (3.1).

Id	nodos	tiempo	gap	objetivo
0	9132	0.57799	0	35617
1	73298	1.65599	3.09683e-05	32291
2	22547	0.96899	5.85120e-05	34181
3	13955	1.56399	8.89917e-05	33711
4	1941617	60.02099	0.00064	30981
5	23	0.08499	9.42151e-05	31842
6	10858	0.66700	0	32272
7	5001	0.38099	0	30405
8	61699	1.55700	3.33255e-05	30007
9	64048	1.59999	0	29523
10	1711734	60.02900	0.00158	30847
11	4694	0.38700	6.53338e-05	30612
12	35325	1.29699	0	29986
13	1718949	60.04500	0.00124	31222
14	1946204	60.02900	0.00142	29573
15	6	0.04699	9.25468e-05	32416
16	23	0.07800	3.37086e-05	29666
17	492722	9.55500	8.95121e-05	33515
18	2048428	60.00500	0.00191	31310
19	13281	0.60199	0	29700
20	1909434	60.00900	0.00178	29177
21	16584	0.96799	9.23929e-05	32470
22	21100	0.71900	2.91860e-05	34263
23	561056	14.18099	9.36651e-05	32029
24	2034371	60.01600	0.00183	33284

Tabla A.2: Datos correspondientes a los problemas con desigualdades de cubrimiento (3.4) en su formulación.

Id	nodos	tiempo	gap	objetivo
0	9132	1.12800	0	35617
1	73298	3.20000	3.09683e-05	32291
2	22582	0.88800	5.85120e-05	34181
3	13891	0.62199	8.89917e-05	33711
4	2163017	60.02999	0.00051	30981
5	23	0.06299	9.42151e-05	31842
6	10818	0.66600	3.09866e-05	32272
7	4953	0.49800	0	30405
8	62544	1.84400	0	30007
9	64139	1.88800	0	29523
10	1511141	60.00800	0.00162	30847
11	4694	0.47200	3.26669e-05	30612
12	35328	1.08700	0	29986
13	1688032	60.02699	0.00124	31222
14	1666054	60.02700	0.00145	29573
15	18	0.05400	0	32416
16	23	0.06200	3.37086e-05	29666
17	491631	11.59999	8.95121e-05	33515
18	1863867	60.01099	0.00194	31310
19	13753	0.52200	0	29700
20	1814089	60.03599	0.00181	29177
21	16751	0.78199	0	32470
22	19919	0.85899	0	34263
23	562125	14.72000	9.36651e-05	32029
24	2153890	60.03500	0.00183	33284

Tabla A.3: Datos correspondientes a los problemas con desigualdades de cubrimiento extendidas (3.7) en su formulación.

Bibliografía

- [1] L. Albacete-Maza. knapsack covers. https://github.com/lourdesalb/knapsack_covers, 2023.
- [2] M. F. Balcan, T. Dick, T. Sandholm, and E. Vitercik. Learning to branch. In *Proceedings of Machine Learning Research*, 2018.
- [3] S. Bradley, A. Hax, and T. Magnanti. *Applied Mathematical Programming*. 1977.
- [4] D. Chen, R. Batson, and Y. Dang. *Applied Integer Programming: Modeling and Solution*. 2010.
- [5] M. Conforti, G. Cornuéjols, and G. Zambelli. *Integer Programming*. 2014.
- [6] G. Cornuéjols. Valid inequalities for mixed integer linear programs. *Mathematical Programming*, 2007.
- [7] D. Davendra. *Traveling Salesman Problem: Theory and Applications*. 2010.
- [8] Blog de Gestión de Operaciones. Formulación de un problema de localización y transporte (programación entera mixta). <https://www.gestiondeoperaciones.net/programacion-entera/formulacion-de-un-problema-de-localizacion-y-transporte-programacion-entera-mixta/>. [Online; accessed 30-April-2023].
- [9] C. Domínguez-Sánchez. *El Problema de Tarifación Basado en Preferencias*. PhD thesis, Universidad Libre de Bruselas and Universidad de Murcia, 2021.
- [10] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2023.
- [11] F. Hillier and G. Lieberman. *Introducción a la Investigación de Operaciones*, volume 9. 2010.
- [12] J. Hooker and V. Vinay. Branching rules for satisfiability. *Journal of Automated Reasoning*, 1995.
- [13] T. Kleinert, M. Labbé, M. Schmidt, and F. Plein. There is no free lunch: on the hardness of choosing a correct big-M in bilevel optimization. *Operations Research*, 2020.
- [14] G. Nemhauser and L. A. Wolsey. *Integer and Combinatorial Optimization*. 1988.
- [15] Y. Onoue. Test instances for knapsack problems. <https://github.com/likr/kplib>, 2019.

- [16] S. Pineda and J. M. Morales. Solving linear bilevel problems using big-Ms: not all that glitters is gold. *IEEE Transactions on Power Systems*, 2019.
- [17] D. Pisinger. Where are the hard knapsack problems? *Computers and Operations research*, 32, 2004.
- [18] G. V. Rossum and F. L. Drake. *Python 3 Reference Manual*. CreateSpace, 2009.
- [19] G. Sierksma and Y. Zwols. *Linear and Integer Optimization*. 2015.
- [20] F. Trespalacios and I. E. Grossmann. Improved big-m reformulation for generalized disjunctive programs. *Computers and Chemical Engineering*, 2015.
- [21] UC3M. Asignatura de investigación operativa en la universidad uc3m. https://halweb.uc3m.es/fjnm/io/doc_generica/archivos/PE.pdf, 2006/2007. [Online; accessed 06-June-2023].
- [22] UMA. Apuntes de la asignatura de optimización en la Universidad de Málaga, 2020/2021.
- [23] W. L. Winston. *Investigación de operaciones: aplicaciones y algoritmos*. 1987.
- [24] L. A. Wolsey. *Integer Programming*. 2021.