UNIVERSITAT POLITÈCNICA DE CATALUNYA
Facultat d'Informàtica

DISEÑO Y ESTUDIO COMPUTACIONAL DE ALGORITMOS HÍBRIDOS PARA PROBLEMAS DE SET PARTITIONING

Memoria presentada por Elena Fernández Aréizaga, bajo la dirección del Profesor Jaime Barceló Bugeda, para acceder al grado de Doctor en Informática por la Universitat Politècnica de Catalunya.

Barcelona
Enero, 1988

A todos los que no han terminado todavía

Quiero agradecer a ...

Jaime Barceló, que me inició en la Programación Matemática y, posteriormente, aceptó dirigir este trabajo, por su paciencia y dedicación.

JK, una vez más, por su afecto y optimismo constantes. Su confianza en la terminación de este trabajo ha sido, a menudo, muy superior a la mía.

Todas las personas de la Facultad de Informática de Donosti que, a pesar de la distancia, me han seguido y apoyado todo este tiempo.

Las personas que han trabajado en el LCFIB durante los tres últimos años, por su cariño y palabras de ánimo permanentes. La tolerancia del equipo de sistema y el formidable entorno que entre todos han creado ha sido fundamental para la terminación de esta tesis. En especial, a África, Isabel y Rosa.

Todos aquellos que, en algún momento, dedicaron parte de su tiempo y paciencia a encontrar una restricción agregada. Aunque posteriormente comprobé que su trabajo era inútil, debido a un error mío de planteamiento, no por ello su ayuda resultó menos de agradecer. También a Andrés, cuyas sugerencias no he seguido por completo, a pesar de que me han sacado de más de un atolladero.

Los magníficos amigos, que son muchos, que en algún momento han sido generosos al compartir su tiempo conmigo, comprendiéndome, animándome y acompañándome. Especialmente a Vera.

Pilar y Víctor que, además, han participado activamente en la composición del texto final. Víctor, como asesor de estilo (asumo todas las culpas ya que no le he hecho mucho caso) y detectando erratas, y Pilar, en la pesadísima tarea de formatear las páginas y completar los detalles finales.

Fernando que, además, me ha enseñado lo que es un "tipo abstracto".

ÍNDICE

CAPÍTULO 0

INTRODUCCIÓN

Dentro de la programación matemática, los problemas en los que las variables que intervienen son enteras resultan actualmente de especial interés. Su importancia radica fundamentalmente en el hecho de que la formulación matemática de gran número de situaciones reales responde a este tipo de problemas.

Los problemas enteros requieren, además, unos procedimientos específicos para su resolución, ya que los métodos clásicos aplicables a los problemas con variables continuas no los resuelven, puesto que las soluciones óptimas de los problemas enteros no son vértices del poliedro definido por el conjunto de restricciones.

La dificultad para resolver los problemas enteros se manifiesta fácilmente observando que, a pesar de que el número de soluciones posibles para los mismos es finito, dicho número crece exponencialmente respecto al número de variables que intervienen en los problemas. Por este motivo, cualquier método de resolución que considere todas las posibles soluciones, proporcionará una solución óptima, pero su eficiencia computacional resultará muy pobre.

En este sentido, resulta fundamental disponer de métodos aproximados de resolución para los problemas enteros, que proporcionen buenas soluciones posibles y que resulten computacionalmente rentables.

Desde el punto de vista teórico son, sin duda, las interpretaciones geométricas de los problemas las que han proporcionado perspectivas valiosas para el estudio de los mismos: inicialmente, la teoría de planos secantes y, posteriormente, la combinatoria poliédrica y programación disyuntiva. En los años 50, la teoría de los planos secantes de Dantzig [Dan59], que posteriormente fue generalizada por Gomory [Gom63], abrió una primera vía para resolver este tipo de problemas. Con estos métodos se obtienen desigualdades válidas para los problemas, las cuales son violadas por la solución óptima de los problemas lineales asociados, pero que son satisfechas por todas las soluciones enteras.

La combinatoria poliédrica y la programación disyuntiva son las teorías que actualmente ofrecen expectativas más prometedoras. La combinatoria poliédrica [Grö81] se ocupa de caracterizar los polítopos definidos por la envolvente convexa del conjunto de soluciones posibles de los problemas enteros. Si se consigue caracterizar mediante un sistema de desigualdades lineales el polítopo así definido asociado a un problema entero, entonces será posible aplicar el método del Simplex al sistema obtenido, puesto que la solución óptima del problema entero sí será un vértice de este poliedro. La existencia de un único sistema lineal que caracteriza el polítopo asociado a un problema entero está asegurada por el Teorema de Weyl [Wey35], pero desgraciadamente, en muy pocos casos se dispone de procedimientos que proporcionen, ya no todas, sino alguna de las desigualdades lineales (facetas) que definen este sistema.

La programación disyuntiva [Bal69] procura obtener formulaciones equivalentes de los problemas enteros como problemas en los que el conjunto de restricciones viene dado por una familia de disyunciones. Dicha familia se obtiene a partir de las condiciones lógicas que vienen implicadas por el conjunto original de restricciones. Estas disyunciones permiten, en algunos casos, obtener desigualdades válidas para los problemas, que son las que se pueden utilizar algorítmicamente.

Hay que resaltar, sin embargo, que los resultados teóricos obtenidos gracias a la combinatoria poliédrica y la programación disyuntiva no siempre resultan aplicables desde una perspectiva algorítmica, puesto que los procedimientos derivados resultan, en ocasiones, de una complejidad similar a la de la resolución de los problemas originales.

En este contexto, el diseño de heurísticas que proporcionen soluciones posibles para los problemas resulta especialmente sugerente. Las cotas asociadas al valor de la función objetivo para estas soluciones posibles pueden resultar de gran utilidad en procedimientos enumerativos, puesto que permitirán eliminar gran cantidad de nodos del árbol de exploración. Asimismo, a partir de estas soluciones posibles, en ocasiones, pueden generarse desiguladades válidas para los problemas. No olvidemos, además, que en ciertas situaciones será suficiente para nuestro objetivo práctico disponer de una buena solución.

Por otra parte debemos resaltar que los procedimientos heurísticos, al ser en general muy sencillos, resultan computacionalmente rentables.

En los últimos años se observa una tendencia al diseño de algoritmos híbridos que combinen de una forma inteligente todas las herramientas capaces de proporcionar información importante para la resolución de los problemas. En general este tipo de algoritmos utilizan heurísticas para obtener soluciones posibles (cotas superiores en el caso de problemas de minimización), algún tipo de relajación asociada a los problemas para obtener cotas inferiores (en el caso de problemas de minimización), planos secantes y, posiblemente, enumeración implícita.

Una familia especial de problemas enteros son aquellos en los que todas las variables que intervienen pueden interpretarse como variables de decisión; es decir, están restringidas a tomar los valores 0-1. La importancia de estos problemas, además de sus numerosas aplicaciones prácticas, radica en el hecho de que cualquier problema con variables enteras generales puede formularse como problemas de esta familia. Dentro de esta familia los problemas de set partitioning, junto con los de set packing y set covering, forman una clase muy especial.

Existen problemas con variables 0-1 tales como el del travelling salesman, el de knapsack, o el de set packing, para los cuales están caracterizadas gran parte de sus facetas. A diferencia de ellos, apenas se tiene conocimiento sobre el polítopo de set covering y especialmente sobre el de set partitioning. En particular, el cálculo de la dimensión del polítopo de set partitioning resulta ser un problema NP completo. Por ese motivo, el estudio de este problema se suele enfocar desde una de las dos perspectivas siguientes: caracterización de facetas de alguna de sus relajaciones o generación de desigualdades válidas para el problema ya sean planos secantes o desigualdades válidas a partir de programación disyuntiva.

En este trabajo se estudian los problemas de set partitioning desde una perspectiva algorítmica. El objetivo que se persigue es el diseño de una clase de algoritmos híbridos que alcancen un compromiso entre la calidad de las soluciones obtenidas y el esfuerzo computacional requerido. Se analiza la única heurística conocida para los problemas de set partitioning de maximización y se propone una para minimización. Asimismo, se propone la utilización de ciertas desigualdades válidas a partir de cotas condicionales (programación disyuntiva), que hasta ahora sólo habían sido utilizadas en el contexto de problemas de set covering y de travelling salesman. También se analizan y proponen distintas alternativas de relajaciones surrogadas, lagrangianas, y la utilización de un método de refuerzo dual.

La incorporación a los problemas originales de las desigualdades válidas obtenidas, da lugar a unos problemas en los que el conjunto de restricciones está formado por una familia de desigualdades de tipo set covering y una familia de igualdades tipo set partitioning. Para estos nuevos problemas también se propone una heurística para obtener soluciones posibles, se justifica la validez y se propone la utilización de las desigualdades válidas obtenidas a partir de cotas condicionales. Asimismo, para los problemas ampliados se estudian una serie de métodos duales, análogos a los ya citados para un problema de set partitioning puro.

Algunas de las relajaciones lagrangiana que se proponen tienen una estructura de problemas de knapsack con dos restriciones. En particular, aparecen unos problemas en los que las dos restricciones son desigualdades del mismo sentido y otros en los que las dos restricciones son de sentido contrario. Estos dos tipos de problemas se han estudiado en detalle. Para ambas versiones se proponen cotas inferiores, heurísticas y algoritmos de enumeración implícita.

Finalmente se propone una clase de algoritmos híbridos para los problemas de set partitioning en los que se utilizan los siguientes procedimientos: heurísticas primales y duales, planos secantes y algún método dual. Se analiza con detalle la eficiencia de los distintos procedimientos a partir de los resultados obtenidos en una amplia experiencia computacional. Para ello se ha utilizado una batería de problemas de distintas dimensiones generados aleatoriamente.

Esta tesis se estructura en cinco capítulos, cada uno de los cuales se divide en secciones que a su vez pueden contener distintos apartados. En el primer capítulo se realiza una introducción a los problemas de set partitioning así como un resumen de los principales resultados teóricos relacionados con los mismos y un análisis de las distintas alternativas algorítmicas que se han utilizado para su resolución. En el segundo capítulo se presentan las distintas heurísticas y desigualdades válidas obtenidas a partir de cotas condicionales tanto para los problemas de set partitioning como para los nuevos problemas ampliados que aparecen al incorporar las desigualdades obtenidas a los problemas originales. En el tercer capítulo se estudian distintos métodos duales aplicables tanto a los problemas de set partitioning puros como a los problemas ampliados. En el cuarto capítulo se estudian los problemas de knapsack con dos restricciones que ya se han mencionado y finalmente, en el quinto capítulo se propone una clase de algoritmos híbridos para resolver los problemas de set partitioning y se exponen los resultados de las experiencias computacionales realizadas.

CAPÍTULO I

PROBLEMAS DE SET PARTITIONING, SET PACKING Y SET COVERING

I.1. Introducción

Dentro de la programación matemática es bien conocida la importancia de los programas enteros y en particular de aquellos cuyas variables son 0-1. Los problemas de set covering, set packing y set partitioning forman una familia de especial interés dentro de estos últimos; ello se debe, fundamentalmente, al hecho de que existen numerosas situaciones reales cuya formulación matemática puede expresarse en términos de uno de ellos. Por este motivo, en los últimos años y especialmente en la década de los 70, se han realizado numerosos esfuerzos para encontrar algoritmos que resuelvan estos problemas de forma eficiente.

Desde una perspectiva teórica, se trata de problemas con una estructura combinatoria que, a pesar de que su conjunto de soluciones posibles es finito y puede, por tanto, ser enumerado, resultan muy difíciles de resolver ya que todos ellos son NP completos. Desde una perspectiva de aplicación, lo anterior significa que el tiempo de cálculo para realizar la enumeración anterior crece de forma exponencial con la dimensión de los mismos.

Como ya se ha mencionado anteriormente los enfoques desde un punto de vista geométrico han sido los que han proporcionado un mayor conocimiento de las estructuras de todos los problemas enteros dentro de los cuales la familia que estamos considerando no es una excepción. En particular, los distintos planos secantes y desigualdades válidas obtenidos a partir de formulaciones como programas disyuntivos han permitido diseñar algoritmos para resolver estos problemas [Bal74, Bal75, Bal76, Bal79].

Sin lugar a dudas la combinatoria poliédrica ha proporcionado, sin embargo, los resultados más espectaculares para la resolución de los distintos problemas enteros. Gracias al Teorema de Weyl [Wey35] se conoce que es posible caracterizar la envolvente convexa de las soluciones posibles mediante de un sistema lineal de ecuaciones. Por lo tanto el estudio de los polítopos que definen los distintos programas enteros y la caracterización de sus facetas proporciona una herramienta extraordinariamente potente para su resolución.

De los tres problemas que estamos considerando, el polítopo asociado a los problemas de set packing es el que ha sido estudiado con más detalle y del que existe mayor número de familias de facetas que ya están caracterizadas [Pad73, Pa74, Pad79, Bapa79, NeTr74]. Se conocen también algunas familias de facetas para los problemas de set covering [BaNg85, Sas85]. Desgraciadamente, no ocurre lo mismo en el caso de los problemas de set partitioning de los que ni siquiera puede calcularse la dimensión del polítopo asociado, ya que este problema resulta ser también NP completo.

Lo anterior no invalida, en absoluto, la potencia de la combinatoria poliédrica como herramienta algorítmica en el estudio de estos problemas ya que, como veremos posteriormente, las facetas de los polítopos asociados a distintas relajaciones del problema de set partitioning pueden definir desigualdades válidas para el problema original que, en ocasiones, serán más potentes que ciertos planos secantes y desigualdades válidas obtenidos específicamente para los mismos.

La formulación de la familia de problemas a la que nos estamos refiriendo es la siguiente:

Definición 1: Sean $M=\{1,...,m\}$, $N=\{1,...,n\}$ y una familia P de subconjuntos de M, $P=\{P_j\}_{j\in N}$.

Un subconjunto $N^*\subseteq N$ se dice que es una *subpartición* de M si

$$\bigcup_{j\in N^*} P_j \subseteq M$$

y además $P_j\cap P_k=\emptyset \; \forall j,k\in N^*, j\neq k$.

Un subconjunto $N^*\subseteq N$ se dice que es una *partición* de M si

$$\bigcup_{j\in N^*} P_j = M$$

y además $P_j\cap P_k=\emptyset \; \forall j,k\in N^*, j\neq k$.

Un subconjunto $N^*\subseteq N$ se dice que es un *recubrimiento* de M si

$$\bigcup_{j\in N^*} P_j \supseteq M$$

Supongamos que cada elemento $j\in N$ tiene asociado un coste c_j no negativo, entonces el problema de set packing consiste en hallar la subpartición de M de peso máximo, el de set partitioning en encontrar la partición de M de peso mínimo y el de set covering en buscar el recubrimiento de M de peso mínimo.

Gran variedad de problemas de programación matemática y en general todos los problemas de la familia que estamos considerando responden al siguiente planteamiento:

Dados i) Un conjunto finito M

ii) Un conjunto de restricciones que define una familia $P=\{P_j\}_{j\in N}$ de subconjuntos 'aceptables' de M y

iii) Un coste asociado a cada elemento de la familia P,

encontrar una colección de elementos de P de coste mínimo que sea una partición de M.

Definiendo una matriz A de dimensiones m x n cuyos elementos a_{ij} son tales que

$$a_{ij} = \begin{cases} 1 \text{ si } i\in P_j \\ 0 \text{ en otro caso} \end{cases}$$

los problemas anteriores pueden formularse como:

Problema de set packing: (SPk) $\min \{ cx \,/\, Ax \leq e,\, x_j\in \{0,1\},\, j\in N \}$
Problema de set partitioning: (SP) $\min \{ cx \,/\, Ax = e,\, x_j\in \{0,1\},\, j\in N \}$
Problema de set covering: (SC) $\min \{ cx \,/\, Ax \geq e,\, x_j\in \{0,1\},\, j\in N \}$

Gran cantidad de problemas que aparecen en la realidad tienen una formulación matemática que se corresponde con alguno de estos tres tipos. A modo de ejemplo, podemos citar los siguientes: asignación de tripulaciones en todo tipo de medios de transporte, recuperación de información, coloreamiento y political districting.

Veamos a continuación algunos ejemplos representativos de situaciones cuya modelización puede venir dada en términos de problemas de set partitioning.

Problema de coloreamiento de grafos

Dado un grafo G, el problema de *coloreamiento del grafo* (graph-colouring problem) (GC) consiste en colorear todos los nodos del mismo con la cantidad mínima de colores de forma que no existan dos nodos adyacentes con el mismo color.

Este tipo de problemas se corresponde con gran variedad de situaciones reales en las que se debe planificar un horario para una serie de actividades que deben de tener lugar, algunas de las cuales no pueden suceder simultáneamente: horarios de exámenes, horarios de competiciones de atletismo, horarios de clases, etc. A pesar de que, evidentemente, podría establecerse una programación en la que ninguno de los eventos tuviese lugar al mismo tiempo, a menudo se desea conseguir una planificación que asegure la menor duración total posible.

Para formular estos problemas se define un grafo en el que cada nodo representa una de las actividades y cada arco (i,j) significa que las actividades i y j no pueden ocurrir simultáneamente. El problema consiste en particionar el conjunto de nodos en la menor colección posible de subconjuntos disjuntos, de forma que no existan dos nodos adyacentes pertenecientes al mismo subconjunto. Es decir, considerando la familia de todos los posibles subconjuntos de actividades que pueden realizarse simultáneamente, se trata de encontrar la subfamilia de estos subconjuntos de cardinalidad mínima que defina una partición del conjunto de nodos.

Para modelizar este problema en términos de un programa de set partitioning se consideran todos los posibles subconjuntos de nodos $S_1,\dots,S_t$ tales que no contengan dos nodos adyacentes y a cada elemento S_j, $j=1,\dots,t$ se le asocia una variable de decisión w_j. La matriz $A=[a_{ij}]$ se define de forma que

$$a_{ij}=\begin{cases} 1, & \text{si el nodo i pertenece al subconjunto } S_j \\ 0, & \text{en otro caso} \end{cases}$$

El problema puede por tanto formularse como

$$\begin{aligned} \min \quad & \sum_{j=1}^{t} w_j \\ \text{t.q.} \quad & \sum_{j=1}^{t} a_{ij} w_j = 1, && i=1,\dots,n \\ & w_j \in \{0,1\}, && j=1,\dots,t \end{aligned}$$

Evidentemente, se trata de un problema de set partitioning.

Problema de itinerario de vehículos

Consideremos ahora un problema en el que un conjunto de 'vehículos' situados en un almacén central debe 'distribuir' unas cantidades conocidas de ciertos productos a unos 'clientes' que están geográficamente dispersos. Suponiendo que todos los recorridos deben empezar y terminar en el depósito central, el *problema de recorrido de vehículos* (vehicle routing problem, vehicle scheduling problem) (VR) consiste en determinar el recorrido óptimo que minimice una cierta función objetivo.

Este problema genérico comprende un gran abanico de situaciones reales como son la recogida de cartas de los buzones, los recorridos de los autobuses escolares, los recorridos de las camionetas de lavanderías, las visitas a domicilio para efectuar las lecturas de gas etc...En cada situación la operación distribuir puede adoptar una forma diferente.

Evidentemente un problema que responde a tantas situaciones diferentes puede dar lugar a gran cantidad de formulaciones distintas. Sin embargo, todas ellas responden a un misma formulación básica:

Sea $X=\{x_1,...,x_n\}$ el conjunto de los clientes y x_0 el depósito central.
Consideremos el conjunto $V=\{v_1,...,v_m\}$ de todos los vehículos existentes.

Un cliente x_i tiene las siguientes características:
1) Una cantidad q_i de un cierto producto que debe distribuirle algún vehículo.
2) Un tiempo u_i, que es el que empleado por el vehículo en 'visitar' al cliente (cargar y/o descargar el producto).
3) Una prioridad δ_i.

Un vehículo v_k tiene las siguientes características:
1) Una capacidad total Q_k de transporte de mercancías.
2) Un horario de trabajo comprendido entre unos instantes T_k^s y T_k^f.
3) Un coste fijo C_k.

Se puede suponer que se conoce el coste c_{ij} del recorrido mínimo entre los clientes x_i y x_j así como el tiempo t_{ij} empleado en dicho recorrido. Los costes c_{ij} se denominan costes variables.

Existen diversos objetivos que pueden intentar alcanzarse, algunos de los más habituales son:

1) Maximizar la suma de las prioridades de los clientes atendidos.
2) Minimizar los costes fijos de transporte de los vehículos utilizados.
3) Minimizar el coste total de los costes variables de los recorridos.

La mayoría de los objetivos que se persiguen en la práctica se corresponden con alguno de los anteriores o son alguna combinación lineal de algunos o todos ellos. En este último caso se asignan arbitrariamente unos coeficientes de peso en función del nivel de preferencia que quiera asignarse a cada uno de los objetivos anteriores.

Existe una relación muy estrecha entre este problema básico y el problema del viajante de comercio (travelling salesman problem) (TSP); no resulta, por tanto, sorprendente que la formulación que a menudo se elige para (VR) sea muy similar a la de (TSP). Esta formulación presenta, sin embargo, dos inconvenientes importantes: por un lado la grandísima cantidad de variables que aparecen y, por otro lado, la imposibilidad práctica de añadir restricciones adicionales al problema básico debido a la gran complejidad que ello entrañaría. No resulta sorprendente que en esta situación se hayan intentado buscar formulaciones alternativas [BaCr81]. A continuación veremos que, bajo ciertas restricciones, es posible plantear el problema en términos de (SP).

Supongamos que para cada vehículo v_k se conoce la familia S_k de todas las posibles rutas que puede realizar. Por consiguiente, si S_{jk} es el conjunto de clientes que el vehículo k visita en un cierto recorrido j, suponemos también conocido el coste variable total asociado al recorrido óptimo entre todos esos clientes. Hay que resaltar que la tarea anterior no resulta inmediata cuando la cardinalidad $|S_{jk}|$ es suficientemente grande, ya que el cálculo del recorrido óptimo supone resolver un (TSP) en S_{jk}.

Sea la matriz $G=[g_{ij(k)}]$ que tiene una fila asociada a cada cliente x_i y n bloques de columnas. El k-ésimo bloque de columnas está asociado al vehículo v_k y la j(k)-ésima columna de ese bloque se corresponde con el j-ésimo recorrido de dicho vehículo. Los elementos $g_{ij(k)}$ de la matriz valen 1 ó 0 según el cliente x_i sea o no visitado en el recorrido S_{kj}. Sea también $C(S_{jk})$ el coste asociado a la ruta S_{jk} del vehículo k.

El problema (VR) puede ahora considerarse como el problema de elegir la familia de recorridos que minimice la suma total de costes cumpliendo los siguientes requisitos:

a) contenga como máximo un recorrido de cada vehículo
b) cada cliente sea visitado exactamente una vez por alguno de los vehículos.

En concreto, se trata de elegir como máximo una columna de cada bloque de G de forma que cada fila de la matriz tenga exactamente un coeficiente 1 en el conjunto de columnas elegidas y que se minimice la suma total de costes.

Definiendo variables de holgura asociadas a aquellas restricciones que imponen que como máximo se elija una columna de cada uno de los bloques y asignando a estas nuevas variables unos coeficientes de coste suficientemente grandes, el problema (VR) se convierte en un caso particular de un problema (SP).

Problema de asignación de tripulaciones

Las compañías aéreas (y todo tipo de compañías de transporte público) se enfrentan continuamente con el problema de determinar los turnos de trabajo de las distintas tripulaciones. El problema consiste en asignar a cada uno de los miembros de las tripulaciones los vuelos y actividades de mantenimiento y formación que debe realizar en los siguientes meses. La asignación debe realizarse teniendo en cuenta los convenios sindicales de manera que la carga de trabajo sea la misma para todas las personas de una misma categoría e intentando conseguir un rendimiento satisfactorio.

Este problema resulta muy complejo ya que el número de personas que trabajan en estas compañías suele ser muy elevado y, además, las restricciones de carácter laboral son bastante complicadas. Por este motivo, normalmente el problema se descompone en dos fases; en la primera se generan los distintos turnos y en la la segunda se asignan tripulantes a cada uno de los turnos establecidos.

El problema de la generación de los distintos turnos (crew scheduling problem) consiste en la obtención de todas las distintas posibles secuencias de vuelos que comiencen y terminen en alguno de los lugares base de las tripulaciones y que tengan una duración que se ajuste a la normativa establecida. Cada uno de estos turnos está compuesto por una serie de vuelos que se llaman 'periodos de guardia', cada uno de los cuales suele corresponderse con una jornada de trabajo, separados entre sí por 'periodos de reposo'. Por tanto la secuencia {vuelo, descanso, vuelo} puede formar un turno. Las limitaciones de las regulaciones laborales exigen que los periodos de descanso tengan una duración mínima así como ciertas restricciones en las duraciones de los periodos de vuelo.

Teniendo en cuenta que casi todos los vuelos se repiten por lo menos semanalmente y algunos de ellos incluso diariamente, suele aprovecharse esta periodicidad formulando el problema en unidades diarias para cada semana, es decir encontrando el conjunto de turnos que cubre todos los vuelos de cada día de la semana con un coste total mínimo.

Por consiguiente, una vez obtenidos los distintos turnos el problema de asignación de las tripulaciones puede formularse de la siguiente manera:

Sea $M=\{v_1,...,v_m\}$ el conjunto de los distintos vuelos que ofrece la compañía en un periodo semanal y c_j el coste del turno j. Se define una matriz $A=[a_{ij}]$ que tiene una fila asociada a cada uno de los vuelos que tienen lugar en una semana y una columna correspondiente a cada uno de los n posibles turnos, tal que sus elementos a_{ij} valen 1 o 0 según el vuelo v_i esté o no cubierto por el turno j. Se desea conocer la familia de turnos que cubre todos los servicios semanales y cuyo coste sea mínimo, es decir se desea resolver el problema

$$\begin{aligned} \min \quad & \sum_{j=1}^{n} c_j x_j \\ \text{t.q.} \quad & \sum_{j=1}^{n} a_{ij} x_j = 1 \,, \quad i=1,...,m \\ & x_j \in \{0,1\} \,, \quad j=1,...,n \end{aligned}$$

Esta formulación responde a un problema de set partitioning. Ahora bien, en ocasiones es necesario permitir que, para cumplir las limitaciones de duración máxima de turnos, una tripulación regrese a su base de trabajo como pasajera en un vuelo asignado a otra tripulación. En este caso, las restricciones pueden relajarse ligeramente obteniendo la siguiente formulación como problema de set covering:

$$\begin{aligned} \min \quad & \sum_{j=1}^{n} c_j x_j \\ \text{t.q.} \quad & \sum_{j=1}^{n} a_{ij} x_j \geq 1 \,, \quad i=1,...,m \\ & x_j \in \{0,1\} \,, \quad j=1,...,n \end{aligned}$$

I.2 Conversión de un problema (SP) en (SC) y en (SPk)

La formulación de estos tres tipos de problemas refleja la semejanza que existe entre los mismos. Una observación más detenida permite, sin embargo, apreciar que la relación entre (SC) y (SP) es más lejana que la que existe entre (SPk) y (SP). Ello resulta bastante intuitivo a partir del hecho de que las restricciones de (SPk) y (SP) son de "$\leq$" (cada restricción exige que como máximo una, o exactamente una variable, respectivamente, valga 1), mientras que las de (SC) son de "$\geq$" (como mínimo una de las variables debe valer 1).

Las relaciones existentes entre estos tres tipos de problemas se reflejan en la estructura de sus soluciones posibles, ya que tanto en el caso de los problemas (SP) como en el de los de (SPk), las componentes fijadas a 1 en las soluciones se corresponden con columnas de la matriz que son ortogonales dos a dos, mientras que esto no ocurre en el caso de los problemas de set covering. No obstante, los problemas de set partitioning pueden considerarse como una clase muy particular de problemas de set covering en los que todas las restricciones se satisfacen como estricta igualdad. Una interpretación análoga puede establecerse entre los problemas de set partitioning y los de set packing.

Una primera propiedad referente a estos problemas es que en el caso de (SP) y de (SPk) todas las soluciones posibles se corresponden con puntos extremos del conjunto de soluciones posibles de sus respectivas relajaciones lineales. Ello se debe al hecho de que las variables que están fijadas a 1 en las soluciones posibles, tanto para (SP) como para (SPk), forman un conjunto de vectores linealmente independientes, ya que la suma de todas las columnas correspondientes de la matriz A será igual (menor o igual en el caso de (SPk)) a 1. En el caso de (SC), dada una solución posible, ésta no tiene por qué ser vértice del conjunto de soluciones posibles de la relajación lineal, pero, a partir de ella, siempre se puede construir otra que sí sea un vértice y para la que la función objetivo tome un valor menor o igual.

Dado un problema (SP) siempre es posible transformarlo en uno de la forma (SC).

Sean $\theta > \sum_{j\in N} c_j$ y el vector $c' = \theta eA + c$, podemos reescribir (SP) como

$$\min \{ cx+\theta ey \ / \ Ax-y = e, \ y\geq 0, \ x_j\in \{0,1\}, j\in N\}.$$

Teniendo en cuenta que las restricciones implican que $y=Ax-e$, este nuevo problema puede escribirse como

$$\min \{ -\theta m+c'x \ / \ Ax\geq e, \ x_j\in \{0,1\}, \ j\in N\}$$

que es un problema de tipo set covering.

Es importante resaltar que el problema obtenido es equivalente a (SP) en el sentido de que tiene el mismo conjunto de soluciones óptimas. Ahora bien, ambos problemas no son en general equivalentes ya que no tienen el mismo conjunto de soluciones posibles.

De forma análoga es posible transformar (SP) en (SPk). Para ello se elige una cantidad θ que, como antes, sea mayor a la suma de los costes de todas las variables de (SP) y se define el vector $c'' = \theta eA - c$, con lo que (SP) puede reescribirse como

$$\min \{cx + \theta ey \;/\; Ax + y = e,\ y \geq 0,\ x_j \in \{0,1\},\ j \in N\}.$$

Teniendo en cuenta que las restricciones del problema imponen que $y = e - Ax$ podremos reescribir de nuevo

$$\min\{\theta m - c''x \;/\; Ax \leq e,\ x_j \in \{0,1\},\ j \in N\}.$$

Evidentemente este problema es equivalente a

$$\max\{c''x \;/\; Ax \leq e,\ x_j \in \{0,1\},\ j \in N\},$$

que es un problema de tipo (SPk).

Al igual que en el caso anterior, la equivalencia entre el problema original (SP) y el problema de set packing obtenido se limita al conjunto de soluciones óptimas.

I.3 Algunos resultados teóricos

En esta sección se recogen algunos de los principales resultados teóricos relacionados con problemas de Set Partitioning. Todos estos resultados se deben a trabajos previos realizados por otros autores (casi todos se encuentran en [Chv72, Edm65, NeTr74, Bal74, Bal75, Bal76, BaZe76, Bal77, Pad73, Pad74, Pad75, Pad79]). Hay que resaltar que tanto E. Balas como M. Padberg han proporcionado la mayoría de ellos, habiendo realizado además un excelente trabajo de síntesis [BaPa79] en el que recopilan no sólo sus propios resultados sino aquellos debidos a otros autores.

El estudio a nivel teórico y desde distintas perspectivas de los problemas de Set Partitioning, y de otros problemas relacionados con los mismos, ha proporcionado un cierto conocimiento de algunas propiedades importantes, llegando a caracterizarlas en algunos casos. Hay que resaltar que este conocimiento es enriquecedor no sólo porque ayuda a comprender mejor la estructura subyacente de los problemas que se estudian, sino porque además, en ocasiones, estas propiedades pueden utilizarse en algún esquema algorítmico diseñado para resolver de forma eficiente los mismos.

El estudio del polítopo de los problemas de Set Partitioning supera en dificultad al de otros polítopos asociados a programas enteros cero-uno. Ello se debe a que ni siquiera puede abordarse la difícil tarea de intentar caracterizar las facetas de dicho polítopo, ya que para ello sería necesario conocer la dimensión del mismo, problema que en este caso resulta ser NP completo. El desconocimiento de la dimensión de este polítopo, situa el problema en un contexto en el que lo más que se puede lograr es reconocer y caracterizar distintas caras del polítopo, pero nunca saber si estas caras son de dimensión máxima, es decir si son facetas. Este aspecto resulta decepcionante a la hora de atacar los problemas de Set Partitioning desde la teoría de combinatoria poliédrica, pero ello no excluye la posibilidad de intentar una aproximación a los mismos desde esta perspectiva. En este sentido, veremos que se conocen facetas de ciertos polítopos que resultan ser los asociados a distintas relajaciones de los problemas de Set Partitioning que pueden utilizarse como desigualdades válidas para los problemas originales.

El alto esfuerzo computacional necesario para la obtención de estas facetas justifica el interés por otros tipos de desigualdades válidas para los problemas de Set Partitioning. En este sentido, veremos que el análisis de las condiciones lógicas implicadas por el conjunto de restricciones de estos programas enteros cero-uno, da lugar a una formulación de los mismos como programas disyuntivos, es decir, como programas enteros sujetos a ciertas restricciones en términos de familias de disyunciones. Estas disyunciones serán, en algunos casos, computacionalmente sencillas de generar a partir de tablas fraccionales del Simplex.

Por otro lado, las relajaciones lineales de los problemas de Set Partitioning poseen en general un alto grado de degeneración, debido a que normalmente existe un número bastante grande de bases asociadas a la misma solución posible básica. Ello hace que a menudo no resulte sencillo obtener las tablas del Simplex asociadas a la solución óptima de la relajación lineal. Por este motivo, se exponen también otras aproximaciones que permiten evitar este obstáculo.

La primera de ellas consiste en la obtención de desigualdades válidas para los problemas de Set Partitioning, que no estén relacionadas con las tablas del Simplex, a partir de las implicaciones lógicas del conjunto original de restricciones. Otra aproximación se basa en el conocimiento de ciertas propiedades que deben cumplir los vértices correspondientes a soluciones posibles básicas enteras en los polítopos de Set Partitioning y de Set Packing asociados a sus respectivas relajaciones lineales. La caracterización de estas propiedades permite obtener soluciones posibles enteras que son vértices de los polítopos antes mencionados, sin necesidad de utilizar el algoritmo del Simplex.

A continuación se expone la notación que será utilizada en el resto del capítulo.

Notación:

(SP) $\min \{ cx \,/\, Ax=e,\ x_j \in \{0,1\},\ j \in N \}$	• Problema de Set Partitioning.
($\overline{SP}$) $\min \{ cx \,/\, Ax=e,\ x \geq 0 \}$	• Relajación lineal del Problema de Set Partitioning.
$\overline{P} = \{ x \in R^n \,/\, Ax=e,\ x \geq 0 \}$	• Conjunto de soluciones posibles de la relajación lineal asociada a un problema de Set Partitioning.
$P = \text{conv} \{ x \in R^n \,/\, Ax=e,\ x_j \in \{0,1\},\ j \in N \} =$ $= \text{conv} \{ x \in \overline{P} \,/\, x \text{ entero} \}$	• Polítopo de Set Partitioning.
(SPk) $\min \{ cx \,/\, Ax \leq e,\ x_j \in \{0,1\},\ j \in N \}$	• Problema de Set Packing.
($\overline{SPk}$) $\min \{ cx \,/\, Ax \leq e,\ x \geq 0 \}$	• Relajación lineal del Problema de Set Packing.
$\overline{Pk} = \{ x \in R^n \,/\, Ax \leq e,\ x \geq 0 \}$	• Conjunto de soluciones posibles de la relajación lineal asociada a un problema de problema de Set Packing.
$Pk = \text{conv} \{ x \in R^n \,/\, Ax \leq e,\ x_j \in \{0,1\},\ j \in N \}$ $= \text{conv} \{ x \in \overline{Pk} \,/\, x \text{ entero} \}$	• Polítopo de Set Packing.
(SC) $\min \{ cx \,/\, Ax \geq e,\ x_j \in \{0,1\},\ j \in N \}$	• Problema de Set Covering.

($\overline{SC}$) $\min \{ cx / Ax \geq e, x \geq 0 \}$ • Relajación lineal del Problema de Set Covering.

$\overline{P}c = \{ x \in R^n / Ax \geq e, x \geq 0 \}$ • Conjunto de soluciones posibles de la relajación lineal asociada a un problema de Set Covering.

$Pc = \text{conv} \{ x \in R^n / Ax \geq e, x_j \in \{0,1\}, j \in N \} =$ • Polítopo de Set Covering.
$= \text{conv} \{ x \in \overline{P}c / x \text{ entero} \}$

I.3.1 Problemas de grafos asociados a problemas de Set Partitioning

A continuación se definen cuatro tipos de problemas definidos sobre grafos, que están todos ellos relacionados entre sí. Dos de ellos son casos particulares de problemas de Set Packing, mientras que los otros dos lo son de problemas de Set Covering.

Sea G=(N,E) un grafo finito, no dirigido, con n = |N| nodos y q = |E| arcos. Sea A_G la matriz de incidencia nodos-arcos de G de dimensión nxq. Un *edge matching* (acoplamiento por arcos) de G es un subconjunto E' de arcos de E tal que cada nodo de G es incidente con un arco de E' como máximo. Si todos los nodos de G son incidentes con exactamente un arco de E', se dice que E' es un matching perfecto. **El problema de Edge Matching** consiste en encontrar un edge matching de cardinalidad máxima en G y puede por tanto formularse como:

(EM) $\alpha_1 = \max \{ e_q \cdot y / A_G y \leq e_n, y_i = 0,1, i=1,\dots,q \}$

Un *edge covering* (recubrimiento por arcos) en G es un recubrimiento de los nodos mediante arcos y se define como un subconjunto E" de arcos tal que cada nodo de G es incidente como mínimo con un arco de E". **El problema de Edge Covering** consiste en encontrar un edge covering de cardinalidad mínima en G y puede formularse como:

(EC) $\beta_1 = \min \{ e_q \cdot y / A_G y \geq e_n, y_i = 0,1, i=1,\dots,q \}$

Un *node packing* (empaquetado de nodos) o *vertex packing* (empaquetado de vértices) en G es un subconjunto N' de nodos tal que cada arco de G es incidente como máximo con un nodo de N'. Un node packing recibe a menudo otras denominaciones como son *internally stable node set* (conjunto de nodos internamente estable) o *independent node set* (conjunto independiente de nodos). **El problema de Node Packing** consiste en encontrar un node packing de cardinalidad máxima en G, y puede formularse como:

(NP) $\alpha_0 = \max \{ e_n.x \,/\, A_G{}^T x \leq e_q,\ x_j=0,1,\ j=1,...,n \}$

donde $A_G{}^T$ es la matriz transpuesta de A_G.

El problema de Node Covering, o problema de encontrar un node cover (recubrimiento por nodos) de las aristas de G de cardinalidad mínima puede formularse como:

(NC) $\beta_0 = \min \{ e_n.x \,/\, A_G{}^T x \geq e_q,\ x_j=0,1,\ j=1,...,n \}$

Es evidente que (EM) y (NP) son casos particulares de (SPk); en el caso concreto de un matching perfecto se trata de un caso particular de (SP). Análogamente, tanto (EC) como (NC) son casos particulares de (SC). La relación que existe entre los cuatro problemas anteriores, en el caso en el que el grafo G sea conexo, fue demostrada por Gallai en 1958 [Gal58] y viene dada por $\alpha_0 + \beta_0 = \alpha_1 + \beta_1$.

Además, un edge matching máximo y un edge cover mínimo en un grafo G son fácilmente obtenibles uno a partir del otro ya que son complementarios en el grafo G. Lo mismo ocurre para un node packing máximo y un node covering mínimo. En el caso particular que el grafo G sea bipartito (es decir, exista una partición de los nodos de G en dos subconjuntos tal que cada arco del grafo sea incidente exactamente con un nodo de cada uno de los subconjuntos) la matriz A_G es totalmente unimodular. Esto significa que el determinante de cada submatriz cuadrada contenida en A_G vale ± 1 y que, en concreto, la aplicación del algoritmo del Simplex a la relajación lineal ordinaria proporcionará el óptimo del problema sin relajar. Por lo tanto, cuando el grafo G sea bipartito pueden sustituirse estos cuatro problemas por sus respectivas relajaciones lineales.

De los cuatros problemas anteriores, el par (EM), (EC) resulta el más sencillo de resolver ya que existe una caracterización completa del polítopo asociado a (EM). Incialmente fue Edmonds [Edm65] quien definió un sistema lineal de inecuaciones que caracterizan dicho polítopo y propuso un algoritmo de complejidad $O(n^3)$ para resolver el problema. Posteriormente Pulleyblank y Edmonds [PuEd73] han mejorado dicha caracterización identificando el único sistema lineal de desigualdades minimal que define el polítopo del problema (EM).

El par de problemas (NP), (NC) resulta, sin embargo, sensiblemente más difícil de resolver y no existe hasta el momento una caracterización completa del polítopo asociado a ninguno de los dos. Se han identificado, no obstante, algunas clases de facetas de (NP) [Pad73, Tro74, NeTr74, BaZe76]. Estas clases de facetas se definen a partir de algunos problemas de tipo (NP) relacionados con un problema clásico (SPk).

Sea a_j la j-ésima columna de la matriz A de un problema (SPk). El grafo de intersección G_A=(N,E) de A es aquél que tiene un nodo asociado a cada columna de A y un arco para cada par de columnas no ortogonales de A; es decir, $(i,j)\in E$ si y sólo si $a_i.a_j\geq 1$. Sea A_G la matriz de incidencia nodos-arcos de G_A y sea (NP) el problema de Node Packing con pesos, siendo los costes c_j asociados a cada nodo los mismos que los de (SPk), es decir:

(NP) $\max \{ cx \,/\, A_G{}^T x\leq e_q,\ x_j=0,1, j=1,...,n \}$.

Una primera observación es que el conjunto de soluciones posibles (y en particular óptimas) es el mismo para (SPk) y para (NP). Por lo tanto resulta equivalente resolver cualquiera de los dos programas enteros. No obstante, es sencillo construir ejemplos en los que se aprecia que no ocurre lo mismo con sus respectivas relajaciones lineales, teniéndose en ese caso que

$$\max \{ cx \,/\, Ax\leq e_m,\ x\geq 0 \} \leq \max \{ cx \,/\, A_G{}^T x\leq e_q,\ x\geq 0 \}.$$

Sólo en el caso en el que el grafo G sea bipartito, se dará la igualdad en la expresión anterior.

Ejemplo: N = { 1,2,3,4,5}

$$\text{Sea } A = \begin{bmatrix} 1&1&0&1&0 \\ 1&0&1&0&0 \\ 0&1&1&0&1 \\ 1&0&0&1&0 \end{bmatrix}$$

La matriz A_G de incidencia nodos arcos asociada al grafo de intersección G_A viene dada por

$$A_G = \begin{bmatrix} 1&1&1&0&0&0&0 \\ 1&0&0&1&1&1&0 \\ 0&1&0&1&0&0&1 \\ 0&0&1&0&1&0&0 \\ 0&0&0&0&0&1&1 \end{bmatrix}$$

Resulta evidente que el vector $x=(0,{}^1/_2,{}^1/_2,0,{}^1/_2)$ satisface $A_G{}^T x \leq e_q$ pero, en cambio, viola la restricción asociada a la tercera fila de A.

Un *clique* en G es un subgrafo completo maximal. Un *clique covering* de los arcos de G es un conjunto formado por K cliques tal que cada arco de G pertenece al conjunto de arcos de alguno de los cliques de K.

Sea G el grafo de intersección asociado a un problema (SPk). Dado un clique covering de los arcos de G, se define la matriz A_K como la matriz de incidencia de los cliques de K respecto a los nodos de G. De forma análoga al caso de (NP), el conjunto de soluciones posibles (y óptimas) de (SPk) es el mismo que el del problema de set packing obtenido al sustituir la matriz A por A_K. Por lo tanto, un único problema (NP) es equivalente a muchos problemas (SPk) aparentemente diferentes. El conjunto C de todos los cliques de G, es evidentemente un clique covering de G. En consecuencia, (NP) en el que todos los costes son 1 es equivalente al problema de calcular la cardinalidad del mayor conjunto independiente de nodos en G, que viene dado por

$$(SPk_X) \quad \alpha_0(G) = \max \{ e_n.x \; / \; A_C.x \leq e_C, \; x_j=0,1, \; j=1,...,n \}$$

donde A_C es la matriz de incidencia de todos los cliques de G respecto a los nodos de G. Sea el problema (K_C) de encontrar un clique covering de los nodos de G de cardinalidad mínima,

$$(K_C) \quad \omega_0(G) = \min \{ e_C.y \; / \; A_C{}^T y \geq e_n, \; y_i=0,1, \; i=1,...,|C| \}.$$

Teniendo en cuenta que los programas lineales asociados a (SPk_C) y (K_C) son duales uno de otro se tiene que $\alpha_0(G) \leq \omega_0(G)$. Además $\alpha_0(G)=\omega_0(G)$ si y sólo si ambos problemas lineales tienen soluciones óptimas enteras. Un grafo G tal que $\alpha_0(G')=\omega_0(G')$ para todos los subgrafos inducidos G' de G se dice que es perfecto.

I.3.2 Facetas del polítopo de Set Packing y de los grafos asociados

Consideremos un problema (SPk) cuya matriz de restricciones A no posee ninguna fila ni columna compuesta únicamente por ceros. Así el conjunto de soluciones posibles del problema lineal asociado estará acotado. Consideremos también los polítopos Pk y $\overline{Pk}$. Es fácil observar que dim(Pk)=dim($\overline{Pk}$)=n. Por lo tanto las facetas o caras n-1 dimensionales de Pk serán por definición conjuntos $Pk \cap \{ x \in R^n \; / \; \pi x = \pi_0 \}$ tales que $\pi x \leq \pi_0 \; \forall x \in Pk$ y $\pi x = \pi_0$

exactamente para n puntos afinmente independientes de Pk. En adelante se llamará faceta a la desigualdad $\pi x \leq \pi_0$ que define la misma. Existe una clase de facetas triviales para Pk, que son las de la forma $x_j \geq 0$, $j \in N$. La no negatividad de los elementos de la matriz A implica,además, que cualquier otra faceta de Pk diferente de las anteriores debe satisfacer $\pi_j \geq 0$ $\forall j \in N$ y $\pi_0 > 0$. A continuación se expone bajo qué condiciones alguna de las desigualdades de $Ax \leq e_n$ es una faceta.

Teorema 1 [Pad74]: La desigualdad

$$\sum_{j \in K} x_j \leq 1, \quad K \subseteq N$$

es una faceta de Pk si y sólo si K es el conjunto de nodos de un clique en G_A.

Como consecuencia del teorema anterior se tiene que $Pk \subseteq P_C \subseteq \overline{Pk}$.

En general se trata de contenidos estrictos, aunque existe una amplia clase de matrices, llamadas matrices perfectas, para las que estos tres polítopos coinciden. Las matrices perfectas están muy relacionadas con los grafos perfectos y contienen como caso particular las matrices no negativas unimodulares.

Siempre que Pk≠PC existirá alguna faceta de Pk de las caracterizadas por el teorema anterior que no formará parte del conjunto de desigualdades que definen P. Sin embargo, para caracterizar P, se necesitan en general otras clases de desigualdades más complicadas. Como todas estas desigualdades estarán relacionadas con el grafo de intersección G_A asociado a la matriz A, en adelante se denotará por Pk(G) al único polítopo de node packing / set packing definido por G_A. Los siguientes teoremas se refieren a ciertos grafos cuyos polítopos asociados de set packing tienen, como en el caso del último teorema, alguna faceta cuyos coeficientes en la parte izquierda sean todos iguales a 1. Previamente se necesitan ciertas definiciones.

Definición 1:

Un *chordless cycle* (ciclo sin cuerda) C en un grafo G es un ciclo tal que cada uno de sus nodos es adyacente con exactamente otros 2 nodos de C.

Un *hole* (agujero) es un chordless cycle de longitud mayor o igual que 3.

Cualquier ciclo de longitud 3 es obviamente chordless y además se trata de un clique.

Un *anti-hole* (anti agujero) es el complementario de un hole.

Un *web* (telaraña) es un grafo $W_{(n,k)} = (N,E)$ tal que $|N|=n \geq 3$, $1 \leq k \leq [n/2]$ y

$$(i,j) \in E \Leftrightarrow j=i+h, \quad h=k,\dots,n-k$$

El web $W_{(n,k)}$ es regular de grado n-2k+1 y tiene exactamente n node packings máximos de tamaño k.

Un *anti-web* (anti-telaraña) $\overline{W}_{(n,k)}$ es el complementario de un web $W_{(n,k)}$.

El anti-web $\overline{W}_{(n,k)}$ es regular de grado 2(k-1) y contiene exactamente n cliques máximos de tamaño k.

Teorema 2 [Pad74, NeTr74]:

i) Si G es un hole, entonces

$$\sum_{j \in N} x_j \leq k$$

es una faceta de Pk(G) con k=(n-1)/2

ii) Si G es un anti-hole, entonces

$$\sum_{j \in N} x_j \leq k$$

es una faceta de Pk(G) con k=2

Teorema 3 [Tro74]:

i) Si $G=W_{(n,k)}$ es un web, entonces la desigualdad

$$\sum_{j \in N} x_j \leq k$$

es una faceta de Pk(G) si y sólo si n y k son primos entre sí.

ii) Si G= $\overline{W}_{(n,k)}$ es un anti-web con n y k primos entre sí y k≥2, entonces la desigualdad

$$\sum_{j\in N} x_j \leq [k/n]$$

es una faceta de Pk(G).

Una observación importante es que en las facetas obtenidas mediante los tres teoremas anteriores, el término independiente coincide con el número de independencia, es decir con la cardinalidad máxima de un conjunto independiente de nodos, del grafo G asociado con Pk(G): 1 en el caso de un grafo completo o un clique, (n-1)/2 y 2 en el caso de un hole y un anti-hole respectivamente, y k y [k/n] en el caso de un web y un anti-web respectivamente.

La cuestión que se plantea ahora es la de saber si existen condiciones necesarias o suficientes para afirmar que una desigualdad en la que todos los coeficientes sean igual a 1 y el término independiente coincida con el número de independencia $\alpha(G)$ de un grafo G es una faceta de Pk(G). Los dos siguientes teoremas proporcionan condiciones suficientes que no son necesarias. Posteriormente se verá una condidion necesaria que no es suficiente y finalmente una condición necesaria y suficiente.

Definición 2:

Una arista $u\in E$ de G=(N,E) se dice que es *α-crítica* si su eliminación produce un grafo con un número de independencia mayor que $\alpha(G)$.

Un *cutset* (conjunto de corte) $C=(N_1,N_2) \subseteq E$ es un conjunto (posiblemente vacío) de arcos de G que une los nodos de N_1 con los de $N_2=N\backslash N_1$.

Un cutset se dice que es *α-crítico* si $\alpha(G_1) + \alpha(G_2) \geq \alpha(G) + 1$, siendo G_1 y G_2 los subgrafos de G inducidos por N_1 y N_2 respectivamente.

Teorema 4 [Chv72]: Sea E^* el conjunto de aristas α-críticas de G. Si $G^*=(N,E^*)$ es conexo, entonces

$$\sum_{j\in N} x_j \leq \alpha(G) \qquad (1) \qquad \text{es una faceta de Pk(G)}$$

Es fácil construir ejemplos en los que se observa que la condición necesaria dada por el teorema anterior no es suficiente. El concepto de arco α-crítico proporciona, sin embargo una condición que sí es necesaria.

Teorema 5 [BaZe76]: Sea $G=(N,E)$ un grafo tal que ó bien G^* es conexo, ó bien existe un cutset $C=(N_1,N_2)$ en G tal que se cumplen la siguientes condiciones para $i=1,2$:

a) La desigualdad $\sum_{j\in N} x_j \leq \alpha(G_i)$ es una faceta de $Pk(G_i)$

b) Cualquier conjunto independiente de nodos de cardinalidad máxima de G_i está contenido en algún conjunto independiente de nodos de cardinalidad máxima de G.

Entonces, la desigualdad (1) es una faceta de Pk(G)

Teorema 6 [BaZe76]: Si la desigualdad

$$\sum_{j\in N} x_j \leq \alpha(G) \qquad (1)$$

es una faceta de Pk(G) , entonces cualquier cutset de G es α-crítico.

Como consecuencia del Teorema 6, cuando (1) es una faceta de Pk(G) se tienen las siguietes propiedades:

1) El grafo G es conexo.

2) Cualquier vértice de G tiene un grado mayor o igual que 2.

3) Para cualquier clique K de G, esiste un conjunto independiente de nodos de G de cardinalidad máxima que no contiene ninguno de los nodos de K.

Teorema 7 [BaZe76] : Sea $G_1=(N_1,E_1)$ el subgrafo de $G=(N,E)$ inducido por algún subconjunto propio N_1 de N, y sea

$$\sum_{j\in N_1} x_j \leq \alpha(G_1) \qquad (2)$$

una faceta de $Pk(G_1)$.

Para cada $k\in N_2=N\backslash N_1$ sea G^k el subgrafo inducido por $N_1 \cup \{k\}$. Entonces, (2) es una faceta de Pk(G) si y sólo si para todo $k \in N_2$, el cutset (k,N_1) de G^k no es α-crítico.

Las facetas del polítopo Pk(G) definidas hasta ahora forman una clase amplia pero se conoce tambien otra familia que es aquella formada por facetas obtenidas a partir de facetas de algún polítopo de dimensión menor asociado a algún subgrafo G' contenido en G. A diferencia de la clase de facetas conocida hasta ahora, en esta nueva familia los coeficientes no se restringen al valor 1.

Teorema 8: Sea G' el subgrafo de G inducido por un conjunto $S \subseteq N$. Si

$$\sum_{j \in S} \alpha_j x_j \leq s \qquad (3)$$

es una faceta de Pk(G'), entonces existen enteros β_j, $0 \leq \beta_j \leq s$, tales que

$$\sum_{j \in N} \alpha_j x_j + \sum_{j \in N \setminus S} \beta_j x_j \leq s \qquad (4)$$

es una faceta de Pk(G).

Fue Pollatschek (1970) quien estableció este tipo de facetas. Independientemente Padberg [Pad73] demostró el mismo teorema para el caso en el que G' es un hole impar y propuso un procedimiento secuencial para calcular los coeficientes β_j. Nemhauser y Trotter [NeTr74] demostraron en 1974 que tanto el resultado como el procedimiento son extensibles para el caso de un grafo G' cualquiera.

El procedimiento de *lifting* (desproyección) mediante el cual se puede obtener una faceta para Pk(G) a partir de una faceta del polítopo Pk(G') asociado a cualquier subgrafo inducido G', consiste en calcular los coeficientes β_j de (4). Para ello es necesario resolver una sucesión de problemas de set packing, uno para cada una de las variables $j \in N \setminus S$, siendo $S \cup S'$ el conjunto de índices de variables que interviene en cada uno de los subproblemas, donde S'es el conjunto de índices de variables para las que ya se ha resuelto previamente el problema de set packing correspondiente.

El teorema anterior no define, sin embargo, todas las facetas de Pk(G) obtenibles mediante procedimientos de lifting de alguna faceta de Pk(G'). En concreto, existen diversas facetas del tipo (4), en las que los coeficientes β_j no son enteros, que no pueden calcularse mediante un procedimiento secuencial pero que pueden obtenerse a partir de una generalización de este tipo de procedimientos llamado *procedimiento simultáneo*.

En un contexto más general, Balas y Zemel demostraron que cualquier faceta de Pk(G) puede obtenerse a partir de una faceta de Pk(G'), donde G' es algún subgrafo inducido de G, mediante un procedimiento que combina lifting simultáneo con proyecciones en cierto subespacio.

I.3.3 Subgrafos generadores de facetas

Definición 3: Un grafo (G,N) se dice que es *generador de facetas* si existe una faceta $\pi x \leq \pi_0$ de Pk(G) que no puede obtenerse mediante un procedimiento de lifting (secuencial o simultáneo) de una faceta de Pk(G'), para ningún subgrafo G' de G inducido por un conjunto N de nodos de cardinalidad n-1.

G' se dice que es *fuertemente generador de facetas* si la propiedad anterior se impone para cualquier subgrafo G' inducido por un conjunto de nodos N' de cardinalidad menor que n.

En el apartado anterior se ha visto que existe un procedimiento para caracterizar todas las facetas de Pk(G). Ahora bien, veremos a continuación que es posible construir subgrafos arbitrariamente complejos que tienen la propiedad de ser generadores de facetas; por lo tanto, resultará excesivamente complicado utilizar un procedimiento que consista en enumerar todos los posibles subgrafos generadores de facetas de un grafo G.

De los grafos estudiados hasta el momento, los cliques no son generadores de facetas, mientras que los holes y anti-holes impares son fuertemente generadores de facetas. Un web $W_{(n,k)}$ es generador de facetas (también fuertemente) si y sólo si $k \geq 2$ y, además, k y n son primos entre sí; mientras que, cuando n y k son primos entre si, su complementario, el anti-web, es fuertemente generador de facetas si y sólo si $k \lfloor n/k \rfloor = n-1$.

Una pregunta que se plantea en este contexto cuya respuesta podría facilitar en gran medida la tarea de obtener grafos generadores de facetas, es la de saber si el complementario de un grafo generador de facetas también lo es. Desgraciadamente, y al contrario de lo que sugiere el caso de los web y los anti-web la respuesta resulta en general negativa. El siguiente torema formaliza esta cuestión.

Teorema 9 [Pad75]: Sean G=(N,E) un grafo cualquiera, Pk(G) el polítopo asociado de set packing y $P_C = \{x \in R^n / A_C x \leq e_C, x \geq 0\}$, siendo A_C la matriz de clique de G.
Si Pk(G) tiene una faceta $\pi x \leq \pi_0$, tal que $\pi_j > 0$, $j \in N$, y πx alcanza el máximo en P_C en un vértice $\bar{x}$ de P_C tal que $0 < \bar{x}_j < 1$, $\forall j \in N$, entonces Pk($\bar{G}$) tiene una faceta $\pi' x \leq \pi'_0$ tal que $\pi'_j > 0$, $j \in N$.
Además, si la submatriz A_1 de A_C ,formada por las restricciones de $A_C\ \bar{x} \leq e$ que se satistacen como estricta igualdad, es cuadrada e irreducible, entonces $\bar{G}$ es fuertemente generador de facetas.

Nota: Una matriz M es irreducible si no existen matrices de permutación P y R tales que

$$PMR = \begin{bmatrix} N & 0 \\ L & S \end{bmatrix}$$

siendo N y S matrices cuadradas y 0 la matriz formada por ceros.

A continuación se exponen varios procedimientos para obtener grafos generadores de facetas. Los primeros serán los llamados grafos de sustitución [Chv72].

Definición 4: Sean $G=(N,E)$ y $H=(Q,F)$ grafos cuyos conjuntos de nodos son $N=\{v_1,...,v_n\}$ y $Q=\{v'_1,...,v'_q\}$ ($N \cap Q = \emptyset$) respectivamente. El grafo de sustitución G_k^H se obtiene a partir de G sustituyendo H por v_k, y se define como la unión disjunta de $G \setminus \{v_k\}$ y H, junto con los arcos que unen cada nodo de H con aquellos nodos de G adyacentes a v_k.

Teorema 9: [Chv72] Sean

$$\sum_{j \in N} \alpha_{ij} x_j \leq 1 \quad , \ i \in I$$

$$x_j \geq 0 \quad , \qquad j \in N$$

un sistema lineal que define Pk(G) y

$$\sum_{j \in Q} \beta_{kj} x_j \leq 1 \quad , \ k \in K$$

$$y_j \geq 0 \quad , \qquad j \in Q$$

un sistema lineal que define Pk(H). Entonces

$$\sum_{j \in N \setminus \{n\}} \alpha_{ij} x_j + \alpha_{in}^+ \sum_{j \in Q} \beta_{kj} y_j \leq 1 \quad , \ i \in I, \ k \in K$$

$$x_j \geq 0 \quad j \in N \setminus \{n\}, \qquad y_j \geq 0 \ , \qquad j \in Q$$

es un sistema lineal que define $Pk(G_n^H)$ donde $\alpha_{in}^+ = \max \{0, \alpha_{in}\}$, $i \in I$.

El teorema anterior caracteriza el polítopo de set packing $Pk(G_n^H)$ en términos de polítopos Pk(G) y Pk(H) de dimensiones más bajas. Deja sin embargo abierta la cuestión de saber cuando los sistemas lineales que definen los polítopos son minimales. Ello se establece en el siguiente teorema.

Teorema 10 [BaZeme76] : Si las desigualdades

$$\sum_{j\in N} \alpha_j x_j \leq 1 \qquad (5)$$

y

$$\sum_{j\in Q} \beta_j x_j \leq 1 \qquad (6)$$

son facetas de Pk(G) y Pk(H) respectivamente, entonces la desigualdad

$$\sum_{j\in N\setminus\{n\}} \alpha_j x_j + \alpha_n \sum_{j\in Q} \beta_j y_j \leq 1 \qquad (7)$$

es una faceta de $P(G_n{}^H)$.

Recíprocamente, si la desigualdad

$$\sum_{j\in N\setminus\{n\}} \alpha_j x_j + \sum_{j\in Q} \gamma_j y_j \leq 1 \qquad (8)$$

es una faceta de $Pk(G_n{}^H)$, entonces (5) es una faceta de Pk(G) con

$$\alpha_n = \max \left\{ \sum_{j\in Q} \gamma_j y_j \;/\; y \in Pk(H) \right\} \quad ;$$

y si además $\gamma_k > 0$ para algún $k \in Q$, entonces (5) es una faceta de P(H), con $\beta_j = \gamma_j \alpha_n{}^{-1}$, $j \in Q$.

El Teorema 10 implica que si los sistema lineales introducidos en el Teorema 9 que definen Pk(G) y Pk(H) son ambos minimales, entonces también lo será el sistema lineal obtenido para Pk(GnH). Recíprocamente, si el sistema lineal minimal que define $Pk(G_n{}^H)$ es de la forma establecida en el Teorema 9 con

$$\max \left\{ \sum_{j\in Q} \beta_{kj} y_j \;/\; y \in Pk(H) \right\} = 1 \qquad k \in K$$

entonces los correspondientes sistemas lineales que definen Pk(G) y Pk(H) también son minimales, con $\alpha_{in} = \alpha_{in}{}^{+}$, $i \in I$.

A continuación se da una condición necesaria y suficiente para que un grafo de sustitución sea generador de facetas. La caracterización viene dada en términos de unos grafos llamados *k-casi generadores de facetas*. Un grafo G' cumple la propiedad anterior si Pk(G') tiene una faceta $\pi x \leq \pi_0$ que no es faceta para $Pk(G' \setminus \{v_i\})$ para ningún $i \in N' \setminus \{k\}$. Esto quiere decir que el único posible subgrafo inducido por G' generador de facetas es el que tiene N'\{vk} como conjunto de nodos.

Teorema 11 [BaZe76] : G_n^H es un grafo generador de facetas si y sólo si H es generador de facetas y G es n-casi generador de facetas.

Las siguientes definiciones son necesarias para considerar otro procedimento de construcción de grafos generadores de facetas que se expondrá a continuación.

Definición 5: Una *estrella* $K_{1,n}$ es un grafo completo bipartito con n+1 arcos, cuyos arcos son (i,n+1), i=1,...,n.
Dado un grafo G=(N,E) el grafo $G^{\#}$ se define como aquél formado por $G \cup K_{1,n}$ junto con todos los arcos que unen el i-ésimo nodo de G con el i-ésimo de $K_{1,n}$, i=1,...,n.

Teorema 12 [Pad75]: Si Pk(G) tiene una faceta $\pi x \leq \pi_0$ tal que $p_j>0$, $j \in N$, entonces la desigualdad

$$\pi x + \pi y + (\pi e - \pi_0)\, y_n+1 \leq e$$

donde $e \in R^n$, e=(1,...1), $y \in R^n$, y y_i es la variable asociada al i-ésimo nodo de $K_{1,n}$, i=1,...,n+1,es una faceta de $Pk(G^{\#})$ y $G^{\#}$ es fuertemente generador de facetas.
Además, si πx alcanza su máximo en $P_C=\{x \in R^n \,/\, A_C x \leq e,\ x \geq 0\}$, siendo A_C la natriz de cliques de G, en un vértice $\bar{x}$ tal que $0 < \bar{x}_j < 1$, $\forall j \in N$, entonces

$$\bar{x}.x + (e-\bar{x}).y + \bar{x}_{j*}\, y_{n+1} \leq 1, \text{ con } \bar{x}_{j*} = \min \{ \bar{x}_j \,/\, j \in N \}$$

es una faceta de Pk($\bar{G}^{\#}$), siendo $\bar{G}^{\#}$ el complementario de $G^{\#}$.

Es importante resaltar que en el Teorema 12 la desigualdad $\pi x \leq \pi_0$ no necesita estar generada por G. Veremos como a partir del teorema anterior pueden construirse clases de facetas que resultan de interés.

Definición 6: Dada una faceta $\pi x \leq \pi_0$ de Pk(G), un arco $u=(v_i,v_j)$ se dice que es *π-crítico* si existe un conjunto independiente de nodos S contenido en G\{u} tal que

$$\sum_{j \in S} \pi_j > \pi_0 \quad \text{pero} \quad \sum_{j \in S \setminus \{v_h\}} \pi_j = \pi_0 \quad \text{para h=i o j.}$$

Se dice que se *han insertado dos nodos v_{n+1} y v_{n+2}* en el arco (v_i,v_j) del grafo G, cuando se haya sustituido el arco (v_i,v_j) por el camino $(v_i,v_{n+1},v_{n+2},v_j)$.

Teorema 13 [Pad75]: Sea G=(N,E) un grafo con |N|=n ≥3 nodos, y sea $\pi x \geq \pi_0$ una faceta de Pk(G) tal que $\pi_j>0$, $j \in N$. Si G'=(N',E') es un grafo obtenido a partir de G insertando dos nodos v_{n+1}, v_{n+2} en un arco π-crítico de G, entonces

$$\pi x + \pi_{j*} (x_{n+1} + x_{n+2}) \leq \pi_0 + \pi_{j*}, \quad \text{con } \pi_{j*} = \min \{\pi_i, \pi_j\},$$

es una faceta de Pk(G').

Los resultados expuestos en este apartado ponen de manifiesto la variedad de clases de facetas que existen para el polítopo de set packing. Es, por tanto, natural preguntarse si el conocimiento de alguna de estas clases puede resultar de utilidad a la hora de resolver un problema de set packing, dado que la caracterización completa de todas las clases de facetas resulta tan extremadamente amplia. Afortunadamente, como veremos en el siguiente ejemplo, la respuesta es afirmativa, ya que el conocimiento de las clases de facetas que aparecen más a menudo permite resolver el problema o acercarse considerablemente a la solución del mismo.

Ejemplo: Consideremos el problema de node packing de cardinalidad máxima definido en un anti-hole $\overline{H}_n$ de tamaño n.

Obviamente, un vector x tal que $x_j=1/2$, $j\in N$ es una solución posible para las restricciones $x_i+x_j\leq 1$ asociadas a cada arco (i,j) y $x_j\geq 0$, $j\in N$ que aparecen en la relajación lineal ordinaria. Es fácil comprobar que este vector proporciona también una solución óptima para dicha relajación de valor de la función objetivo n/2. Por lo tanto, el valor del óptimo de la relajación ordinaria aumenta con el número n de nodos del problema original.

Consideremos ahora la clase más sencilla de facetas, que es la asociada a los cliques de cardinalidad máxima de $\overline{H}_n$, que pueden incorporarse a la formulación del programa lineal. La matriz añadida de esta forma a la formulación de la relajación lineal es la matriz circulante de orden n que tiene (n-1)/2 unos en cada fila y cada columna. Esto resulta del hecho de que los cliques de caardinalidad máxima en $\overline{H}_n$ son de tamaño (n-1)/2 y que $\overline{H}_n$ es simétrica para cada nodo perteneciente a cada uno de los (n-1)/2 cliques. Definiendo ahora el vector x dado por $x_j=2/(n-1)$, $j\in N$, se trata claramente de una solución posible para el nuevo problema lineal, que además es óptima, dando un valor de $2n/(n-1)=2+2/(n-1)$ para la función objetivo. Por lo tanto a medida que n aumenta, este valor se aproxima a 2 que es el valor de la solución óptima para el problema entero original.

El ejemplo anterior refleja el efecto tan espectacular que puede conseguir la adición de algunas facetas a la formulación de la relajación lineal. Hemos visto como, en el caso de un anti-hole, la formulación de la relajación lineal ordinaria, proporciona cotas inferiores que, siendo malas en todos los casos, empeoran a medida que aumenta el tamaño del conjunto de nodos. Si embargo, la incorporación de algunas facetas asociadas a los cliques de cardinalidad máxima produce, también vía programación lineal, una cota que es muy parecida a valor óptimo del programa entero (nótese que para $n\geq 4$ el valor de la cota puede redondearse al valor

2 de la solución óptima).

En todo el procedimiento utilizado en el ejemplo anterior, hay que resaltar que el número de cliques de cardinalidad máxima de $\overline{H}_n$, que es el número de desigualdades que se añaden al la formulación de la relajación lineal, constituye una parte muy pequeña del número total de cliques de $\overline{H}_n$ (especialmente cuando n es grande).

I.3.4 Facetas de polítopos relajados: Cortes a partir de disyunciones

En este apartado se estudian alguna facetas de ciertos polítopos que también son relajaciones de P. Estas facetas no están relacionadas con el polítopo de set packing, y a diferencia de las que ya se han visto, pueden obtenerse fácilmente a partir de una tabla fraccional de Simplex de forma que violen un determinado vértice del conjunto de soluciones posibles de la relajación lineal del problema original. La ventaja que presentan respecto a otros métodos clásicos, que también utilizan las tablas fracionales de Simplex asociadas a la relajación lineal, resulta del hecho que tienen en cuenta la estructura particular del problema de set partitioning que se intenta resolver.

El enfoque que permitirá generar estas nuevas facetas es el de la programación disyuntiva [Bal4, Bal75, Bal76, Bal879], que consiste en formular los problemas como programas lineales con condiciones lógicas que, a su vez, pueden expresarse como disyunciones entre conjuntos de desigualdades lineales. La programación disyuntiva es aplicable no sólo a programas enteros; pero para este tipo de problemas, así como para otros problemas combinatorios que tienen una estructura específica, una formulación como programa disyuntivo ofrece mayores ventajas. En particular, mediante esta formulación podrán derivarse planos secantes, que son computacionalmente muy sencillos de generar y además los coeficientes de los mismos serán de distintos signos; esta propiedad, a menudo, ayudará a reducir la degeneración dual que tiende a aparecer en los algoritmos de planos secantes [GaNe69, GaNe72].

Sea $\overline{P}$ el conjunto de soluciones posibles del programa lineal asociado a un problema de set partitioning,

$$\overline{P} = \{ x \in Rn \ / \ Ax=e, x \geq 0 \}$$

y sea

$$P = \text{conv} \{ x \in \overline{P} \ / \ x \text{ entero} \}$$

el polítopo de set partitioning.

La *forma disyuntiva normal* de un programa disyuntivo es

(DP) min $\{cx \,/\, x \in F\}$, donde

$$F = \left\{ x \in R^n \,/\, \bigvee_{h \in Q} (A^h x \geq b^h, x \geq 0) \right\}$$

siendo cada A^h una matriz de dimension $m_h x n$ y cada vector b^h de dimensión m_h, el símbolo $\vee$ significa que debe satisfacerse por lo menos uno de los $|Q|$ sistemas $A^h x \geq b^h$.

En un contexto en el que las desigualdades $\alpha x \geq \alpha_0$ implicadas por las restricciones de F puedan considerarse como planos secantes potenciales, resulta conveniente expresar el problema en términos de las variables no básicas asociadas a una solución óptima del problema lineal. Así, una desigualdad $\alpha x \geq \alpha_0$ violará la solución actual x=0 si y sólo si $\alpha_0 > 0$.

Sea Q^* el conjunto de índices $h \in Q$ tal que $\{x \in R^n \,/\, A^h x \geq b^h, x \geq 0\} \neq \emptyset$.

Teorema 14: $\forall x \in F$ se cumple la desigualdad $\alpha x \geq \alpha_0$ si y sólo si existen vectores $\theta^h \in R^m$, $h \in Q^*$, tales que
$\alpha \geq \theta^h A^h$, $\alpha_0 \leq \theta^h b^h$, $\theta^h \geq 0$, $h \in Q^*$.

Además, si conv F es cerrado y de dimensión completa entonces se cumplen las dos siguientes propiedades:

i) Si $\alpha_0 \neq 0$, $\alpha x \geq \alpha_0$ es una faceta de conv F si y sólo si $\alpha_0 \neq 0$ es un vértice de $F^{\#} = \{y \in R^n \,/\, y \geq \theta^h A^h, \theta^h b^h \geq \alpha_0, \theta^h \geq 0, \forall h \in Q^*\}$

ii) Si $\alpha x \geq 0$ es una faceta de conv F, entonces $\alpha \neq 0$ es una dirección extrema de $F^{\#}$.

En el caso de que alguna de las desigualdades que forman F se sustituya por una igualdad, el Teorema 14 sigue siendo cierto, sin más que eliminar la restricción de no negatividad de la componente del vector θ correspondiente a dicha igualdad.

Los vértices de $F^{\#}$, que definen la facetas de conv F, pueden obtenerse resolviendo un programa lineal. Cuando Q sea grande, esto puede resultar muy pesado a pesar de la estructura particular del programa. Sin embargo, relajando parte del conjunto de restricciones del problema de set partitioning y manteniendo únicamente un subconjunto adecuado, puede sacarse partido de la estructura del problema lineal resultante y resolverlo de forma trivial.

Supongamos, por ejemplo, que el programa lineal asociado a (SP) tiene una solución óptima de la forma

$$x_i = \bar{a}_{i0} + \sum_{j\in J} \bar{a}_{ij}\,(-t_j) \quad i\in I\cup J$$

donde I y J son los índices de variables básicas y no básicas respectivamente; es decir para $i\in J$ $\bar{a}_{i0}=0$ y $\bar{a}_{ij} = 0$ para $i\neq j$, $\bar{a}_{ij} =-1$ para $i=j$. Sea

$$\sum_{j\in Q} x_j = 1$$

una de las restricciones de (SP) tal que $Q'\neq\emptyset$, siendo $Q'=\{i\in Q/\ 0 < \bar{a}_{i0} <1\}$.

Teniendo en cuenta que la solución actual satisface esta igualdad, no puede utilizarse como plano secante. Ahora bien, esta ecuación implica una condición lógica que significa que exactamente una de las variables $x_j\in Q$ debe valer 1, siendo todas las demás igual a 0; es decir, la solución actual viola la expresión

$$\bigvee_{i\in Q} \left(x_i = 1,\ \sum_{j\in Q\setminus\{i\}} x_j = 0 \right)$$

que puede, por lo tanto utilizarse para generar un plano secante.

Sea $\sigma=(\sigma_i)$ un vector q-dimensional, donde $q=|Q|$, tal que $0\leq\sigma_i\leq 1$, $\forall i\in Q$. Si se utilizan los números σ_i, $1-\sigma_i$ para formar combinaciones lineales convexas de las dos ecuaciones que aparecen en la disyunción anterior, ésta puede reescribirse como

$$\bigvee_{i\in Q} \left[\ (1-\sigma_i)x_i + \sigma_i\Big(\sum_{h\in Q\setminus\{i\}} x_h \Big) = 1-\sigma_i \ \right]$$

Consideremos ahora la familia de las relajaciones del polítopo P cuyos miembros pertenecen a conv $F(\sigma)$, para algún $\sigma\in R^q$ tal que $0\leq\sigma_i\leq 1$, $i\in Q$, siendo

$$F(\sigma) = \left\{ t\in R^n \;\middle|\; \begin{array}{l} x_i = \bar{a}_{i0} + \sum_{j\in J} \bar{a}_{ij}\,(-t_j)\ ,\quad i\in I\cap Q \\ t_j \geq 0\ ,\quad j\in J \\ \bigvee_{i\in Q} \left[\ (1-\sigma_i)x_i + \sigma_i\Big(\sum_{h\in Q\setminus\{i\}} x_h \Big) = 1-\sigma_i \ \right] \end{array} \right.$$

El siguiente teorema se debe a Balas [Bal74] y en él se definen las facetas de conv $F(\sigma)$ que violan la solución actual.

Teorema 15: Para cualquier $\sigma \in R^q$ tal que $0 \leq \sigma_i \leq 1$, $\forall i \in Q$, la única facetas de conv $F(\sigma)$ que viola la solución actual $t_j=0$, $j \in J$, es

$$\sum_{j \in J} \alpha_j(\sigma)\, t_j \geq 1$$

donde

$$\alpha_j(s) = \max_{i \in Q} \Big(\sigma_i \sum_{h \in Q} \bar{a}_{hj} - \bar{a}_{ij} \Big) (1-\bar{a}_{ij})^{-1} \quad j \in J$$

Estos cortes no resultan computacionalmente difíciles de generar y además existe cierta libertad en la elección del parámetro s de forma que el corte resulte más potente en una u otra dirección. Una elección de s que hace que el corte resulte especialmente sencillo de calcular es

$$\sigma_i = (1-\bar{a}_{i0}) \quad \forall i \in Q, \text{ dando como coeficientes}$$

$$\alpha_j = \sum_{h \in Q} \bar{a}_{hj} - \min_{i \in Q} \bar{a}_{ij} (1 - \bar{a}_{i0})^{-1}$$

Además, es probable que estas desigualdades tengan algún coeficiente negativo, ya que los coeficientes $\bar{a}_{hj}$ de la tabla del Simplex actual no están restringidos en signo.

El siguiente teorema proporciona otro plano secante, que resulta todavía más sencillo de generar.

Teorema 16 [Bal76]: Sean Q y Q* definidos como en el Teorema 13. Dados $i_1, i_2 \in Q$, sean

$$S_1 = \Big\{ j \in J \;/\; \frac{\bar{a}_{i_1 j}}{\bar{a}_{i_1 0}} \geq \frac{\bar{a}_{i_2 j}}{\bar{a}_{i_2 0}} \Big\} \qquad S_2 = J \setminus S_1$$

entonces la desigualdad

$$\sum_{j \in J} \alpha_j t_j \geq 1$$

es un corte válido, siendo los coeficientes α_j

$$\max \Big\{ \frac{\bar{a}_{i_1 j} - 1}{\bar{a}_{i_1 0}}, \frac{\bar{a}_{i_2 j}}{\bar{a}_{i_2 0}} \Big\} \qquad j \in Q \cap S_1$$

$$\max \left\{ \frac{\bar{a}_{i_1 j}}{\bar{a}_{i_1 0}}, \frac{\bar{a}_{i_2 j} - 1}{\bar{a}_{i_2 0}} \right\} \qquad j \in Q \cap S_2$$

$$\min \left\{ \frac{\bar{a}_{i_1 j}}{\bar{a}_{i_1 0}}, \ \max \left\{ \frac{\bar{a}_{i_1 j} - 1}{\bar{a}_{i_1 0}}, \frac{\bar{a}_{i_2 j} + 1}{\bar{a}_{i_2 0}} \right\} \right\} \qquad j \in (J \backslash Q) \cap S_1$$

$$\min \left\{ \frac{\bar{a}_{i_2 j}}{\bar{a}_{i_2 0}}, \ \max \left\{ \frac{\bar{a}_{i_1 j} + 1}{\bar{a}_{i_1 0}}, \frac{\bar{a}_{i_2 j} - 1}{\bar{a}_{i_2 0}} \right\} \right\} \qquad j \in (J \backslash Q) \cap S_2$$

Este corte puede reforzarse de forma que se obtenga la única faceta de conv F' que viola la solución actual, donde

$$F' = \quad t \in R^n \quad \begin{array}{ll} x_i = \bar{a}_{i0} + \sum_{j \in J} \bar{a}_{ij} (-t_j) & i \in I \cap Q \\ t_j \geq 0 & j \in J \\ \left(\sum_{i \in Q_1} x_i \leq 0 \right) \vee \left(\sum_{i \in Q_2} x_i \leq 0 \right) & \end{array}$$

siendo $Q_k = \{i_1\} \cup (Q \cap S)$, $k=1,2$.

En la elección de los índices de variables i_1 y i_2, parece razonable escoger aquellos que tengan el mayor número posible de coeficientes negativos en sus correspondientes filas de la tabla del Simplex ya que de esta forma se obtendrá un corte que, como mínimo, tendrá todos esos coeficientes negativos.

Los cortes que se han obtenido via programación disyuntiva pueden posteriormente reforzarse, con un coste computacional razonable, utilizando las restricciones de integridad en las variables básicas [BaJe75].

Independientemente de que estas desigualdades estén o no reforzadas, dado el procedimiento utilizado para su generación, éstas pueden utilizarse en un contexto algorítmico basado en la utilización del Simplex dual, de una forma análoga a la de los cortes de Gomory. Además, teniendo en cuenta que los cortes anteriores toman en consideración la estructura específica de los problemas de set partitioning, es razonable suponer que su utilización acelerará la convergencia de este tipo de algoritmos.

I.3.5 Otras desigualdades válidas

En el último apartado se han estudiado algunas facetas de relajaciones del polítopo de (SP) que pueden utilizarse como planos secantes en un esquema algorítmico basado en la utilización del método del Simplex. Este enfoque presenta, sin embargo, un inconveniente claro en el caso de los problemas (SP): se trata de la resolución de la relajación lineal para la obtención de la tabla fraccional asociada a una solución óptima del problema relajado. Este inconveniente aparece, no sólo en problemas de (SP), sino, en general, en todo tipo de problemas en los que la degeneración masiva se produzca de forma habitual. Veremos que, en el caso de (SP), existen dos tipos de alternativas que pueden evitar este inconveniente. La primera de ellas es la obtención de desigualdades válidas, que puedan utilizarse como planos secantes, cuya generación no venga dada a través de la tabla del Simplex. Un ejemplo de lo anterior son, como ya se ha visto, los cortes obtenidos a partir de disyunciones. Veremos en el próximo apartado que existen otras posibilidades en este sentido. La segunda de las alternativas algorítmicas mencionadas, pasa por la caracterización de ciertas propiedades que deben verificar los vértices asociados a soluciones enteras en el polítopo correspondiente a la relajación lineal ($\overline{P}$). Ello permitirá obtener la información referente a la tabla del Simplex asociada a dicha solución, y en consecuencia generar planos secantes del tipo de los del apartado anterior, sin necesidad de resolver el algorimo del Simplex.

En este apartado se introducen unas desigualdades válidas para (SP) que se derivan de las condiciones lógicas implicadas por el conjunto de restricciones (Bal77]). Primero se verá una clase canónica de desigualdades homogéneas, ya que a partir de ellas se obtendrán todas las demás y se estudiará bajo qué condiciones son planos secantes, maximales y facetas. Posteriormente se estudiarán las desigualdades compuestas de este tipo.

Siguiendo con la notación introducida anteriormente, sean

$$M_k = \{ i\in M / a_{ik} = 1 \} \ , \ \overline{M}_k = M\backslash M_k \ , \ k\in N,$$
$$N_i = \{ k\in N / a_{ik} = 1 \} \ , \ \overline{N}_i = N\backslash N_i \ , \ i\in M$$
$$N_{ik} = \{ j\in N_i / a_j a_k = 0 \}, \ i\in \overline{M}_k \ , \ k\in N$$

<u>Definición 7</u>: Una desigualdad válida $\pi x\geq\pi_0$ se dice que es *maximal* si para cualquier $k\in N$ y para cualquier $\pi'_k>\pi_k$ existe $x\in P$ tal que

$$\pi'_k x_k + \sum_{j\in N\backslash\{k\}} \pi_j x_j > \pi_0$$

Una desigualdad se dice que es *elemental* si es de la forma

$$x_k - \sum_{j\in Q} x_j \leq 0 \qquad Q\subseteq N_{ik} \text{ para algún } i\in \overline{M}_k \qquad (9)$$

Teniendo en cuenta que las restricciones de (SP) deben todas satisfacerse como estricta igualdad, es fácil observar que $\forall k\in N$ debe existir alguna desigualdad del tipo (9) que sea válida para (SP). Esto es cierto ya que, dado $k\in N$, N_{ik} es el conjunto de índices de variables j que son ortogonales a a_k para las que el elemento a_{ij} de la matriz vale 1. Por lo tanto, por lo menos una de las variables x_j, $j\in N_{ik}$ deberá valer 1 cuando $x_k=1$, ya que $i \in \overline{M}_k \Rightarrow a_{ik}=0$.

Cuando se toma $Q=N_{ik}$ $\Leftarrow k\in N$, $i \in \overline{M}_k$, se tiene la clase de desigualdades

$$x_k - \sum_{j\in N_{ik}} x_j \leq 0 \qquad (10)$$

Esta clase de inecuaciones, es un caso particular de (9), que además es válida para (SP) ya que todas las desigualdades de este tipo son satisfechas por cualquier punto x del polítopo P de set partitioning. Sin embargo, no caracterizan dicho polítopo, ya que es fácil construir vectores enteros x que satisfagan toda la clase de desigualdades (10) y que, sin embargo, no pertenezcan a P.

Existe un primer resultado interesante [Bal77] que demuestra que cualquier vértice, distinto del origen, del polítopo Pk de set packing que satisfaga toda la familia de desigualdades (10) es un vértice de P. Como consecuencia, cualquier vértice no nulo de Pk que no pertenezca a P, violará alguna desigualdad del tipo (10) y, recíprocamente, cada desigualdad de la familia (10) violará algún vértice $x\in Pk\backslash P$.

En un contexto más general se tienen los siguientes resultados:

<u>Teorema 17</u>: Dados $k \in K$, $i \in \overline{M}_k$, $Q \subseteq N_{ik}$, la desigualdad del tipo (9) dada por

$$x_k - \sum_{j\in Q} x_j \leq 0$$

es válida si y sólo si $x\in \text{Vert } P$ y $x_k=1$ implica $x_j=0$, $\forall j\in N_{ik}\backslash Q$.

<u>Teorema 18</u>: Dados $k \in K$, $i \in \overline{M}_k$, $Q \subseteq N_{ik}$, una desigualdad (válida) del tipo (9) es un corte (viola alguna solución $x \in \overline{P}\backslash P$) si y sólo si no existe $\theta\in R_n$ tal que:

$$\theta a_j \geq \begin{cases} -1 & j=k \\ 1 & j\in Q \\ 0 & j\in N\backslash Q \end{cases}$$

$$\theta e \geq 0$$

Una vez conocido bajo qué condiciones una desigualdad del tipo (9) es válida y cuando se trata de un corte, la siguiente cuestión que aparece es la de saber, en el caso en que se trate de un corte, cuando dicho corte es maximal. Para ello será necesario excluir el caso degenerado en el que para alguna variable $j\in N$, $x\in P$ implique $x_j=0$.

Teorema 19: Dados $k \in N$, $i \in \overline{M}_k$, supongamos que (9) es una desigualdad válida para un conjunto $Q \subseteq N_{ik}$; supongamos, además, que $\forall j\in N$ existe $x\in \text{Vert } P$ tal que $x_j=1$. Entonces (9) es maximal si y sólo si:

i) $\forall j \in Q$ existe $x \in \text{Vert } P$ tal que $x_j=x_k=1$
ii) $\forall j \in \overline{N}_i\backslash\{k\}$ existe $x \in \text{Vert } P$ tal que $x_j=1$ y $x_k\geq x_h$, $\forall h\in Q$.

El siguiente corolario nos proporciona un procedimiento para reforzar desigualdades del tipo (9) en el caso que sean válidas pero no maximales.

Corolario 1: Supongamos que para todo $j\in N$ existe $x\in \text{Vert } P$ tal que $x_j=1$. Para algún $k\in N$, $i\in \overline{M}_k$, sea (9) una desigualdad válida pero no maximal con $Q \subseteq N_{ik}$. Sean S_1 y S_2 los conjuntos de índices de variables para los que se violan las condiciones i) y ii) del Teorema 19. Entonces todo $x\in P$ satisface la desigualdad

$$x_k - \sum_{j\in Q\backslash S_1} x_j \leq 0 \qquad (11)$$

y las inecuaciones

$$x_k + \sum_{j\in S_2\cap T} x_j - \sum_{j\in Q} x_j \leq 0 \qquad (12)$$

para todo conjunto $T \subseteq \overline{N}_i\backslash\{k\}$ tal que $a_h a_j\neq 0$ $\forall h,j\in T$.

Evidentemente si existe algún subconjunto S' de N, tal que $\forall j \in S'$ no se cumple la condición de no degeneración impuesta en el teorema y el corolario anterior, entonces los coeficientes de cada x_j, $j \in S'$ pueden hacerse arbitrariamente grandes, además de los cambios permitidos para las variables $j \in S=S_1 \cup S_2$ en virtud del corolario 1.

El corolario 1 proporciona un método para reforzar desigualdades válidas siempre que se conozcan los conjuntos S_1 y S_2. Posteriormente se verá algún procedimiento para identificar tales conjuntos.

Teniendo en cuenta que la dimensión del polítopo P de set partitioning siempre es menor que n (de hecho es como máximo n-m), una desigualdad que sea válida para P, puede definir una cara impropia de P; este será el caso cuando la inecuación se cumpla como estricta igualdad para todo $x \in P$. Surge, por lo tanto, de forma natural la cuestión que se plantea en el siguiente teorema, que es la de saber cuando una desigualdad elemental que sea maximal, es una cara de dimensión máxima (faceta o cara impropia) de P.

<u>Teorema 20</u>: Sea (9) una desigualdad maximal (válida) para P, con $Q \subseteq N_{ik}$ para algún $k \in N$, $i \in \overline{M}_k$. Sean $N'= N \backslash Q \cup \{k\}$, y

$$P_{N'} = P \cap \{x \in R^n \,/\, x_j = 0,\ \forall j \in Q \cup \{k\}\}$$

Entonces, $\dim P \geq \dim P_{N'} + q$, donde $q=|Q|$.
Si $\dim P = \dim P_{N'} + q$ entonces (9) es una cara impropia de P
Si $\dim P = \dim P_{N'} + q + 1$, entonces (9) es ó bien una faceta o bien una cara impropia de P.

A continuación se exponen dos procedimientos para obtener desigualdades válidas que sean más potentes que las desigualdades elementales válidas, para el caso en el que éstas no sean maximales. El primer método utiliza información del resto de desigualdades elementales en las que la variable x_k tiene un coeficiente positivo. En el segundo método, la información que se utiliza es la referente a desigualdades elementales en las que x_j tiene un coeficiente positivo para algún $j \in Q$.

<u>Teorema 21</u>: Dado $k \in N$, supongamos que los conjuntos de índices $Q_i \subseteq N_{ik}$, $i \in \overline{M}_k$, son tales que $\forall x \in P$, se satisfacen las desigualdades

$$x_k - \sum_{j \in Q_i} x_j \leq 0 \quad , \quad i \in \overline{M}_k \qquad (13)$$

Para cada $j \in \bigcup_{j \in \overline{M}_k} Q_h$, sea

$$Q(j) = \bigcup_{\substack{h \in \overline{M}_k \\ j \in Q_h}} Q_h \setminus \{j\}$$

y para cada $i \in \overline{M}_k$, sea

$$T_i = \{ j \in Q_i \,/\, Q(j) \supseteq Q_h \text{ para algún } h \in \overline{M}_k \}$$

Entonces $\forall x \in P$ se satisfacen las desigualdades

$$x_k - \sum_{j \in Q_i \setminus T_i} x_j \leq 0$$

El Teorema 21 puede utilizarse para reforzar de forma iterativa desigualdades. Primero, se sustituye el conjunto N_{ik} por $Q_i = N_i \setminus T_i$, posteriormente puede aplicarse a las desigualdades reforzadas obtenidas, hasta que no pueda reforzarse más.

Una aplicación interesante del teorema anterior aparece cuando se desean reforzar todas las desigualdades elementales en las que una cierta variable x_k tenga coeficiente positivo. En este caso es conveniente trabajar con el conjunto

$$Q_0 = \bigcup_{i \in \overline{M}_k} Q_i$$

y en lugar de construir los conjuntos T_i, comprobando cada $j \in Q_i$, $i \in \overline{M}_k$, construir directamente el conjunto

$$T = \bigcup_{i \in M_k} T_i = \{ j \in Q_0 \,/\, Q(j) \supseteq Q_h \text{ para algún } h \in \overline{M}_k \}$$

inspeccionando una sola vez cada índice $j \in Q_0$, utilizando posteriormente $Q_i \setminus T$ en vez de $Q_i \setminus T_i$.

<u>Teorema 22:</u> Sean los conjuntos de índices $Q_{ik} \subseteq N_{ik}$, $i \in \overline{M}_k$, $k \in N$, tales que $\forall x \in P$ se satisfacen las desigualdades

$$x_k - \sum_{j\in Q_{ik}} x_j \le 0, \quad i\in \overline{M}_k, \quad k\in N$$

Para cada $i \in \overline{M}_k$, $k \in N$, sea el conjunto

$$U_{ik} = \{ j\in Q_{ik} \,/\, Q_{hk} \cap Q_{hj} = \emptyset \text{ para algún } h\in \overline{M}_k \cap \overline{M}_j \}$$

Entonces, $\forall x\in P$ se satisfacen las desigualdades

$$x_k - \sum_{j\in Q_{ik}\setminus U_{ik}} x_j \le 0 \;, \quad i\in \overline{M}_k, k\in N$$

El Teorema 22 puede utilizarse de forma análoga al 21 para reforzar, de forma secuencial desigualdades elementales.

Las desigualdades que se han visto hasta ahora en este apartado son todas homogéneas. Existen, sin embargo, dos clases de inecuaciones no homogéneas que, en el conjunto de soluciones posibles de (SP) son equivalentes a las desigualdades elementales (9).

Teorema 23: Dados $k \in N$, $i\in \overline{M}_k$, sea $Q \subseteq N_{ik}$. Una solución posible x para la relajación lineal del problema de set partitioning satisface una de las siguientes inecuaciones si y sólo si las satisface todas ellas:

$$x_k - \sum_{j\in Q} x_j \le 0 \qquad (9)$$

$$x_k + \sum_{j\in N_i\setminus Q} x_j \le 1 \qquad (14)$$

$$\sum_{j\in N_h\setminus\{k\}} x_j + \sum_{j\in Q} x_j \ge 1, \quad h\in M_k \qquad (15)$$

Es importante resaltar que el teorema anterior sigue siendo cierto si se sustituye la condición de que x sea solución de la relajación lineal del problema de set partitioning por la condición de que x cumpla las restricciones del problema entero (SP).

Otra observación importante es que las tres desigualdades anteriores sólo son equivalentes en el conjunto de puntos $x\in R^n$ tales que $Ax=e$. Es fácil construir ejemplos de distintos vectores x que no cumplen las restricciones $Ax=e$ pero que sí satisfacen dos de las tres inecuaciones anteriores.

Las inecuaciones (10) son del tipo de las restricciones de set packing; sin embargo, en general no serán válidas para el polítopo Pk.

Teorema 24: La desigualdad 14 viola algún punto $x \in Pk \backslash P$ si y sólo si $Q \neq N_{ik}$.

Las desigualdades (14), son un caso particular de las desigualdades del tipo

$$\sum_{j \in V} x_j \leq 1 \qquad (16)$$

introducidas en I.3.1 y pueden interpretarse en términos de grafo de intersección G_A. En concreto, una desigualdad (14) será válida para Pk si y sólo si $\{k\} \cup (N_i \backslash Q)$ es el conjunto de nodos de un subgrafo completo de G_A. Por el Teorema 24 estas desigualdades serán precisamente aquellas para las que $Q=N_{ik}$.

Por analogía con el grafo G_A asociado al polítopo de set packing que tiene un arco para cada par $i,j \in N$ tal que no existe ningún punto de Pk tal que $x_i=x_j=1$ $(a_i a_j \geq 1)$, puede definirse el *grafo de intersección fuerte G(A)* asociado al polítopo de set partitioning, como aquél que tenga un nodo asociado a cada columna de la matriz A y un arco para cada par $i,j \in N$ tal que no existe ningún punto del polítopo P ($x \in \{0,1\}^n$, t.q. $Ax=e$) tal que $x_i=x_j=1$. Además es fácil comprobar que G_A es un subgrafo de G(A). En este contexto se tiene:

Teorema 25:

i) $\forall x \in P$ se satisface la desigualdad (16) si y sólo si V es el conjunto de nodos de algún subgrafo completo G' de G(A).

ii). Supongamos que $\forall j \in V$ existe $x \in P$ tal que $x_j=1$, y que $\forall x \in P$ se satisface (16). Entonces (16) es una desigualda maximal si y sólo si V es el conjunto de nodos de un clique de G(A).

Las desigualdades elementales (9) pueden combinarse entre ellas para generar otras desigualdades que siguen siendo homogéneas. Las nuevas desigualdades son más potentes que las que se obtienen sin más que sumar las desigualdades elementales utilizadas para su obtención. Además todos los coeficientes de la nueva desiguladad siguen siendo 0, 1 ó -1. El siguiente teorema ilusra una de las posibles clases de desigualdades compuestas que se pueden obtener.

Teorema 26: Sea $K \cup N$ tal que

$$\sum_{j \in K} x_j \leq 1 \ , \ \forall x \in \text{Vert } P$$

y sea

$$L(K) = \{ \ j \in N \backslash K \ / \ a^j a^k = 0 \ \text{ para algún } k \in K \ \}$$

Entonces $\forall x \in P$ se satisface la desigualdad

$$\sum_{j \in K} x_j - \sum_{j \in S} x_j \leq 0 \qquad \text{donde } S \subseteq L(K)$$

si y sólo si

$$\sum_{j \in Q} a_j \neq e - a_k \qquad \forall k \in K, \quad \forall Q \subseteq L(K) \backslash S$$

I.3.6 Vértices adyacentes de los polítopos de set partitioning y de set packing

Ya se ha comentado anteriormente que uno de los principales obstáculos en el diseño de algoritmos de planos secantes que utilicen la relajación lineal para resolver problemas (SP) es la obtención de las tablas del Simplex, debido a la degeneración masiva que presentan habitualmente este tipo de problemas y a que no existe ninguna técnica especializada que utilice la estructura particular de los mismos. En ese sentido, resulta potencialmente muy interesante conocer que existen ciertas propiedades de adyacencia, que verifican los vértices de los polítopos de set packing y de set partitioning, que permiten resolver los problemas (SP) mediante una versión modificada del algorimo del Simplex, que genera únicamente soluciones enteras y sin necesidad de utilizar planos secantes. En este apartado (LSP) denotará la relajación lineal asociada a un problema (SP)

Considerando que dos vértices de un polítopo son adyacentes cuando pertenecen a una misma arista (cara unidimensional) del polítopo, es natural definir dos *bases adyacentes* como un par de bases de (LSP) tales que difieran entre sí en una única columna; es decir, pueden obtenerse una a partir de la otra mediante una pivotación.

Teorema 27 [BaPa72]: Sea x^1 una solución posible entera (pero no óptima) de (LSP) asociada a una base B_1. Si x^2 es una solución óptima para (SP), entonces existe una sucesión de bases adyacentes B_{10}, B_{11}, B_{12}, ...,B_{1p}, tal que $B_{10}=B_1$, $B_{1p}=B_2$ es la base asociada a x^2 y:

i) las soluciones básicas $x^{1i} = B_{1i}^{-1} e$, $i=0,1,...,p$, son todas factibles y enteras.

ii) $cx^{10} \geq cx^{11} \geq \ldots \geq cx^{1p}$.

iii) $p=|J_1 \cap Q_2|$, siendo J_1 el conjunto de índices de variables no básicas en la base B_1 y $Q_2=\{j \in N \,/\, x_j^2=1\}$.

El teorema anterior implica que dada una solución posible básica entera x^1 de (LSP), existe una solución entera mejor que ella si y sólo si existe una solución posible, adyacente a x^1, que es un vértice del polítopo $\overline{P}$ asociado a (LSP), adyacente a x^1. La dificultad que surge en la utilización práctica de este teorema es la identificación de tales vértices adyacentes. Ello se debe a que, debido a la degeneración masiva que poseen los problemas (SP), en general existen muchas bases asociadas a la misma solución y muchos vértices de $\overline{P}$ adyacentes a un vértice dado. Además, en esta situación, las técnicas lexicográficas no resultan útiles ya que la sucesión de pivotaciones necesaria para alcanzar un vértice adyacente, puede incluir un pivot en un coeficiente negativo perteneciente a una fila degenerada ($a_{i0}=0$). Por lo tanto, para poder aplicar el Teorema 27 será necesario conocer las propiedades de adyacencia entre los vértices enteros de $\overline{P}$. Este conocimiento permitirá caracterizar un vértice entero de $\overline{P}$ en función de cualquier otro vértice entero de $\overline{P}$. Posteriormente, esta caracterización permitirá generar todas las aristas de Pk que conecten un vértice entero dado con vértices adyacentes enteros.

Dada una solución posible básica (s.p.b.) entera x^1 de (LSP), de base B^1 y cuyos conjuntos de índices de variables básicas y no básicas son $I_1=\{1,\ldots,m\}$ y J_1 ($I_1 \cup J_1=N$) respectivamente, denotaremos

$$\bar{a}_j = B_1^{-1} a_j , \qquad \bar{a}^j = \begin{bmatrix} \bar{a}_j \\ -e_j \end{bmatrix}$$

donde e_j es el vector unitario n-m dimensional que tiene un 1 en el lugar j, y

$$Q_1 = \{ j \in N \,/\, x_j^1 = 1 \}, \qquad \overline{Q}_1 = N \setminus Q_1$$

<u>Teorema 28</u> [BaPa73]: Sea x^1 una s.p.b. entera de (LSP). Entonces x^2 es una s.p.b. entera de (LSP) si y sólo si existe Q⊂J1 tal que

$$\sum_{j \in Q} \bar{a}_{kj} = \begin{cases} 0 \text{ ó } 1 & k \in Q_1 \\ 0 \text{ ó } -1 & k \in I_1 \cap \overline{Q}_1 \end{cases}$$

y

$$x_j^2 = \begin{cases} 1 & j \in Q_2 = Q \cup S \\ 0 & \text{en otro caso} \end{cases}$$

donde

$$S = \{ k \in Q \,/\, \sum_{j \in Q} \bar{a}_{kj} = 0 \} \cup \{ k \in I_1 \cap \bar{Q}_1 \,/\, \sum_{j \in Q} \bar{a}_{kj} = -1 \}$$

Además, cuando se cumple la condición anterior, entonces

$$x^2 = x^1 - \sum_{j \in Q} \bar{a}^j$$

Como consecuencia del Teorema 28, se tiene que x^2 es un vértice entero adyacente a x^1 en $\bar{P}$ si y sólo si

$$\Lambda = \{ x \in R^n \,/\, x = x^1 - (\sum_{j \in Q} \bar{a}^j)\, \lambda \,,\ 0 \leq \lambda \leq 1 \}$$

es una arista de $\bar{P}$ que también es una arista de P.

Definición 8: Dada una s.p.b. x^1, un subconjunto $Q \subseteq J_1$ que cumple la condición del Teorema 28 se dice que es *descomponible* si puede particionarse en dos subconjuntos Q^* y Q^{**} tales que la condición del teorema sigue siendo cierta para Q^* y Q^{**} respectivamente.

El siguiente teorema proporciona una caracterización de los pares de vértices enteros adyacentes en $\bar{P}$ en función de una de las bases asociadas a uno de los términos del par. A pesar de que esta caracterización puede utilizarse directamente para generar vértices enteros adyacentes a uno dado, a partir de cualquier tabla del Simplex asociada al mismo, resulta conveniente tener una caracterización equivalente en términos de la matriz A, sin relación con ninguna base específica. Por este motivo, resulta conveniente definir un concepto análogo al de la descomponibilidad en términos de las columnas de la matriz A.

Definición 9: Dado cualquier par de subconjuntos $S \subset N$, $T \subset N$, tales que

$$\sum_{j \in S} a_j = \sum_{j \in T} a_j \leq e \qquad (17)$$

se dice que (S,T) es *descomponible por pares* (*pairwise decomposable*) si existe un par de subconjuntos propios no vacíos (S', T'), $S' \subset S$, $T' \subset T$ tal que la condición (17) sigue siendo cierta cuando se sustituyen S y T por S' y T' respectivamente.

Teorema 29: Sean x, y dos vértices cualesquiera de P, y sean

$$Q(z) = \{ j \in N \,/\, z_j = 1 \}, \quad \bar{Q}(z) = \{ j \in N \,/\, z_j = 0 \}$$

definidos para z=x ó bien z=y, indistintamente.

Además, sea J(x) el conjunto de índices de variables no básicas para alguna base (elegida arbitrariamente) de x. Entonces, las siguientes afirmaciones son equivalentes:

i) x e y son adyacentes en $\overline{P}$.

ii) x e y son adyacentes en P.

iii) $Q(y) \cap J(x)$ no es descomponible.

iv) $[\, Q(x) \cap \overline{Q(y)}\, ,\ \overline{Q(x)} \cap Q(y)\,]$ no es descomponible por pares.

El Teorema 28 situa los problemas (SP) en una clase de problemas que tienen una propiedad geométrica más fuerte que la, ya conocida para todos los programas enteros 0-1, de que todos los vértices de la envolvente convexa del conjunto de soluciones posibles del programa son vértices del polítopo asociado a la relajación lineal del problema. En particular, en virtud del Teorema 28, las aristas de P serán tambien aristas de $\overline{P}$.

I.4. Tests lógicos de eliminación

Antes de pasar a hacer un breve repaso de las principales alternativas existentes para resolver problemas (SP) resulta imprescindible comentar la importancia de los procedimientos de reducción previos que pueden aplicarse a estos problemas [GaNe72, Sa75, BaPa79]. La utilización de estos procedimientos permite fijar variables al valor 0 ó 1, con lo cual se pueden reducir considerablemante las dimensiones de los problemas.

El conocimiento de ciertas condiciones lógicas implicadas por la estructura de la matriz de restricciones puede proporcionar, en una fase previa a la resolución de los problemas, información suficiente como para eliminar de dicha matriz algunas de las filas y las columnas. Este tipo de procedimiento previo tiene, sin duda, gran importancia ya que, a pesar de la sencillez de concepto y el poco esfuerzo computacional que requiere la implementación de los test que se aplican, reduce sensiblemente las dimensiones del problema original fijando algunas de las variables a 0 ó a 1. La utilización de estos procedimientos, a pesar de estar justificada en cualquier caso, resulta especialmente efectiva en problemas en los que la densidad de la matriz de restricciones es baja.

Otra de las cualidades de estos tests resulta ser su potencial utilización en un esquema algorítmico general de enumeración implícita. Resulta evidente que, si la estructura de los subproblemas generados en cada uno de los nodos del árbol de exploración es también la de un problema de (SP), la utilización de los test de eliminación reducirá también sus dimensiones con lo que el análisis del subproblema generado será más sencillo, acelerando de este modo la convergencia total del algoritmo de enumeración.

Veremos en el capítulo posterior otra aplicación muy interesante de estos procedimientos. En particular, los tests de reducción se aplicarán en algún paso intermedio de ciertas heurísticas de tipo greedy para encontrar soluciones posibles de problemas de (SP). De esta forma, no sólo se acelerará también sensiblemente la convergencia de la heurística, sino que se ayudará a mejorar la calidad de la solución encontrada.

A continuación se exponen algunos de estos tests:

Sean M y N los conjuntos de índices de filas y de columnas de la matriz A. Denotando por a^i la i-ésima fila y por a_j la j-ésima columna de dicha matriz, Test 1. Si para algún $i \in M$, $k \in N$ se tiene que $a_{ik}=1$ y $a_{ij}=0$ $\forall j \in N \backslash \{k\}$, entonces:

a) Fijar $x_k=1$ y eliminar la columna k.

b) Eliminar todas las columnas $j \in N \backslash \{k\}$ tales que $a_k a_j \geq 1$.

c) Eliminar todas las filas $h \in M$ tales que $a_{hk}=1$.

Test 2. Si para algún $i,h \in M$ se tiene que $a^i \leq a^k$, entonces,

a) Eliminar la fila h.

b) Eliminar todas las columnas $j \in N$ tales que $a_{ij}=0$ y $a_{hj}=1$.

Test 3. Si para algún $k \in N$ y para algún subconjunto $N' \subset N$, se tiene que

$$\sum_{j \in N'} a_j = a_k \quad \text{y} \quad \sum_{j \in N'} c_j \leq c_k$$

entonces se puede eliminar la columna k.

Test 4. Para cada $i \in M$ sea $N_i = \{j \in N \,/\, a_{ij}=1\}$. Si para algún $i \in M$ se tiene que $a_k a_j \geq 1$ $\forall j \in N_i$ entonces se puede eliminar la columna k.

El primer test fija a 1 todas las variables correspondientes a filas que tengan un único elemento ya que cualquier solución posible deberá satisfacer dichas restricciones y esas variables son las únicas que lo consiguen. Posteriormente se hacen las actualizaciones corresponcientes.

El segundo test elimina aquellas filas cuyo conjunto de elementos contenga los elementos de alguna otra fila. Evidentemente cualquier solución posible que satisfaga la restricción con menos elementos, satisfará la que se ha eliminado.

El tercer test elimina (fijando a 0) aquellas variables cuyo conjunto de elementos sea la unión de los de otras columnas tales que la suma total de sus coeficientes de coste no supere al de la variable eliminada. Cualquier solución posible que contuviese la variable eliminada podría mejorarse sustituyendo esa componente por todas las variables tales que la suma de sus correspondientes columnas coincide con la variable eliminada, ya que la suma de sus coeficientes de coste resulta mejor. Obviamente, este test está planteado para problemas de minimización de (SP). En el caso, poco probable, que se trate de un problema de maximización, el signo de la desigualdad en la condición referente a la suma de los costes deberá ser el contrario.

El cuarto de los test resulta computacionalmente algo más complicado de implementar que los anteriores. En cualquier caso su utilización resulta provechosa. Este test elimina todas aquellas columnas que resulten ser no ortogonales con todas las componentes de alguna fila determinada. Evidentemente para satisfacer esa fila determinada debería eliminarse la columna considerada.

I.5. Alternativas algorítmicas

A la hora de resolver problemas de set partitioning se presentan distintas alternativas algorítmicas que son, la mayoría de las veces, versiones especializadas de algoritmos generales para problemas enteros de programación matemática. La más clásica consiste en utilizar algún procedimiento de enumeración implícita que particione la matriz por bloques de columnas y que explore un cierto árbol asignando valores a las variables en base a las condiciones lógicas que se derivan de la estructura de la matriz. Los ejemplos más conocidos de este tipo de algoritmos para (SP) son los desarrollados por Garfinkel y Nemhauser [GaNe69] y Marsten [Mar74]. En ambos casos se utiliza la información obtenida en la resolución de la relajación lineal ordinaria para acotar los problemas y eliminar eventualmente algúno de los nodos.

Otro tipo de procedimiento que se ha utilizado son los métodos de planos secantes basádos en la información de la tabla óptima del algoritmo del Simplex asociado a la relajación lineal. Este enfoque se basa en la apreciación de que a menudo las soluciones óptimas asociadas a las relajaciones lineales de los problemas resultan ser enteras y por lo tanto óptimas para los problemas originales. Existen referencias [BaPa79] de un trabajo no publicado de Martin en 1969 en el que se generaba una versión específica de planos secantes de Gomory para la resolución de problemas (SP). También Délorme [Del74] ha utilizado este enfoque para la resolución de problemas de (SP) con dos restricciones adicionales que aparecen en la generación de turnos en el problema de asignación de tripulaciones. También , en un trabajo de Salkin y Konkal [SaKo73], se ha utilizado el algoritmo "all integer" de Gomory para diseñar un método específico para problemas (SP) de generación de planos secantes basado en las tablas del Simplex.

Los algoritmos de plano secante anteriores se basan todos ellos en la utilización de planos secantes obtenidos a partir de la tabla del Simplex. Ya se ha comentado anteriormente como, debido a la degeneración masiva que presentan en general los problemas de (SP) la obtención de esta tabla óptima suele presentar problemas. Por ese motivo, en los últimos años, se ha intentado desarrollar métodos que sean especializaciones del Simplex para la estructura específica de las relajaciones lineales asociadas a los problemas de set partitioning, o bien métodos para obtener las soluciones óptimas de dicha relajación lineal sin necesidad de aplicar el algoritmo del Simplex.

Dentro de esta primera línea se encuentran los algoritmos de generación de columnas que son versiones específicas del algoritmo del Simplex que tiennen en cuenta la estructura particular que aparece en las relajaciones lineales de los problemas de set partitioning (LSP).

Estos algoritmos se basan en las propiedades de adyacencia ya comentadas en el apartado anterior que permiten obtener una sucesión de soluciones todas ellas enteras y que converge a una solución óptima. Dos ejemplos de estos algoritmos son los desarrollados por Balas, Gerritsen y Padberg [BaGePa74] y posteriormente por Marsten y Shepardson [MaSh81].

La segunda de las vías apuntadas para obtener una solución óptima de (LSP) sin necesidad de utilizar el algoritmo del Simplex consiste en la obtención de dicha solución por un método del tipo Subgradiente [HeKa71, HeWoCr74]. Este enfoque es actualmente muy utilizado y existen numerosos trabajos dedicados al mismo. Algunos de los más originales son el de Samuelsson [Sam74] y el de Nemhauser y Weber [NeWe79]. El trabajo de Samuelsson formula los problemas de (SP) como problemas de (SPk) y posteriormente utiliza simultáneamente los problemas lineales primal y dual en un algorimo de tipo subgradiente para obtener la solución óptima de la relajación lineal. En un trabajo posterior debido a Balas y Samuelsson [BaSa74] se utiliza este enfoque en un algoritmo de plano secante.

Nemhauser y Weber [NeWe79] formulan en su trabajo el problema (SP) como un problema equivalente de matching perfecto al que aplican la relajación lagrangiana ordinaria que resuelve mediante un procedimiento de tipo subgradiente.

Existe un enfoque algorítmico de planos secantes que resulta, sin lugar a dudas, más sugerente que todos los anteriores. Se trata de procedimientos que obtienen desigualdades válidas sin considerar para nada la relajación lineal. Ya se ha visto cómo la programación disyuntiva ofrece herramientas para ello, así como los distintos tipos de desigualdades válidas que pueden obtenerse sin necesidad de resolver (LSP) [Bal77]. Este tipo de aproximaciones no ha sido, sin embargo, ampliamente probado computacionalmente, aunque existen trabajos [Barce85, CaMa83, Cam83a, Cam83b, Cam83c] en los que los resultados obtenidos resultan muy prometedores.

Como conclusión, esta introducción deja claramente sentada la dificultad que presentan los problemas de esta familia, lo cual explica el relativo fracaso en muchos casos de los algoritmos basados en un único procedimiento. Sin embargo, esta característica es común a muchos de los procedimientos para la resolución de problemas con estructura combinatoria. Por este motivo, la tendencia actual [Barce82] de los métodos algorítmicos para resolver problemas de estructura combinatoria, y en particular los de tipo (SP) y (SC), apunta al diseño de algorítmos híbridos.

Este tipo de algoritmos es el que ofrece sin duda más garantías de éxito, ya que en ellos se combinan algunos de los métodos antes reseñados, potenciándose de esta manera cualidades individuales de cada uno de ellos. Sin lugar a dudas, ha sido Balas [BaSa74, Bal77, Bal79, BaHo79] quien más ha potenciado la utilización de este tipo de métodos. Posteriormente, Campello y Maculan [CaMa83, Cam83a, Cam83b, Cam83c] han utilizado esta línea algorítmica para diseñar algoritmos para problemas de Set Partitioning.

El trabajo realizado se enmarca dentro de esta tendencia algorítmica. En los capítulos siguientes se propondrá la utilización de distintas técnicas, así como un algoritmo de tipo híbrido para resolver eficientemente los problemas de set partitioning. Este algoritmo combina métodos heurísticos, desigualdades válidas a partir de formulaciones de los problemas como programas disyuntivos y relajaciones lagrangianas. Además, en ningún momento se utilizará el método del Simplex, eliminandose así los problemas de degeneración masiva que tan a menudo aparecen en este contexto.

CAPÍTULO II

HEURÍSTICAS Y PLANOS SECANTES PARA PROBLEMAS DE SET PARTITIONING

Las heurísticas juegan actualmente un papel muy importante en la resolución de problemas enteros. Ello se debe a que la obtención de soluciones posibles para los problemas proporcionará cotas del valor óptimo de los mismos. Esta información tradicionalmente se ha utilizado en algoritmos de enumeración implícita para eliminar distintos nodos del árbol de exploración. Sin embargo, esta no es la única virtud de estos métodos ya que las soluciones posibles obtenidas pueden, en algunos casos, utilizarse para obtener desigualdades válidas para los problemas.

En cualquier caso, la calidad de las soluciones proporcionadas por las heurísticas resulta fundamental tanto en los procedimientos de enumeración implícita, ya que cuanto más ajustada sea la cota obtenida más nodos se podrán eliminar, como en la generación de desigualdades válidas, ya que a mejor cota más potente será la desigualdad derivada.

En este capítulo se verá que en el caso de los problemas de (SP) es posible obtener ciertas desigualdades válidas para el problema original cuando se disponga de soluciones posibles para los mismos. En ese sentido, para la obtención de dichas desigualdades será fundamental disponer de un procedimiento que proporcione soluciones posibles. Por ese motivo, en las dos primeras secciones se centra la atención en el estudio de procedimientos heurísticos para problemas de set partitioning.

Cabe resaltar que algunas de las diferencias existentes entre los problemas de (SP) y los de (SC) y (SPk) se manifiestan claramente en la dificultad que existe para diseñar heurísticas que obtengan soluciones posibles para (SP). A diferencia de (SC) y (SPk), para los que es inmediato conocer una solución posible "trivial" ($x_j=1$, $\forall j\in N$ en el caso de (SC) y $x_j=1$, $\forall j\in N$ en de (SC)) esto no ocurre en el caso de problemas de (SP). En particular, existen distintas heurísticas que proporcionan soluciones mejores que las ya citadas tanto para (SC) como para (SPk) [Chv79, BaHo80], pero la estructura de los problemas de set partitioning resulta mucho más rígida y dificulta el diseño de este tipo de procedimientos aproximados. Tanto es así, que solamente se conoce una heurística para problemas de (SP) [FiKe86] que únicamente es aplicable a problemas de maximización. Su reciente aparición en 1986 ha supuesto un avance importante en la resolución de estos problemas y su utilización abre nuevas vías en los enfoques algorítmicos a considerar. En la primera sección se expondrá esta heurística.

Resulta cuando menos sorprendente, sin embargo, que el contexto en el que los autores han formulado y utilizado esta heurística sea en el de los problemas de maximización puesto que, como ya se ha visto en el Capítulo I, la inmensa mayoría de aplicaciones de los problemas de (SP) es para problemas de minimización y, no en vano, la formulación que se utiliza en el estudio teórico de estos problemas (I.3) es también de minimización. Hay que considerar, además, que la generalización de la heurística de Fisher y Kedia [FiKe86] para problemas de (SP) de minimización no resulta trivial. Por este motivo, y considerando la importancia que ya se ha señalado de este tipo de procedimientos, en la segunda seción se propone una heurística específica para problemas de minimización.

Posteriormente en la tercera sección se verá como la heurística propuesta se puede utilizar para obtener un tipo particular de desigualdades válidas para problemas de (SP). Finalmente, se propone otra heurística para una clase de problemas en los que el conjunto de restricciones está formado por una familia de igualdades de tipo (SP) y una familia de desigualdades de tipo (SC). La justificación para proponer esta heurística se debe al hecho de que las desigualdades válidas para (SP) que se obtendrán en la Sección 3 son del tipo de (SC) y si éstas se incorporasen al problema original (SP) se obtendría un nuevo problema cuya estructura es la que se acaba de mencionar.

II.1 Heurística de Fisher y Kedia para problemas de maximización de Set Partitioning.

Consideremos ahora el par de problemas

$$
\begin{aligned}
\text{(SP)}\quad & \max\ c\,x \\
& A\,x = e \\
& x_j \in \{0,1\}\ ,\ j \in N
\end{aligned}
$$

y su problema lineal dual asociado

$$
\begin{aligned}
\text{(D)}\quad & \min\ u\,e \\
& u\,A \geq c
\end{aligned}
$$

donde A es una matriz de dimensión mxn, $c \in Z^{n+}$ y e es el vector m-dimensional formado por todo unos.

La heurística que se expone a continuación se caracteriza por buscar un par de soluciones (x,u) posibles para (SP) y (D) respectivamente de forma que el par de soluciones obtenido se acerque, en la medida de lo posible, a satisfacer las condiciones de holgura complementaria (c-uA)x=0.

Por este motivo, la heurística procede en dos fases. En la primera de ellas se construye una solución posible u para el problema lineal dual (D) asociado a (SP). Una vez obtenida esta solución, se calcula el vector s=c-uA de costes reducidos asociados a la misma y se van fijando a 1 algunas de las variables primales de forma secuencial hasta conseguir una solución posible para el problema primal (SP).

La heurística dual propuesta por Fisher y Kedia [FiKe86] en su trabajo tiene en cuenta la estructura específica del problema dual. Considerando que se trata de un problema lineal de set covering generalizado de minimización, en una primera etapa se intentan satisfacer las restricciones de forma secuencial, asignando a todas las variable duales que intervienen en cada restricción la misma cantidad, que es la mínima necesaria para asegurar que dicha restricción se va a satisfacer. Posteriormente se intenta una mejora, buscando variables duales que intervengan únicamente en restricciones que se sobresatisfacen, y disminuyendo el valor de dichas variables en el mínimo de todas las holguras. Finalmente se intenta un intercambio, por el que se disminuye el valor de dos variables a la vez, en una cierta cantidad, a cambio de incrementar el valor de una tercera en la misma cantidad. Este intercambio se itera tantas veces como sea posible.

Siguiendo con la notación introducida anteriormente, sean

$M = \{ 1,2,...,m \}$ conjunto de índices de filas de la matriz A
$N = \{ 1,2,...,n \}$ conjunto de índices de columnas de la matriz A
$M_i = \{ j \in N \,/\, a_{ij}=1 \}$, $\forall i \in M$, $N_j = \{ i \in M \,/\, a_{ij}=1 \}$, $\forall j \in N$.

Se supone en el resto de esta sección que los costes estan ordenados crecientemente:

$$c_j \leq c_{j+1} \quad \forall j \in N$$

La heurística dual es como sigue:

Heuristica dual

Inicializar $u_i=0 \quad \forall i \in M$
Para $j=1,n$ **hacer**

$$\Delta_j = \frac{\max \{ 0, c_j - \sum_{i \in M_j} u_i \}}{/M_j/}$$

$$u_i \leftarrow u_i + \Delta_j , \quad \forall i \in M_j$$

Fin para
Para $i=1,m$ **hacer**

$$u_i \leftarrow u_i + \min \{ 0, \max_{j \in N_i} (c_j - \sum_{k \in M_j} u_k) \}$$

Fin para

Δ_j es el valor mínimo que se debe asignar a cada variable u_i, $i \in M$ para satisfacer la restricción j-ésima del problema dual.

Heurística de mejora

Dada una solución u, posible dual, se intenta mejorar dicha solución, reduciendo el valor de la función objetivo Σu_i identificando 3 índices de variables $i_1, i_2, i_3 \in M$ y modificando el valor de las variables asociada a dichos índices en una determinada cantidad Δ de forma que

$u_{i1} \leftarrow u_{i1} + \Delta$
$u_{i2} \leftarrow u_{i2} - \Delta$
$u_{i3} \leftarrow u_{i3} - \Delta$

con lo que Σu_i quedará reducido en Δ.

Para realizar el intercambio, es necesario dividir el conjunto de restricciones en activas e inactivas :

$$N^a = \{ j \in N / \sum_{i \in M_j} u_i = c_j \} \qquad N^i = \{ j \in N / \sum_{i \in M_j} u_i > c_j \}$$

Además, es fácil comprobar que dicho intercambio será posible si y sólo si se cumplen las dos condiciones siguientes:

1) $N_{i2} \cap N_{i3} \cap N^a = \emptyset$
2) $(N_{i2} \cup N_{i3}) \cap N^a \subseteq N_{i1}$

La condición 1) significa que los índices de variables duales que se elijan deben de ser tales que no existan restricciones activas en las que intervengan simultáneamente las dos variables que se van a decrementar ya que, aunque interviniese la variable que se incrementa en dicha restricción, el valor total de la ecuación disminuirá como poco en Δ con lo que dicha restricción se violaría.

La condición 2) exige que, en aquellas restricciones activas en las que intervenga alguna de las dos variables cuyo valor se decrementa, deberá intervenir también la variable que se incrementa para no violar dichas restricciones.

Una vez determinados los índices i_1, i_2, i_3 que satisfacen las condiciones anteriores, hay que calcular la cantidad Δ de forma que se siga manteniendo la factibilidad. Por lo tanto, el valor Δ vendrá determinado por las restricciones no activas $j \in N^i$ para las que disminuye con el intercambio el valor

$$\sum_{i \in M_j} u_i$$

Para ello se define $N_i^* = \{ j \in N_i / a_{i2j} + a_{i3j} - a_{i1j} > 0 \}$, que es el conjunto de restricciones del problema dual que van a disminuir en Δ con el intercambio.

Si $N_i^*=\emptyset$ entonces el problema dual es no acotado puesto que el intercambio podría realizarse con un valor de Δ arbitrariamente grande sin llegar a violar ninguna de las restricciones. Por lo tanto el problema primal es no factible.

Si $N_i^* \neq \emptyset$ se define

$$\Delta = \min_{j \in N_i^*} \frac{\sum_{i \in M_j} u_i - c_j}{a_{i_2 j} + a_{i_3 j} - a_{i_1 j}}$$

que es el mayor valor que permite hacer el intercambio sin llegar a violar ninguna restricción.

El procedimiento de búsqueda que se realiza para efectuar el intercambio se repite tantas veces como sea posible realizar la mejora. En consecuencia, la heurística de mejora terminará cuando ya no existan los índices i_1, i_2, i_3 para poder realizar el intercambio.

Heurística primal

A continuación se expone la segunda fase de la heurística de Fisher y Kedia [FiKe86]. Se trata de la búsqueda de una solución posible x para (SP) mediante un procedimiento que fija los valores de las variables x_j de forma secuencial manteniendo los requerimientos de factibilidad. El criterio que se sigue es el de incluir en la subsolución parcial que se construye aquella variable de las que no hayan sido consideradas hasta el momento que tenga un mayor coste reducido (menor en valor absoluto). De esta forma se obtendrá una solución posible que se aproxime lo más posible a las condiciones de holgura complementaria. Además, teniendo en cuenta que se trata de un problema de maximización y que interesa que la solución contenga la mayor cantidad de variables posible, la ruptura de empates se realiza eligiendo la variable que interviene en menor cantidad de restricciones.

Los test lógicos expuestos en I.4. juegan un papel muy importante en esta heurística. Su utilización cada vez que una variable primal se fija a 1 hace que el número de variables no consideradas pueda disminuir sensiblemente de una iteración a la siguiente y, además, reduce el riesgo de construir subsoluciones parciales que no puedan completarse en una solución posible, ya que la aplicación de estos tests elimina del subproblema resultante algunas de las variables cuya inclusión en la solución produciría dicha situación.

Asociados a cada solución parcial x que se haya construido, se definen dos conjuntos de índices de filas que particionan el conjunto de restricciones dei problema primal entre las que actualmente se satisfacen y las que no.

Sean $T(x) = \{ i \in M / \sum_{j \in M_i} x_j = 1 \}$ y $U(x) = \{ i \in M / \sum_{j \in M_i} x_j = 0 \}$ dichos conjuntos.

Además, se define el conjunto $J(x) = \{ j \in N / M_j \cap T(x) = \emptyset \}$ de índices de variables que no intervienen en nin una restricción que ya esté previamente satisfecha. La heurística irá seleccionando variables pertenecientes al conjunto J(x) para ir fijándolas a 1.

Dada una solución u posible dual sea s=c-uA el vector de los costes reducidos asociados a dicha variable. El procedimiento es como sigue:

Inicialización

$x=0$	*Todas las variables están a 0.*
$T(x) = \emptyset$	*No existe ninguna fila que esté satisfecha.*
$U(x) = M$	
$J(x) = N$	*Todas las variables son elegibles.*
fin = falso	*El procedimineto terminará cuando se hayan satisfecho todas las restricciones o bien cuando se demuestre que el problema es no factible.*

Fin inicialización

Mientras no fin **hacer** :

Elegir $j \in J(x)$ t.q. $sj = \max_{k \in J(x)} s_k$

En caso de empate elegir j t.q. $|M_j|$ sea mínimo.

Hacer $x_j=1$

Actualizar $T(x) \leftarrow T(x) \cup M_j$

$U(x) \leftarrow U(x) \setminus M_j$

$J(x) \leftarrow J(x) \setminus (\bigcup_{i \in M_j} N_i)$

Aplicar los test lógicos

Si problema no factible **o** $U(x)=\emptyset$ fin= cierto

Fin mientras

Si la heurística no encuentra solución se puede volver a empezar todo el procedimiento eliminando esta vez, en la inicialización, del conjunto J(x) el índice de la variable j que fue la primera en fijarse a 1 en la iteración anterior.

Hay que hacer notar que, aunque esta situación no se produce a menudo en la práctica, existe la posibilidad de que la heurística anterior falle.

Una vez finalizada la segunda fase, los autores del trabajo proponen reducir finalmente el gap de dualidad, en caso que sea necesario, mediante un procedimiento de tipo subgradiente aplicado a la relajación lagrangiana ordinaria del problema (P). En dicho caso, se toma como vector de multiplicadores inicial la solución dual u obtenida en la primera fase de la heurística.

Es de resaltar que, en el procedimiento anterior, la calidad de la solución dual obtenida en la primera fase puede influir decisoriamente en el éxito de la segunda fase de la heurística, que es en la que se calcula la solución primal. En la medida en la que la solución dual obtenida se acerca al óptimo del problema lineal dual, mejora sensiblemente la correspondiente solución primal.

En ese sentido, considerando que la heurística anterior sería igualmente válida si se sustituyese el procedimiento que calcula una solución dual por cualquier otro método válido, el trabajo de Fisher y Kedia [FiKe86] resulta especialmente enriquecedor puesto que fundamentalmente define la filosofía que puede seguir toda una familia de heurísticas para obtener soluciones primales para (SP). Por ese motivo, y a pesar de que Fisher y Kedia no mencionan nada al respecto, resulta sugerente la utilización de la relajación lagrangiana ordinaria (o quizás, alguna relajación algo más sofisticada) en la primera fase de la heurística para obtener así una solución dual cuya calidad esté asegurada. Esta línea que se acaba de mencionar es la que, como se verá posteriormente, se ha seguido en esta tesis.

II.2 Heurística para problemas de minimización de Set Partitioning.

En esta sección consideramos problemas de set partitioning de minimización. A la hora de resolver estos problemas siempre pueden transformarse en problemas de minimización de Set Covering o de Set Packing cuyo conjunto de soluciones óptimas sea el mismo, mediante un cambio de variable adecuado, y, en el caso de que queramos encontrar soluciones posibles, aplicar alguna heurística ad hoc [BeRa71, SaKo73, Chv79, BaHo80, HaHo85]. Ahora bien, cuando se elige la alternativa anterior, la transformación de los problemas hace que se pierda la estructura estructura original de los mismos y que no sólo sea más difícil buscar una interpretación intuitiva al modelo matemático resultante sino que además perdamos toda posibilidad de intentar sacar partido de su estructura original en fases posteriores de su resolución. En el contexto del trabajo que se ha realizado resulta fundamental conservar la estructura de los problemas originales y por ese motivo se propone una heurística específica para los mismos.

Cabe resaltar, una vez más, que la aplicación de la heurística de Fisher y Kedia [FiKe86] expuesta en la sección anterior no resulta trivial para el caso de problemas de minimización. El procedimiento que se propone a continuación, que es específico para estos problemas, se enmarca dentro de toda una familia de heurísticas que podrían diseñarse siguiendo la metodología propuesta en la heurística de Fisher y Kedia [FiKe86] para problemas de maximización. En particular, se ha seguido el criterio de obtener una solución posible para el problema dual y, a partir de ella, construir una solución posible primal que se aproxime en la medida de lo posible a satisfacer las condiciones de holgura complementaria.

Es decir, consta de dos fases, la primera de las cuales proporciona una solución posible para el problema dual lineal asociado al problema primal de minimización (SP). La heurística dual que se propone tiene en cuenta la estructura del problema dual que en este caso resulta ser un problema de multi-knapsack. Asimismo, en la segunda fase se construye una solución posible primal fijando, secuencialmente, variables a 1e intentando alcanzar, en la medida de lo posible, las condiciones de holgura complementaria.

Los problemas que consideramos en esta sección son de la forma:

$$\begin{aligned} \text{(SP)} \ \min\ & cx \\ & Ax = e \\ & x_j \in \{0,1\} \ , j \in N \end{aligned}$$

y su problema dual lineal asociado

(D) max ue
$uA \leq c$

siendo, como antes A una matriz de dimensión mxn, $c \in Z^{n+}$ y e el vector m-dimensional formado por todo unos.

En este caso, los problemas lineales duales que aparecen ya no son problemas de set covering generalizado, sino que se trata de problemas lineales de multi-knapsack lo cual implica que la heurística dual de Fisher y Kedia no es modificable de forma trivial para este tipo de problemas. Por este motivo, se propone una heurística dual para estos problemas que tiene en cuenta la estructura de los mismos. Ahora querremos encontrar una solución dual que, satisfaciendo todas las restricciones, tenga un valor de la función objetivo lo mayor posible.

En todo momento, la heurística que se propone considera dos parámetros asociados a cada variable dual: por un lado, el valor máximo que puede asignarse a la misma sin que se viole ninguna restricción y, por otro lado, el valor máximo que puede asignarse a cada variable dentro de una restricción determinada. Para cada restrición, se asigna a cada una de las variables que intervienen en la misma el mayor valor tal que cumplan las dos condiciones anteriores. Posteriormente, se incrementa en lo posible el valor de aquellas variables que intervengan únicamente en restricciones que se cumplen como desigualdad estricta.

Supondremos en el resto de este capítulo que las variables primales están ordenadas por orden decreciente de costes, es decir $c_j \geq c_{j+1}$ $\forall j \in N$. Utilizando la notación de la sección anterior la heurística dual es la siguiente:

Heurística dual

Inicialización

$u_i = 0 \quad \forall i \in M$

$d_i = \min\limits_{j \in Ni} c_j \quad \forall i \in M$

Definir para cada variable dual u_i cual sería el mayor valor en que se podría incrementar sin violar ninguna de las restricciones, si todas las demás variables se mantuviesen constantes.

Fin Inicialización

Para $j = n, 1, -1$ **hacer**

δ_j indica en cuanto se podría incrementar cada variable que interviene en la restricción j, para satisfacer esa restricción como estricta igualdad.

$$\delta_j = \frac{c_j - \sum_{i \in M_j} u_i}{/M_j/}$$

Para $i \in M_j$ **hacer**

$\Delta_i \leftarrow \min \{ \delta_j , d_i \}$

$u_i \leftarrow u_i + \Delta_i$

Asignar a u_i el mayor valor que nos asegura que no viola ni la restricción que stamos considerando ni ninguna de las posteriores.

Para $s \in N_i$ **hacer**

Actualizar el valor d_s para todas las variables que intervienen en esa restricción.

si $\min\limits_{l \in M_s} \{c_l - \Delta_i\} < d_s$ **entonces** $d_s \leftarrow d_s - \Delta_i$

Fin para

Fin para

Fin para

Para $i=1,m$ **hacer**

Si existe alguna variable que sólamente interviene en restricciones satisfechas como estricta desigualdad i ncrementar su valor manteniendo la factibilidad.

$$u_i \leftarrow u_i + \max \{ 0, \min_{j \in N_i} (c_j - \sum_{k \in M_j} u_k)$$

Fin para

Heurística de mejora

Afortunadamente, después de haber obtenido una solución posible para (D), en el caso de problemas de minimización de (SP) se puede utilizar una versión de la heurística de mejora de Fisher y Kedia [FiKe86] gracias a la que los valores de dos variables duales se incrementan simultáneamente en una cantidad determinada a cambio de reducir el valor de una tercera en la misma cantidad.

Por lo tanto, dada una solución u, posible dual, se intenta aumentar el valor Σu_i de la función objetivo del problema (D) identificando 3 índices de variables $i_1,i_2,i_3 \in M$ y modificando los valores de las variables asociadas a dichos índices en una cantidad Δ

$$u_{i1} \leftarrow u_{i1} - \Delta$$
$$u_{i2} \leftarrow u_{i2} + \Delta$$
$$u_{i3} \leftarrow u_{i3} + \Delta$$

con lo que el valor $\sum u_i$ quedará finalmente incrementado en Δ.

De forma análoga al caso en el que el problema dual es de minimización, es necesario definir dos conjuntos de índices de filas de (D) de particionan el conjunto de restricciones en activas e inactivas.

Sean $N^a = \{ j \in N / \sum_{i \in Mj} u_i = c_j \}$ y $N^i = \{ j \in N / \sum_{i \in M_j} u_i > c_j \}$ dichos conjuntos.

En este caso, las condiciones en las que el intercambio será posible siguen siendo las mismas, es decir decir los índices de variables i_1, i_2, i_3 deberán ser tales que:

1) $N_{i2} \cap N_{i3} \cap N^a = 0$
2) $(N_{i2} \cup N_{i3}) \cap N^a \subseteq N_{i1}$

La interpretación de las condiciones anteriores es análoga al caso en el que el problema (D) es de minimización, teniendo en cuenta que ahora las restricciones son de menor o igual pero que en este caso dos variables se incrementan y una decrementa su valor.

De nuevo, una vez determinados los índices i_1, i_2, i_3 que satisfacen las condiciones anteriores, hay que calcular la cantidad Δ de forma que se siga manteniendo la factibilidad.

El valor Δ vendrá determinado ahora por las restricciones no activas $j \in N^i$ para las que el valor $\sum_{i \in M_j} u_i$ aumenta con el intercambio.

Para ello se define $N_i^* = \{ j \in N^i / a_{i2j} + a_{i3j} - a_{i1j} > 0 \}$, que es el conjunto de índices de restricciones del problema dual que van a aumentar en Δ con el intercambio. Por un razonamiento similar al caso en el que (D) es de minimización se tiene que:

Si $N_i^* = \emptyset$, entonces el problema dual es no acotado y por lo tanto el problema primal es no factible.

Si $N_i^* \neq \emptyset$, se define

$$\Delta = \min_{j \in N_i^*} \frac{c_j - \sum_{i \in M_j} u_i}{a_{i_2 j} + a_{i_3 j} - a_{i_1 j}}$$

que es la cantidad máxima en la que se pueden modificar las variables asociadas a los índices i_1,i_2,i_3 manteniendo todos los requisitos de factibilidad.

La heurística de mejora anterior se repetirá tantas veces como sea posible terminando cuando ya no se encuentren los índices i_1,i_2,i_3 que permitan realizar el intercambio.

Heurística primal

El procedimiento heurístico que se utilizará para obtener una solución primal es, como ya se ha comentado anteriormente, el que se indica en el trabajo de Fisher y Kedia [FiKe86]. Es decir, una vez obtenido el vector s de costes reducidos asociado a la solución dual u que se haya obtenido, se irán fijando variables a 1 de forma secuencial, manteniendo la factibilidad, de forma que el producto escalar de los vectores c-uA y x sea lo menor posible.

Teniendo en cuenta que, en este caso, dada una solución u posible para (D) el vector de costes reducidos asociado c-uA tendrá todas sus componentes no negativas, para acercarnos en la medida de lo posible a las condiciones de holgura complementaria se deberá elegir en cada paso aquella variable, no considerada hasta el momento, cuya componente en dicho vector sea mínima. Además, considerando que el problema (SP) es de minimización, interesa construir una solución posible que tenga el menor número posible de variables fijadas a 1; en consecuencia, se efectuará la ruptura de empates eligiendo aquella variable que intervenga en la mayor cantidad posible de restricciones, es decir tal que $|M_j|$ sea máximo.

Una vez más, el papel que juegan los test lógicos aplicados en cada iteración de la heurística primal es de gran importancia, ya que se acelera el proceso de convergencia al fijar el valor de más de una variable en cada iteración y se reduce el riesgo de construir subsoluciones parciales que no sea posible completar en soluciones posibles.

De nuevo, si una subsolución parcial no logra completarse en una solución posible, el procedimiento se comienza de nuevo eliminando en la reinicialización del conjunto J(x) de variables elegibles la primera que se haya fijado a 1 en la iteración anterior.

El éxito de este procedimiento heurístico, como el de todos aquellos que se definan en el marco de la heurística propuesta por Fisher y Kedia [FiKe86], no está asegurado aunque como veremos posteriormente la experiencia computacional confirma el hecho de que raramente falla.

Por todos los comentarios hechos anteriormente cabe resaltar, una vez más, que la calidad de la solución que se obtenga mediante esta heurística para los problemas de minimización depende de forma considerable de la bondad de la solución dual obtenida. Por este motivo, se ha implementado también una versión de esta heurística en la que la solución dual se obtiene mediante optimización subgradiente aplicada a la relajación lagrangiana ordinaria. Veremos que los resultados obtenidos avalan su utilización ya que no sólo las cotas inferiores que se obtienen son sensiblemente mejores, sino que la solución primal obtenida a partir de esta solución dual es, en muchos casos, mejor que la obtenida a partir de la solución dual obtenida mediante la heurística dual propuesta.

II.3. Planos secantes a partir de cotas condicionales: una aplicación a problemas de Set Partitioning.

En el Capítulo I se han estudiado las distintas alternativas algorítmicas para resolver los problemas de set partitoning y las posibles ventajas e inconvenientes que su elección podría acarrear. Como ya se ha comentado en dicho capítulo, debido a las dificultades que pueden surgir en la obtención de la tabla óptima del Simplex para la relajación lineal ordinaria de (SP), una de las alternativas que resulta más sugerente es la obtención de desigualdades válidas para los problemas a partir de condiciones lógicas implicadas por los mismos.

En esta sección se expone un caso particular de este tipo de desigualdades para problemas enteros o mixtos de programación matemática obtenidas a partir de disyunciones válidas. Posteriormente, se estudia y se propone su aplicación para problemas de set partitioning.

Estas disyunciones fueron introducidas por Balas [Bal80] en un trabajo en el que se estudia su aplicación al caso de problemas de set covering y utilizadas por Balas y Ho [BaHo80] en una amplia experiencia computiacional para ese tipo de problemas. Posteriormente se ha estudiado su aplicación al caso de problemas de (TSP) en un trabajo de Balas y Christofides [BaCh81].

Las disyunciones que se obtienen pueden utilizarse en dos contextos algorítmicos diferentes: por un lado, al proporcionar una partición del conjunto de soluciones posibles, pueden utilizarse en procedimientos de enumeración implícita y, por otro lado, al facilitar la obtención de unas ciertas desigualdades, éstas pueden emplearse en procedimientos análogos a los de planos secante.

Las desigualdades obtenidas a partir de estas disyunciones no son desigualdades válidas para los problemas originales, ya que, de hecho, se construyen a partir de soluciones posibles para las que son violadas. Se trata, sin embargo, de planos secantes para los problemas, no en el sentido clásico del término, puesto que no se obtienen a partir de soluciones fraccionales, pero sí en el sentido de que limitan el espacio de búsqueda de soluciones posibles para el problema original, ya que, como se acaba de mencionar, se generan a partir de soluciones posibles previamente obtenidas que no las cumplen y, además, deben satisfacerse por cualquier solución mejor que la incumbente, en el caso de que esta exista.

La teoría básica desarrollada por Balas en su trabajo [Bal80] es, en principio, aplicable a todo tipo de problemas enteros y mixtos pero la utilización, en el diseño de algoritmos concretos, de los resultados que allí se obtienen no resulta sencilla para el caso general. En el caso de los problemas de set covering (SC), la aplicación de dichos resultados resulta especialmente adecuada ya que la estructura de los problemas permite generar de forma práctica las desigualdades que, además, son del mismo tipo que las restricciones del problema original [Bal80,BaHo80].

En esta sección veremos que, en el caso de los problemas de Set Partitioning, se cumplen las condiciones necesarias para la obtención de este tipo de desigualdades válidas y se propone la utilización de un procedimiento para generarlas de forma que los planos secantes obtenidos puedan utilizarse en el diseño de algoritmos específicos para resolver los problemas de (SP).

II.3.1. Disyunciones a partir de cotas condicionales.

A continuación se hace un resumen del trabajo de Balas [Bal80] en el que se definen este tipo de desigualdades. Además, puesto que estamos interesados en la obtención práctica de estas desigualdades para los problemas de (SP) se irá estudiando la aplicación de los mismos en el diseño de un procedimiento específico para este tipo de problemas.

Teorema 1: Sean $\pi \in R^{n+}$, $\pi_0 \in R^+$, $N=\{1,...,n\}$, $Q_i \subseteq N$, $i=1,...,p$, $1 \leq p \leq n$.
Existe $v \in R^{p+}$ t.q

$$\sum_{i/j \in Q_i} v_i \leq \pi_j , \quad j \in N \ \text{ y } \ \sum_{i=1}^{p} v_i > \pi_0$$

si y sólo si cualquier entero $x \in N^{n+}$, tal que satisface $\pi x \leq \pi_0$ también satisface la disyunción

$$\bigvee_{i=1}^{p} (x_j = 0 , \ j \in Q_i)$$

La demostración de este teorema se basa en la aplicación del teorema de la alternativa de Farkas al par de sistemas

$$\begin{array}{llcll} \pi x \leq \pi_0 & & & ev > \pi_0 & \\ Tx \leq e & (I) & y & Tv \leq p & (II) \\ x \geq 0 & & & v \geq 0 & \end{array}$$

siendo T la matriz de dimensión pxn definida por

$$t_{ij} = \begin{cases} 1, j\in Q_i \\ 0, j\in N\backslash Q_i \end{cases} \qquad i=1,...,p$$

El teorema anterior sigue siendo cierto si se sustituyen las condiciones

$$\pi x \leq \pi_0 \;\text{ por }\; \pi x < \pi_0 \;\text{ y}$$

$$\sum_{i=1}^{p} v_i > \pi_0 \;\text{ por }\; \sum_{i=1}^{p} vi \geq \pi_0,$$

ya que en este caso también se podría aplicar el teorema de la alternativa de Farkas.

El Teorema 1, que resulta básico para los resultados posteriores, proporciona un método para obtener desigualdades "fuertes" de la forma $\pi x \leq \pi_0$ (o $\pi x < \pi_0$) y poder deducir a partir de ellas disyunciones suficientemente potentes.

Sea el siguiente problema entero mixto :

(P) $\quad \min \{ cx \,/\, A\geq b, x\geq 0, x_j \text{ entero}, j\in N_1 \subset N \}$

y sean z_{sup} una cota superior conocida del problema (P), y u y s dos vectores tales que $u\geq 0$, $s=c-uA\geq 0$.

Para cualquier solución posible x se tiene que $Ax\geq b \Rightarrow -uAx \leq -ub$. Sumando esta expresión a la desigualdad $cx < z_{sup}$, que deberá satisfacer cualquier solución posible x para (P) cuyo valor de la función objetivo sea mejor que la cota superior z_{sup}, se tiene que $(c-uA)x < z_{sup} - ub$.

Por lo tanto $sx < z_{sup}-ub$ es una desigualdad que debera satisfacerse por cualquier solución posible para (P) tal que $cx < z_{sup}$.

Caso particular (SP): Sea el problema de Set Partitioning

(SP) $\quad \min \{ cx \,/\, Ax=e, x\geq 0, x_j\in \{0,1\}, j\in N \};$

su problema dual lineal asociado es (DP) $\max \{ ue \,/\, uA\leq c\}$.

Sean z_{sup} una cota superior conocida del problema (SP), y u y s dos vectores tales que u sea una solución posible de (DP) y $s = c-uA \geq 0$.

A pesar de que en este caso el vector u no está restringido en signo, sigue siendo cierto que $Ax=e \Rightarrow -uAx = -ue$.

Por lo tanto sumando la igualdad anterior a la desigualdad $cx < z_{sup}$ se tiene que $(c-uA)x < z_{sup} - ue$.

En consecuencia, $sx < z_{sup} - ue$ es una desigualdad que deberá ser satisfecha por cualquier solución posible x para (SP) tal que $cx < z_{sup}$.

Volviendo al caso general, utilizando la desigualdad $(c-uA)x < z_{sup} - ub$ como $\pi x < \pi_0$ y definiendo $v_i = s_{j(i)}$, $i=1,...,p$ para algunos índices $j(i) \in N$, se puede aplicar el Teorema 1. De este modo, se obtiene el siguiente

Corolario 1: Sea z_{sup} una cota superior del valor de (P) y dos vectores u,s tales que $u \geq 0$, $s=c-uA \geq 0$.

Si existe $S \subset N_1$, $S=\{j(1),...,j(p)\}$, $1 \leq p \leq |N_1|$, tal que $\sum_{j \in S} s_j \geq z_{sup} - ub$,

entonces para cualquier colección de conjuntos $Q_i \subset N$, $i=1,...,p$ tales que $\sum_{i/j \in Q_i} s_j(i) \leq s_j$, $j \in N$, se cumple que cualquier solución x posible para (P) tal que $cx < z_{sup}$ debe satisfacer la disyunción $\bigvee_{i=1}^{p} (x_j=0, j \in Q_i)$ (1).

El corolario anterior sigue siendo cierto para el caso de problemas de Set Partitioning ya que, aunque el vector u en este caso no esta restringido en signo, se siguen cumpliendo las hipótesis necesarias para la aplicación del Teorema 1 como se ha visto anteriormente.

Es importante resaltar que para la aplicación del Corolario 1 será necesario disponer de una cota superior para el problema y de una solución posible para el mismo. Si no disponemos de un procedimiento heurístico que proporcione ambos elementos, la aplicación de este corolario no será posible a no ser que se esté utilizando un método de enumeración y ya se haya profundizado suficientemente en la exploración como para obtenerlos; lo anterior siempre requerirá un esfuerzo computacional considerable. Por consiguiente, la heurística propuesta en la sección anterior se convierte en una herramienta de gran importancia, que permite generar las disyunciones del tipo (1) para los problemas de (SP), ya que nos proporcionará tanto la solución primal x, y por lo tanto una cota superior para el problema, como el vector u con un esfuerzo computacional reducido. Posteriormente, veremos que también es posible satisfacer la condición adicional para la aplicación del corolario de que la suma de ciertos costes reducidos sea mayor o igual que el gap de dualidad que se tenga.

Considerando de nuevo el caso general de un problema mixto (P), un caso particular muy interesante del Corolario 1 ocurre cuando la disyunción generada consta de un único elemento puesto que ésta se convierte en la condición $x_j= 0, j\in Q_1$. Es decir, definiendo el conjunto $Q_1=\{ j\in N_1 / s_j \geq z_{sup} -ub \}$, cualquier solución x, posible para (P), tal que $cx < z_{sup}$ debe satisfacer $x_j= 0, j\in Q_1$.

La importancia de este caso particular radica en las aplicaciones prácticas, ya que permitirá fijar a 0 y, por lo tanto, eliminar del problema original todas aquellas variables que tengan un coste reducido asociado mayor o igual que al gap entre la cota superior z_{sup} y el valor de la función objetivo asociado a la solución u posible para el problema dual lineal, a partir de la cual se ha generado el vector s de costes reducidos.

Evidentemente, no siempre es fácil conseguir que existan variables que cumplan esta condición, ya que para ello es necesario que el gap antes mencionado sea relativamente reducido lo cual implica que la calidad de las soluciones primal y dual respectivamente sea suficientemente buena, propiedad que, desgraciadamente, no siempre se consigue.

El Teorema 1 y el Corolario 1 tienen una interpretación en términos de cotas condicionales que resulta bastante clarificadora:

Sea el par de programas lineales duales

(L) $\min \{ cx / Ax \geq b, x\geq 0 \}$ y (D) $\max \{ ub / uA \leq c, u\geq 0 \}$.

Cualquier solución u posible para (D) nos proporciona una cota inferior ub del valor de (L) y por lo tanto del valor de (P). Ahora bien, si se modificase el conjunto de restricciones de (P) añadiendo el sistema de desigualdades $Tx \geq e$, los programas (L) y (D) se convertirían en

(LT) $\min \{ cx / A x\geq b, Tx \geq e, x\geq 0 \}$ y (DT) $\max \{ ub+ve / uA+vT \leq c, u\geq 0, v\geq 0 \}$,

con lo que ub+ve sería una cota inferior del valor de (LT) y por lo tanto del de (PT) (problema obtenido a partir de (P) anadiéndole las restricciones $Tx \geq e$).

Ahora bien, si se encontrase un vector v que, junto con el vector u, fuese posible para (DT) y tal que $ub + ve \geq z_{sup}$, se tendría que cualquier solución posible para (LT) (y por lo tanto para (PT)) satisfaría $cx > z_{sup}$ ya que $z_{sup} \leq ub + ve \leq cx$.

Es decir cualquier solución x posible para (P) tal que $cx < z_{sup}$ debe violar el conjunto de restricciones $Tx \geq e$.

Cabe resaltar que el conjunto de restricciones $Tx \geq e$ no forma parte del problema original, y el único motivo para incorporarlas al problema es para poder asegurar que si se satisficiesen se obtendría una cota inferior que sería, por lo menos, tan grande como la cota superior z_{sup}. Por lo tanto, cualquier solución mejor que la que ha proporcionado la cota z_{sup}, en el caso de que ésta exista, deberá violar por lo menos una de dichas desigualdades.

Hay que tener en cuenta que la afirmación recíproca de la anterior no es cierta en el sentido de que pueden existir soluciones posibles para (P) que violen alguna de las desigualdades $Tx \geq e$ y que no proporcionen valores de la función objetivo mejores que el de la cota superior z_{sup}.

El Corolario 1 proporciona una herramienta para obtener una disyunción de la forma (1)

$\bigvee_{i=1}^{p} (x_j=0, i \in Q_i)$ que sea válida para el problema (P).

Para ello sólo se necesita disponer de una cota superior z_{sup}, una solución u posible para el problema dual lineal (D) y un subconjunto $S \subseteq N_1$ para el que se cumpla la condición $\sum_{j \in S} s_j \geq z_{sup}$ -ub (posteriormente se verá que esta condición es fácil de satisfacer).

Dado un conjunto S, cualquier familia de subconjuntos Q_i de N_1 que cumpla $\sum_{i/j \in Q_i} s_j(i) \geq s_j$, $j \in N$, proporcionará una disyunción válida del tipo (1).

La cuestión que se plantea es la de obtener la familia de subconjuntos Q_i que délugar a la disyunción más potente posible, es decir aquella que tenga el menor número de términos (p lo más pequeño posible), siendo cada término lo mayor posible (los conjuntos Q_i, i=1,p lo más grandes posible).

Heurística para generar disyunciones válidas para (P)

A continuación se expone una heurística que genera disyunciones válidas del tipo (1) teniendo en cuenta los objetivos que se acaban de mencionar.

Elegir un subconjunto $S \subset N_1$ minimal tal que

$$\sum_{j \in S} s_j \geq z_{sup} - ub$$

Ordenar $S=\{ j(1),...,j(p) \}$ según los valores decrecientes de $s_j(i)$.

Hacer $Q_1=\{ j \in N_1 / s_j \geq s_{j(1)} \}$

Para i=2,...,p **Definir recursivamente**

$$Q_i = \left\{ j \in N / s_j \geq s_{j(i)} + \sum_{k=1}^{i-1} s_{j(k)} . t_{kj} \right\},$$

siendo $t_{kj}=1$ si $j \in Q_k$, y 0 si $j \notin Q_k$.

Fin para

Los conjuntos Q_i i=1,...,p así definidos dan lugar a una disyunción válida por construcción.

Las disyunciones generadas por la heurística anterior pueden utilizarse en procedimientos exactos de tipo branch and bound. En este caso, éstas pueden refozarse de forma que definan una partición del espacio de búsqueda.

En concreto, $\bigvee_{i=1}^{p} (x_j=0, j \in Q_i)$ puede sustituirse por

$$\bigvee_{i=1}^{p} \left(x_j=0,\ j \in Q_i,\ \sum_{j \in Q_k} x_j \geq 1,\ k=1,...,i-1 \right)$$

Considerando que, por la forma en la que se han definido los conjuntos Q_i, puede asegurarse que $s_j \geq s_{j(i)} > 0$ para $j \in Q_i$, i=1,...,p, en todas las ramas, excepto en la correspondiente a i=1, la cota inferior ub puede reforzarse inmediatamente después de ramificar, asociando a cada desigualdad $\sum_{j \in Q_k} x_j \geq 1$, el multiplicador positivo $s_{j(k)}$.

Es decir, en la i-ésima rama (i>1) la cota inferior ub puede sustituirse por $ub+s_{j(1)}+...+s_{j(i-1)}$.

II.3.2.Los planos secantes

La aplicación de las disyunciones de tipo (1) a procedimientos de tipo branch and bound no resulta ser la única para resolver los problemas mixtos (P) (y en particular los de (SP)). En concreto, en este apartado veremos cómo a partir de dichas disyunciones, pueden otenerse desigualdades válidas, en el sentido que ya se ha explicado anteriormente, para el problema (P). Veremos que estas desigualdades pueden también utilizarse en el contexto de los problemas de set partitioning. Esta utilidad resulta muy interesante ya que permite reducir sensiblemente el espacio de búsqueda de soluciones óptimas para el problema original.

Como ya se ha comentado anteriormente, las desigualdades que se obtengan serán planos secantes (cutting planes), no ya en el sentido clásico del término, puesto que no las obtendremos a partir de soluciones fraccionales del problema lineal (L), pero sí en el sentido de que serán violadas por las soluciones posibles a partir de las cuales se generarán y además se podrá asegurar que deberán satisfacerse por cualquier solución mejor que la que haya proporcionado la cota superior z_{sup}. Podrán también considerarse como desigualdades válidas para el conjunto de soluciones mejores que la que haya proporcionado la cota superior z_{sup} del problema mixto (P) ya que todas ellas (si es que existe alguna) deberán estar contenidas en el semiespacio que definan estas desigualdades.

Las desigualdades que se obtienen, mediante este procedimiento, a partir de las disyunciones del tipo (1) son de la forma $\sum_{j \in W} x_j \geq 1$.

Debido a la estructura de estas desigualdades, la utilización de las mismas puede resultar más sugerente en la resolución de problemas de Set Covering ya que las desigualdades resultan ser del mismo tipo que las restricciones de los problemas. Lo anterior permite, en el caso de problemas (SC), incorporarlas a los mismos obteniendo un problema ampliado cuya estructura resulta ser la misma que la del problema original.

A pesar de que los resultados teóricos en los que se basa la generación de estas desigualdades son ciertos para problemas enteros y mixtos en general, a menudo, la estructura de los mismos (o la carencia de una estructura particular) hará que no sea posible la utilización párctica de estos resultados en el diseño de heurísticas, ya que no siempre se conseguirán las condiciones concretas que faciliten dicha aplicación.

Esto no va a ser así en el caso de los problemas de Set Partitioning, para los que veremos que su estructura permite utilizar los resultados teóricos en el diseño de una heurística que permita la obtención práctica de estos planos secantes. Además, veremos que el cambio que resultará en la estructura original del problema, al incorporar las desigualdades obtenidas al problema original de (SP), no será obstáculo para el diseño de algoritmos que lleven a la resolución de los mismos.

En el resto de esta sección nos limitaremos a considerar problemas de Set Partitioning y los problemas ampliados que se obtengan al incorporar a los mismos las desigualdades que se generen.

Sea, por tanto, (SP) min $\{ cx \,/ Ax=e, x\geq 0, x_j\in \{0,1\}, j\in N \}$

Consideremos el i-ésimo término de una disyunción (1), $x_j=0$, $j\in Q_i$. Evidentemente, cualquier solución posible x para (SP) que satisfaga este término también deberá satisfacer las desigualdades

$$\sum_{j\in N_h\setminus Q_i} x_j \geq 1,\ h\in M \qquad (2)$$

y, por lo tanto, para cualquier elección de los índices $h(i)\in M$, $i=1,...,p$, cualquier solución posible x que satisfaga la disyunción

(1) $\bigvee_{i=1}^{p} (x_j=0, j\in Q_i)$, deberá también cumplir la disyunción

$\bigvee_{i=1}^{p} \left(\sum_{j\in N_{h(i)}\setminus Q_i} x_j \geq 1, j\in Q_i \right)$ lo cual implica que,

para x entero, se debe también satisfacer la desigualdad $\sum_{j\in W} x_j \geq 1$,

siendo $W = \bigcup_{i=1}^{p} (N_{h(i)}\setminus Q_i)$

Expresando en estos términos el Terorema 1 se obtiene el siguiente

Corolario 2: Sean z_{sup} una cota superior de (SP) y u,s tales que $s = c-uA \geq 0$. Si existe un conjunto de índices de columnas $S= \{ j(1),...,j(p) \}$, $0\neq S\subseteq N$ tal que

$$\sum_{j\in S} s_j \geq zsup - ue,$$

entonces, para cualquier conjunto de p índices de filas $h(i)\in M$, $i=1,...,p$, y para cualquier colección de conjuntos $Q_i\subseteq N$ tal que

$$\sum_{i/j \in Q_i} s_{j(i)} \leq s_j \ , \ \ j \in N,$$

se tiene que cualquier solución posible x para (SP) tal que $cx < z_{sup}$ satisface la desigualdad

$$\sum_{j \in W} x_j \geq 1, \quad \text{siendo } W = \bigcup_{i=1}^{p} (N_{h(i)} \setminus Q_i) \qquad (3)$$

En el Corolario 2 la condición $Q_i \subseteq N$ se puede sustituir por $Q_i \subseteq N_{h(i)}$ sin que ello afecte al conjunto W obtenido.

Como ya se ha expuesto repetidas veces, para la obtención de disyunciones válidas y, por lo tanto, para la obtención de los planos secantes del tipo (3) es necesario disponer de un conjunto S de índices de columnas para el que se verifique la condición

$$\sum_{j \in S} s_j \geq zsup - ue.$$

Interesa, por consiguiente, saber bajo qué condiciones se cumple la desigualdad enterior. Veremos que la respuesta a esta cuestión viene dada en términos de condiciones adicionales para el vector u.

<u>Teorema 2</u>: Sean u, s dos vectores que satisfacen $s = c\text{-}uA \geq 0$, y sean x una solución posible para (SP) y $S(x)=\{ j \in N \ / \ x_j=1 \}$ el soporte de la solución x. Entonces,

$$u(Ax\text{-}e) = 0 \ \Rightarrow \ \sum_{j \in S} s_j \geq zsup - ue \ \text{ para } S=S(x).$$

<u>Demostración</u>: Sea el par de programas lineales duales

(L1) $\min \{ cx \ / \ Ax=e, \ x_j \geq 1, j \in S(x), \ x_j \geq 0, j \in N \setminus S(x) \}$ y
(D1) $\max \{ ue + \sum_{j \in S(x)} s_j \ / \ uA+s = c, j \in N, s \geq 0 \}$

Evidentemente, x es una solución posible para (L1) y el par (u,s) lo es a su vez para (D1). Además x y (u,s) cumplen las condiciones de holgura complementaria para los problemas (L1) y (D1) puesto que

$$u(Ax\text{-}e)=0$$
$$s_j(xj\text{-}1)=0, \ j \in S$$
$$s_j x_j=0, \ j \in N \setminus S.$$

Por lo tanto, se trata de soluciones óptimas para los problemas (L1) y (D1) respectivamente y se cumple que

$$ue + \sum_{j \in S(x)} s_j = cx \,.$$

Como $cx \leq z_{sup}$ se tiene que $\sum_{j \in S(x)} s_j \geq z_{sup} - ue$ (4) .♦

Este teorema resulta especialmente afortunado en el caso de problemas de Set Partitioning ya que nos asegura que siempre se va a satisfacer la condición (4), puesto que para cualquier solución x posible para (SP) siempre se deben cumplir las restricciones $Ax-e=0$ y, por tanto, la condición $u(Ax-e)=0$, independientemente de la solución lineal dual u que tengamos.

Todo lo anterior nos permite afirmar que, dada una solución posible x, siempre vamos a estar en una situación favorable a la hora de generar desigualdades del tipo (3).

Una versión más general del Teorema 2 se puede demostrar para problemas de Set Covering. Además, en este caso, es fácil ver que siempre se puede conseguir cumplir la condición $u(Ax-e)$ fijando a 0 todas las componentes del vector u asociadas a restricciones del problema primal que no se satisfagan como estricta igualdad. En contrapartida, se debilitará el valor de la cota inferior asociada a la nueva solución dual que se obtenga.

Dado un problema (SP), una vez obtenida una desigualdad válida (4) para el mismo, ésta puede incorporarse al conjunto de restricciones del problema original (SP) obteniendo así un nuevo problema. Teniendo en cuenta que las desigualdades (4) son siempre del tipo "$\pi x \geq 1$", la estructura del problema ampliado que se obtenga ya no será la de un problema de (SP) sino que su conjunto de restricciones constará de ecuaciones de dos tipos: unas del tipo $Ax=e$ (las del problema original), y otras del tipo $Gx \geq e$ (las correspondientes a los planos secantes obtenidos que se han incorporado).

El Teorema 2 también permite asegurar que se pueden generar desigualdades válidas del tipo (3) a problemas de la forma

$$\text{(SPA)} \quad \min \{ cx \,/\, Ax=e,\ Gx \geq e,\ x_j \in \{0,1\},\ j \in N \}$$

una vez que se disponga de una solución posible primal para (SPA) y de una solución posible para el problema lineal dual asociado. Para ello tendremos tan sólo que reducir, tal vez, el valor de algunas variables duales asociadas a las restricciones de la forma $Gx \geq e$.

El comentario anterior pone de manifiesto la necesidad de disponer también de soluciones posibles para los problemas (SPA). Hay que resaltar que para estos problemas no son válidas ni las heurísticas para problemas del tipo (SP), ni aquellas para los de la familia (SC). Posteriormente veremos que es posible obtener soluciones posibles para este tipo de problemas mediante un procedimiento heurístico que se propondrá en la próxima sección.

A continuación se exponen algunas de las propiedades teóricas de las desigualdades de tipo (4).

Definición 1: Una desigualdad de la forma $\sum_{j\in W} x_j \geq 1$ se dice que es *nueva* si no existe $i\in M$ tal que $N_i \subset W$.

Propiedad: Si $\sum_{j\in W} x_j \geq 1$ viola una solución posible x^* para (SP) entonces $\sum_{j\in W} x_j \geq 1$ es nueva.

La propiedad anterior se deduce del hecho que si $\sum_{j\in W} x_j \geq 1$ viola una solución posible x^*, $\Rightarrow \sum_{j\in W} x_j^* = 0 \Leftrightarrow S(x^*) \cap W = 0$, siendo $S(x^*) = \{ j\in N \,/\, x_j^*=1 \}$ el soporte de la solución x^*.

Además, como x^* es solución posible $\Rightarrow \forall\, i\in M\ \exists\, j\in N_i$ t.q. $j\in S(x^*) \Rightarrow \forall\, i\in M\ \exists\, j\in N_i$ t.q. $j\notin W \Rightarrow Ni \not\subset W \Rightarrow$ la desigualdad es nueva.

Consideremos ahora el poliedro

$P = \text{conv}\ \{ x\in R^n \,/\, Ax\geq e,\ x\geq 0,\ x_j \text{ entero}, j\in N \}$.

Teorema 3: Una desigualdad de la forma $\sum_{j\in N_i} x_j \geq 1$, $i\in M$ define una faceta de P si y sólo si no existe $k\in M$ tal que $N_k \subset N_i$, $N_k \neq N_i$.

Es decir, una desigualdad $\sum_{j\in W} x_j \geq 1$, define una faceta de P si y sólo si es nueva.

Hay que resaltar que el poliedro P es no acotado y por lo tanto las desigualdades del tipo (3), a pesar del interés teórico del resultado anterior, no resultan adecuadas para el estudio de los polítopos de los problemas tanto de Set Covering como de Set Partitioning.

El siguiente teorema nos proporciona las condiciones que deben satisfacer los índices de columnas j(i) y los índices de filas h(i) para garantizar que las desigualdades obtenidas violan una determinada solución. Por lo tanto, el siguiente teorema proporciona una herramienta para garantizar que las desigualdades que se obtengan sean planos secantes que reduzcan el espacio de búsqueda de soluciones para el problema original (SP).

Dada una solución posible x sea $T(x)=\{ i\in M \,/\, a^i x=1\}$, el conjunto de índices asociado a las restricciones satisfechas como estricta igualdad por la solución x.

Teorema 4: Sean z_{sup}, u y s tales que $s = c-uA \geq 0$, $S = \{ j(1),...,j(p) \}$, $\emptyset \neq S \subseteq N$ tal que $\sum_{j\in S} s_j \geq z_{sup} - ue$, $Q_i \subset N$, $i=1,...,p$ tales que $\sum_{i/j\in Q_i} s_{j(i)} \leq s_j$, $j\in N$, y sea $j(i)\in Q_i$, $i=1,...,p$.

Si x^* es una solución posible para (SP) tal que $S\subseteq S(x^*)$ y $h(i)\in T(x^*)\cap M_{j(i)}$, $i=1,...,p$, entonces la desigualdad

$$\sum_{j\in W} x_j \geq 1, \quad \text{siendo } W = \bigcup_{i=1}^{p} \left(N_{h(i)} \setminus Q_i \right),$$

es violada por x^*.

Demostración:

Sea $h(i) \in T(x^*) \cap M_{j(i)}$, $i=1,...,p$

$$h(i) \in M_{j(i)} \Rightarrow j(i) \in N_{h(i)}, \; i=1,...p$$

Además,

$$h(i) \in T(x^*) \Rightarrow | S(x^*) \cap N_{h(i)} | = 1, \; i=1,...,p.$$

Por otro lado,

$$j(i) \in S \subseteq S(x^*) \Rightarrow x_{j(i)}{}^* = 1 \Rightarrow S(x^*) \cap N_{h(i)} = \{j(i)\}, \; i=1,...,p.$$

Además como

$$j(i) \in Q_i, \; i=1,...,p \Rightarrow S(x^*) \cap (N_{h(i)} \setminus Q_i) = \emptyset, \; i=1,...,p \Rightarrow S(x^*) \cap W = \emptyset$$

y por lo tanto la desigualdad es violada por x^*. ♦

Observación:

Cabe resaltar que el Teorema 4 puede aplicarse tanto al caso de problemas de Set Covering como de Set Partitioning. Ello es así puesto que su demostración se centra en el hecho de que para las restricciones que se satisfacen como estricta igualdad existe un único elemento de las mismas que está en el soporte de la solución y además dicho elemento no pertenece al conjunto W por construcción del mismo. Esta propiedad, que es común a los problemas de (SC) y de (SP), pone de manifiesto la semejanza parcial entre estos dos tipos de problemas. Permite además, asegurar que la desigualdad obtenida es un plano secante en el sentido de que viola la solución posible a partir de la que se ha obtenido, característica que justifica la calidad de la desigualdad obtenida.

En el caso de problemas de Set Covering, siempre debe existir al menos una restricción que se satisfaga como estricta igualdad para una solución dada, ya que en caso contrario, dicha solución sería redundante puesto que todas las restricciones estarían sobresatisfechas y podríamos eliminar del soporte de la solución por lo menos un índice de variable.

Por el contrario para los problemas de Set Partitioning el conjunto T(x) coincide con el conjunto total de los índices de filas, ya que cualquier solución, para ser posible, debe satisfacer todas las restricciones como estricta igualdad.

La validez del Teorema 4 para problemas (SP) permite, desde una pespectiva algorítmica, diseñar un procedimiento para obtener desigualdades válidas para este tipo de problemas a partir de un par (x,u) de soluciones posibles para los problemas primal y dual respectivamente.

Además, el Teorema 4 también podrá aplicarse a problemas del tipo (SPA) ya que, por la estructura que presenta este tipo de problemas (familia de restricciones de tipo (SP) y familia de restricciones de tipo (SC)), se cumple la condición en la que se centra la demostración del mismo debido a que dicha concición es cierta, como se acaba de demostrar, tanto para las restricciones de tipo (SP) como para las restricciones de tipo (SC). En consecuencia, este teorema sugiere utilizar un procedimiento iterarivo para resolver los problemas de (SP) en el que se generen desigualdades válidas para (SP) y se incorporen las mismas al conjunto original de restricciones. De esta forma, se obtendrán problemas ampliados del tipo (SPA) para los que, como consecuencia de la aplicación del Teorema 4, también se podrán derivar de una forma práctica estos planos secantes.

A pesar de que ahora ya se dispone de una herramienta para generar planos secantes válidos que además son todos distintos, las desigualdades generadas por este procedimiento no serán todas igualmente "potentes". Teniendo en cuenta que todos los coeficientes de estas desigualdades son 0 ó 1 y el término independiente es siempre 1 podemos afirmar que una desigualdad de este tipo será tanto más potente en la medida que tenga menos coeficientes iguales a 1. Por este motivo, intentaremos que el conjunto

$$\bigcup_{i=1}^{p} (N_{h(i)} \setminus Q_i)$$

sea lo menor posible.

Para este fin, podemos tomar $S=\{j(1),...,j(p)\}$ como el conjunto de los índices asociados a los p mayores costes reducidos, siendo p el menor número entero para el que se cumple la condición

$$\sum_{i=1}^{p} s_{j(i)} \geq z_{sup} - ue.$$

Además, elegiremos los índices $h(i) \in M_{j(i)}$ intentando que en cada paso se minimice el tamaño del conjunto $W_k \setminus W_{k-1}$, siendo

$$W_k = \bigcup_{i=1}^{k} (N_{h(i)} \setminus Q_i).$$

A continuación se expone un procedimiento para la generación de estos planos secantes que es válido tanto para problemas de Set Covering, como para problemas de Set Partitioning así como para los problemas del tipo (SPA) que tienen parte de sus restricciones del tipo de (SP) y las restantes del tipo de (SC). Este procedimiento está basado en la aplicación del Teorema 4 y tiene en cuenta los comentarios hechos anteriormente.

Procedimiento de generación de cortes

Sean una solución posible x de soporte S(X) y dos vectores u,s tales que $s=c-uA \geq 0$.

Sea $S^{+}=\{j \in S(x) \ / \ s_j>0 \}$ el conjunto de índices de variables cuyos costes reducidos asociados son estrictamente positivos. El procedimiento es como sigue:

Inicialización

$W=\emptyset$

$S=S^{+}$

$z_l=ue$

Fin Inicialización

Mientras $z_l < z_{sup}$ **hacer**

definir $s_{j(i)} = \max_{j \in S} s_j$

$Q = \{ j \in N \,/\, s_j \geq s_{j(i)} \}$

elegir $h(i)$ tal que $| (N_{h(i)} \backslash Q) \cup W | = \min_{h \in M_{j(i)}} | (N_h \backslash Q) \cup W |$

hacer $W \leftarrow W \cup (N_{h(i)} \backslash Q)$

$z_l \leftarrow z_l + s_{j(i)}$

si $z_l \geq z_{sup}$ **entonces**

$Q_i = Q \cap N_{h(i)}$

$s_j \leftarrow s_j - s_{j(i)} \;\forall\, j \in Q_i$

$S \leftarrow S \setminus \{j(i)\}$

Fin si

Fin mientras

Añadir la desigualdad

$$\sum_{j \in W} x_j \geq 1 \,.$$

Fin

II.4. Heurística para problemas de minimización de Set Partitioning ampliados con restricciones de tipo Set Covering

Consideremos ahora problemas de la forma

$$
\text{(SPA)} \quad \begin{aligned}
&\min\ cx \\
&Ax = e \\
&Gx \geq e \\
&\pi x \geq 1 \\
&x_j \in \{\ 0,1\ \},\ j \in N
\end{aligned}
$$

siendo $c \in Z^{n+}$, A y G matrices de dimensiones mxn y r-1xn, respectivamente, cuyos elementos son 0 ó 1, e_m y e_{r-1} los vectores m-dimensional y r-1-dimensional, respectivamente, formados por todo unos y π un vector n-dimensional cuyas componentes son 0 ó 1.

En la sección anterior, se han obtenido unas desigualdades válidas para los problemas (SP) a partir de cotas condicionales que son de la forma $\pi x \geq 1$. Además, un procedimiento que utilice dichas desigualdades para problemas de set partitioning puros y que incorpore las mismas al problema original irá generando nuevos problemas cuya estructura será la de (SPA).

Por lo tanto, teniendo en cuenta que las desigualdades que se obtengan serán valiosas en la medida que éstas nos proporcionen información adicional sobre el problema original, y que la única forma de obtener dicha información es resolviendo el problema resultante de incorporar la desigualdad obtenida al problema original, la eficiencia de un procedimiento que genere dichas desigualdades para problemas (SP) vendrá dada en términos de nuestra capacidad para resolver (SPA).

Asimismo, se ha visto que los Teoremas 2 y 4 de II.3, que son los que aseguran la posibilidad de obtener desigualdades válidas a partir de cotas condicionales y que las desigualdades así obtenidas son violadas por la solución posible a partir de la que se obtienen, siguen siendo ciertos para problemas con la estructura de (SPA). Ahora bien, al igual que en el caso de problemas de (SP) puros, la utilización de los resultados antes descritos sólo será posible a partir de soluciones posibles para estos problemas. Por consiguiente, un procedimiento para obtener soluciones posibles para (SPA) nos permitirá, además, establecer un proceso algorítmico en el que se obtengan también desigualdades válidas para estos nuevos problemas.

Considerando que el conjunto de restricciones de estos problemas está claramente particionado en una familia de restricciones del tipo de set partitioning (matriz A), y otra familia con restricciones del tipo de set covering (matriz G, vector π), cuando utilicemos procedimientos heurísticos para encontrar soluciones posibles para (SPA), no podremos aplicar heurísticas que sean específicas para problemas de set covering (no aseguran que las restricciones asociadas a la matrtiz A se vayan a satisfacer como estricta igualdad), ni tampoco las que sean específicas para los problemas de set partitioning (ya que estaríamos eliminando del conjunto de soluciones posibles todas aquellas que satisfaciendo A como estricta igualdad satisfagan alguna restricción de G o la ecuación $\pi x \geq 1$ como estricta desigualdad). Por este motivo, para encontrar soluciones posibles para problemas del tipo (SPA) será necesario diseñar alguna heurística específica.

En esta sección se propone una heurística para resolver estos problemas que se basa en la propuesta en II.2 para problemas de minimización de (SP) puros y considera la estructura específica de estos nuevos problemas. Así como la heurística propuesta anteriormente, la que se propondrá a continuación se enmarca dentro de la familia de procedimientos que pueden diseñarse teniendo en cuenta el espíritu de la heurística de Fisher y Kedia [FiKe86] para problemas de maximización de (SP) puros. Por lo tanto, el procedimiento que se propone obtiene primero una solución posible (u,v,w) para el problema lineal dual asociado (DPA) y, a partir de ella, una solución posible x para (SPA). Una vez más, se utiliza la información que proporciona el vector de costes reducidos asociados a la solución (u,v,w) para intentar acercarse, en la medida de lo posible, a las condiciones de holgura complementaria; pero en este caso, tanto para obtener una subsolución posible para el problema de set covering puro (SC) contenido en (SPA), como para completar dicha subsolución, obteniendo una solución posible para el subproblema de ser partitioning puro (SPP) que resulta después de aplicar las condiciones lógicas que se derivan de fijar a 1 las variables que forman parte de la subsolución obtenida para (SC).

La notación que se utiliza es la misma que la introducida anteriormente, es decir:

A es una matriz de dimensión $(m \times n)$, tal que los elementos a_{ij} valen 0 ó 1.
$M = \{ 1,2,...,m \}$ conjunto de índices de filas de la matriz A.
$N = \{ 1,2,...,n \}$ conjunto de índices de columnas de la matriz A.
$M_i = \{ j \in N \: / \: a_{ij}=1 \}$, $\forall i \in M$
$N_j = \{ i \in M \: / \: a_{ij}=1 \}$, $\forall j \in N$

G es una matriz de dimensión $(r\text{-}1 \text{x} n)$ siendo g_{ij} 0 ó 1.
$K = \{ 1,2,...,r\text{-}1 \}$ conjunto de índices de filas de la matriz G.
$K_l = \{ j \in N \: / \: g_{lj}=1 \}$, $\forall l \in K$
$L_j = \{ i \in K \: / \: g_{ij}=1 \}$, $\forall j \in N$

II.4.1 Heurística dual

Sea el problema

$$(SPA_r) \qquad \begin{array}{l} \min cx \\ Ax = e \\ Gx \geq e \\ \pi x > 1 \\ x_j \in \{0,1\},\ \forall j \in N \end{array}$$

El problema dual continuo asociado es de la forma

$$(DA_r) \qquad \begin{array}{l} \max ue+ve+wx \\ uA+vG+w\pi \leq c \\ v \geq 0,\ w \geq 0 \end{array}$$

siendo r-1 el número de filas de la matriz G.

Teniendo en cuenta que el subproblema

$$(SPP) \qquad \begin{array}{l} \min cx \\ Ax = e \\ x_j \in \{0,1\},\ \forall j \in N \end{array}$$

es un subproblema de set partitioning puro que está contenido en (SPA_r) $\forall r$, a partir de cualquier solución posible u* del problema lineal dual asociado a (SPP)

$$(SDP) \qquad \begin{array}{l} \max ue \\ uA \leq c \end{array}$$

podremos disponer, al menos, de una solución posible de la forma (u*,0,0) para (DA_r).

Por lo tanto, podemos suponer que conocemos una solución inicial posible de la forma $(u^*,v^*,0)$ para (DA_r) y que deseamos obtener otra solución posible diferente de $(u^*,v^*,0)$, ya que de esta forma la información referente la restricción π que se están añadiendo al problema primal se reflejarán también en la nueva solución dual que se obtenga.

Inicialmente a la variable dual w se le asigna un valor no negativo de forma que se pueda asegurar que la nueva solución que se obtenga sea diferente de la inicial $(u^*,v^*,0)$. La variable w toma el máximo valor que mantendría la factibilidad en la última restricción dual si todas las demás componentes fuesen 0. Por lo tanto, el vector (u^*,v^*,w) ya no tiene por qué seguir siendo una solución posible para (DA_T).

El procedimiento que se propone irá reduciendo lo menos posible las componentes de las restantes variables duales u y v hasta conseguir obtener una nueva solución posible dual. Primero se reducen los valores de las variables duales asociadas a las restricciones del problema primal de "≥" y posteriormente, si es necesario, las asociadas a las restricciones de igualdad del problema primal. La cantidad en la que se reduce la variable correspondiente es la mínima necesaria para que dejen de violarse todas las restricciones del problema dual en las que interviene dicha variable. Teniendo en cuenta que en el caso de las variables asociadas a las restricciones de "≥" del problema primal, no siempre será posible reducir estas variables en dicha cantidad ya que están restringidas en signo a ser positivas, dichas variables se reducirán en lo que sea posible.

Finalmente, si existen variables duales que sólo intervienen en restricciones que se satisfacen como estricta desigualdad, dichas variables se incrementarán en la máxima cantidad posible, manteniendo la factibilidad del vector resultante.

Sea $s = c - u^*A - v^*G$ el vector de costes reducidos asociados a la solución inicial $(u^*,v^*,0)$, y sea $P = \{ j \in N / \pi_j = 1 \}$. La heurística es como sigue:

__Heurística__

Inicialización

$w = \min \{ c_j / j \in P \}$	
Para $j = 1,n$ **hacer**	*Actualizar el vector de costes reducidos*
si $j \in P$ **entonces** $s_j \leftarrow s_j - w$	*para el nuevo valor de la variable w.*
Fin para	
Sean $V = \{ j \in P / s_j < 0 \}$	*Conjunto de índices de restricciones violadas por el nuevo vector.*
$M_A = \{ i \in M / \exists\, j \in N_i \cap V \}$	*Conjunto de índices de variables duales u_i que intervienen en alguna restricción violada.*
$M_G = \{ l \in K / \exists\, j \in L_l \cap V$ t.q. $v_j > 0 \}$	*Conjunto de índices de variables duales v_l que tiene un valor estrictamente positivo y que intervienen en alguna restricción violada.*

Para $i \in M_A$ **hacer**
    $numviol\_i = |M_i \cap V|$
Fin para

numviol_i es el número de restricciones violadas en las que interviene la variable dual asociada u_i.

Para $l \in M_G$ **hacer**
    $numviol\_l = |M_l \cap V|$
Fin para

numviol_l es el número de restricciones violadas en las que interviene su variable dual asociada v_l.

Fin inicialización

Mientras $V \neq \emptyset$ **hacer**
  Si $M_G \neq 0$ **entonces**
    Sea $l \in M_G$ t.q. $numviol\_l = \min_{k \in M_G} numviol\_k$
      $\Delta_l = \min \{ v_l, - \min_{j \in K_l} s_j \}$
      $v_l \leftarrow v_l - \Delta_l$
    Para $j \in K_l$ **hacer**
      Si $s_j < 0$ y $s_j + \Delta_l > 0$ **entonces**
        Para $i \in M_j$ $numviol\_i \leftarrow numviol\_i - 1$
        Para $s \in L_j$ $numviol\_s \leftarrow numviol\_s - 1$
      Fin si
      $s_j \leftarrow s_j + \Delta_l$
    Fin para
  En otro caso
    Sea $i \in M_A$ t.q. $numviol\_i = \min_{k \in M_A} numviol\_k$
      $\Delta_i = - \min_{j \in M_i} s_j$
      $u_i \leftarrow u_i - \Delta_i$
    Para $j \in N_i$ **hacer**
      Si $s_j < 0$ y $s_j + \Delta_i > 0$ **entonces**
        Para $k \in M_j$ $numviol\_k \leftarrow numviol\_k - 1$
        Para $l \in L_j$ $numviol\_l \leftarrow numviol\_l - 1$
      Fin si
      $s_j \leftarrow s_j + \Delta_i$
    Fin para
  Fin si
Fin mientras

Para $i \in M$ **hacer**

$u_i \leftarrow u_i + \max \{ 0, \min_{j \in N_i} s_j \}$

Fin para

Para $s \in K$ **hacer**

$v_s \leftarrow v_s + \max \{ 0, \min_{j \in K_s} s_j \}$

Fin para

Aplicar la heurística de mejora para problemas duales de problemas de set partitioning puros a las variables duales u_i, $i \in M$.

<u>**Fin heurística**</u>

Hay que hacer notar que en el caso que $r = 1$ la solución inicial $(u^*,0)$ se obtiene mediante la heurística para problemas duales asociados a problemas de minimización de set partitioning propuesta anteriormente. Para valores de $r > 1$, aplicando de forma iterativa el procedimiento anterior, siempre será posible conocer una solución inicial de la forma $(u^*,v^*,0)$ donde por lo menos una componente del vector v^*, la última, será no nula. Dicha solución vendrá dada por $(u_{r-1}{}^*,v_{r-1}{}^*,w_{r-1}{}^*,0)$, siendo $(u_{r-1}{}^*,v_{r-1}{}^*,w_{r-1}{}^*)$ la solución obtenida por este procedimiento para el problema (DA_{r-1}) en el que la desigualdad $\pi x \geq 1$ es la restricción correspondiente a la (r-1)-ésima fila de matriz G.

II.4.2 Heurística primal

A continuación se propone una heurística para encontrar soluciones posibles para problemas con la estructura de (SPA). Por cuestión de notación supondremos que el conjunto de todas las restricciones de tipo set covering (incluida la que en la heurística dual se denomina π) viene representado por la matriz G. Por lo tanto, ahora G será una matriz de dimensión rxn.

La heurística, que se aplica de forma iterativa mientras no se encuentra una solución posible y mientras no sea vacío el conjunto de variables que se consideran candidatas a formar parte de una solución, consta de dos fases: en la primera, se intenta encontrar una solución posible para el subproblema de set covering

(SPC) $\min \{ cx \,/\, Gx \geq e\}$,

imponiendo además la condición de que en cada momento la sub-solución parcial que se esté construyendo pueda ser parte de una sub-solución parcial posible del subproblema de set partitioning (SPP) min { cx / Ax=e }; es decir imponiendo dos condiciones:

1) que el conjunto de variables que se ha fijado a 1 sea ortogonal con respecto a la matriz A y que

2) el subproblema resultante al eliminar todas las variables que se hayan fijado a 1 no sea no factible.

En la segunda fase se intenta completar la sub-solución parcial obtenida en la primera fase. Para ello se plantea el subproblema (SPP) de set partitioning puro que resulta de eliminar de (SPA) todas las restricciones del tipo de set covering, que ya sabemos se satisfacen con la sub-solución parcial obtenida, y todas aquellas restricciones del tipo de set partitioning que también se satisfagan con la sub-solución parcial actual. Posteriormente, se aplican los test lógicos al subproblema (SPP) y se aplica la heurística propuesta en II.2 al problema de set partitioning de minimización que se obtenga despues de hacer las eliminaciones pertinentes.

La decisión de intentar satisfacer en primer lugar el conjunto de restricciones asociado a la matriz G se debe a que ninguna de las soluciones obtenidas previamente, a las que no hemos impuesto la condición de satisfacer la última restricción asociada a la matriz G (la desigualdad $\pi x \geq 1$), ha proporcionado un valor mejor que z_{sup} y al hecho, ya conocido, de que cualquier solución posible para el problema original con un valor de la función objetivo mejor que el que nos haya proporcionado el valor z_{sup}, si es que existe, deberá satisfacer esta última desigualdad.

Sean

R el conjunto de índices de variables que son candidatas a entrar en la solución.

M^* el conjunto de índices de filas de A que ya están recubiertas por la solución parcial que se tiene.

K^* el conjunto de índices de filas de G que ya están recubiertas por la solución parcial que se tiene.

N_i^* el conjunto de índices de variables que intervienen en la i-ésima restricción de A, susceptibles de ser elegidas para formar parte de la solución.

K_l^* el conjunto de índices de variables que intervienen en la k-ésima restricción de G, susceptibles de ser elegidas para formar parte de la solución.

Un esquema de la heurística es el siguiente:

Heurística

Inicialización	
$R=N$	*Al principio el conjunto de índices de variables elegibles es todo el conjunto N.*
$M^{*}=\emptyset$ $K^{*}=\emptyset$	*Inicialmente ninguna fila de A ni de G está recubierta.*
$N_i^{*}=N_i, \forall i \in M$ $K_l^{*}=K_l, \forall l \in K$	*Para todas las filas, tanto de A como de G, todas las variables que intervienen en cada fila puede ser en principio candidata a formar parte de la solución.*
fin=falso	*El procedimiento terminará cuando se haya encontrado una solución posible para (SPA) o bien cuando el conjunto de variables elegibles sea vacío.*
Fin Inicialización	

Mientras no fin **hacer**
  Buscar solución posible para (SPC)
  Si la solución posible de (SPC) no es solución posible para (SPA) **entonces**
    Construir el subproblema (SP) resultante de aplicar los test lógicos a la matriz A obtenida al fijar a 1 las variables que forman la solución de (SPC).
    Aplicar heurística para set partitioning a (SP).
    Si no se encuentra solucion posible para (SP) **entonces**
      Sea j^{*} el índice de la última variable fijada a 1
      Hacer

$x_{j}^{*} \leftarrow \emptyset$	*Eliminar j^{*} de la solucion actual de (SP).*
$M^{*} \leftarrow M^{*} \setminus M_j^{*}$ $K^{*} \leftarrow K^{*} \setminus K_j^{*}$	*Actualizar el conjunto de filas satisfechas tanto en A como en G.*

      Fin hacer
    Fin si
Fin si
Fin mientras
Fin heurística

Con respecto de la heurística anterior, hay que resaltar que no está asegurado el éxito de la misma. Una vez más, hay que señalar que aunque existe la posibilidad de que falle, en la práctica ésto raramente ocurre. De hecho, veremos posteriormente que el criterio de terminación que se ha establecido en el algoritmo completo para resolver problemas de (SP) que se propone es precisamente el que esta heurística falle. Esto se debe a que si en un momento determinado disponemos de una solución óptima y se genera una desigualdad válida, el nuevo problema ampliado no tiene por qué tener solución posible. En ese sentido, el fallo de la heurística puede ser un buen indicador de dicha situación.

La primera fase del procedimiento que se ha propuesto para obtener soluciones posibles para problemas del tipo (SPA) obtiene una solución posible para el subproblema de (SC) contenido en (SPA). En esta primera parte, se intentan satisfacer las restricciones asociadas a la matriz G de forma secuencial fijando a 1 ciertas variables con una heurística de tipo greedy.

El criterio que se sigue es, como siempre, intentar satisfacer en la medida de lo posible las condiciones de holgura complementaria. Para ello, se eligen, de forma secuencial, variables que se fijan a 1. En cada paso se elige, entre las variables que no hayan sido consideradas todavía, aquella variable que tenga un coste reducido lo menor posible, que intervenga en la última restricción no satisfecha de la matriz G, que sea ortogonal respecto a la matriz A con todas las variables que ya estén previamente fijadas a 1 y cuya eliminación no produzca un subproblema no factible con respecto a la matriz A ni a la matriz G.

El procedimiento que busca una solucion posible para (SPC) es el siguiente:

Heurística

Inicialización

fin1=falso	*El procedimiento terminará cuando se encuentre una solución posible para (SPC) o bien cuando el conjunto restante de variables elegibles sea vacío. También terminará en caso de que veamos que el conjunto restante de variables elegibles resultante no permita satisfacer ni siquiera la primera restricción de G.*

Fin inicialización

Mientras no fin1 **hacer**

Sea l el último elemento de $K \backslash K^*$ — *l es el índice de la última restricción no satisfecha de la matriz G.*

Buscar $j \in K_l^*$ tal que: — *Buscar, entre las variables que intervengan en la fila l, una que esté enel conjunto de las elegibles.*

i) $s_j = \min_{k \in K_l^*} s_k$ — *Tenga el coste reducido lo menor posible*

ii) $M_j \cap M^* = \emptyset$ — *No intervenga en ninguna de las restricciones ya satisfechas de la matriz A*

iii) $|N_i^* \backslash \{j\}| \neq 0, \forall i \in M_j \cap (M \backslash M^*)$ — *No existe ninguna fila de la matriz A que no se pueda satisfacer al eliminar la variable j y aplicar los tests lógicos .*

iv) $|K_s^* \backslash \{j\}| \neq 0, \forall s \in L_j \cap (K \backslash K^*)$ — *No existe ninguna fila de la matriz G que no se pueda satisfacer al eliminar la variable j y aplicar los tests lógicos.*

Si existe j **entonces**

Hacer $x_j \leftarrow 1$ — *Añadir j a la solución actual.*

$M^* \leftarrow M^* \cup M_j$ — *Actualizar el conjunto de restricciones*

$K^* \leftarrow K^* \cup K_j$ — *satisfechas tanto en A como en G .*

$R \leftarrow R \backslash \{j\}$ — *Actualizar el conjunto de variables elegibles.*

$N_i^* \leftarrow N_i^* \backslash \{j\}, \ \forall i \in M_j$ — *Para cada una de las filas de A y de G*

$K_s^* \leftarrow K_s^* \backslash \{j\}, \ \forall s \in K_j$ — *actualizar el conjunto de variables elegibles*

en otro caso — *que intervienen en ella.*

si l=r **entonces** — *El conjunto de variables elegibles no permite*

fin1=cierto — *satisfacer ni siquiera la última restricción de*

en otro caso — *G que se ha generado.*

Sea j^* el índice de la última variable que se ha fijado a 1

Hacer $x_j^* \leftarrow 0$ — *Eliminar de la solucion actual de (SP) la última variable fijada a 1 .*

$M^* \leftarrow M^* \backslash M_j$ — *Actualizar el conjunto de restricciones*

$K^* \leftarrow K^* \backslash K_j$ — *satisfechas tanto en A como en G*

Fin si

Fin si

si $K^* = K$ o $R = \emptyset$ **entonces** fin1=cierto

Fin mientras

<u>**Fin heurística**</u>

CAPÍTULO III

MÉTODOS DUALES PARA PROBLEMAS DE SET PARTITIONING

En el capítulo II se ha visto cómo obtener distintas soluciones posibles, tanto primales como duales, mediante procedimientos heurísticos para problemas de set partitioning. Para mejorar la calidad de las soluciones primales nos hemos servido del procedimiento de generación de planos secantes obtenidos a partir de disyunciones que reduce el espacio de búsqueda y que, en general, proporciona una muy buena cota superior, que en casi todos los casos se trata del óptimo, después de un número reducido de iteraciones.

Desgraciadamente, la calidad de las cotas inferiores que obtenemos mediante procedimientos heurísticos, en ocasiones no nos permite demostrar que disponemos del óptimo del problema primal en los casos en los que esto es así, ni tampoco asegura nada sobre la calidad de la incumbente ya que el gap que se presenta entre la mejor solución primal y la mejor cota inferior es casi siempre demasiado grande.

Por todo ello, la cuestión que se plantea a continuación es la de poder reforzar la calidad de las cotas inferiores para reducir el gap de dualidad en la medida de lo posible. En las tres primeras secciones de este capítulo se exponen tres técnicas, ya conocidas, que hemos utilizado con este fin en la resolución de problemas de set partitioning. La primera de ellas es la relajación lagrangiana [Geo74, Sha79, Fis81], la segunda es un método de refuerzo dual propuesto por Barcia [Barci85a, Barci85b] y la tercera es la relajación subrogada [Geo69, Glo75, Dye80]. La aplicación de cada una de ellas a problemas de (SP) se estudia en detalle en una sección posterior.

III.1 Relajación Lagrangiana

La relajación lagrangiana es una técnica ampliamente utilizada y que ha dado buenos resultados en la resolución de distintos problemas de programación entera o mixta. Geoffrion [Geo74] realizó un estudio sistemático de la misma en el se analizan las distintas propiedades teóricas y la relación de esta técnica con la teoría general de la dualidad. Esta técnica, permite obtener cotas inferiores para los problemas, mediante una relajación de los mismos, que consiste en incorporar, total o parcialmente, las restricciones del problema original a la función objetivo, de un modo similar al de las funciones de penalización. La idea de incorporar parte de las restricciones, a las que se asocia un conjunto de multiplicadores, a la función objetivo se basa en la conocidísima técnica de los multiplicadores de Lagrange para problemas clásicos de optimización diferenciables.

A continuación se expone un breve resumen de los principales resultados de la teoría de la relajación lagrangiana y de la relación de la misma con la teoría general de la dualidad [Geo74, Sha79, Fish81]. En este resumen se considera que el problema de optimización es un problema general de minimización en el que las restriciones son de "≥". La dualización de estos problemas exige que el vector dual asociado esté restringido en signo a ser no negativo. Sin embargo, los resultados que se exponen a continuación siguen siendo todos ciertos en el caso en el que las restricciones del problema original sean de estricta igualdad. Ahora bien, en este último caso el vector dual asociado no estará restringido en signo.

Dado un problema de minimización

$$
\text{(P)} \quad \begin{array}{l} \min cx \\ Ax \geq b \\ Bx \geq d \\ x_j \text{ entero, } j \in I \end{array}
$$

siendo A,B matrices dadas de dimensiones adecuadas y c,b,d vectores dados de dimensiones conocidas.

Se define la relajación lagrangiana de (P) respecto a las restricciones $Ax \geq b$, y al vector no negativo de multiplicadores u como el problema

$$(PRu)\quad \min_{x\geq 0}\; cx + u\,(b-Ax)$$

$$Bx \geq d$$

$$x_j \text{ entero, } j\in I$$

Cualquier relajación lagrangiana de un problema (P) es una relajación en el sentido clásico del término ya que se cumplen las dos siguientes propiedades:

1) El conjunto de soluciones posibles de (P) está contenido en el conjunto de soluciones posibles de (PRu), es decir $F(P) \subseteq F(PRu)$, y

2) El valor de la función objetivo de (PRu) es menor o igual que el valor de la función objetivo de (P) para cualquier solución posible de F(P), ya que si x es una solución posible para (P) se tiene que $Ax\geq b$ con lo que $u(b-Ax)\leq 0$.

En particular, denotando por v(P) el valor óptimo del problema (P), se tendrá que para cualquier $u\geq 0$ $v(PRu) \leq v(P)$.

Teniendo en cuenta el hecho de que una determinada relajación lagrangiana será tanto mejor en la medida en que su valor óptimo se acerque al de (P), podemos definir un criterio sobre la bondad de un vector de multiplicadores concreto. Este criterio será el de considerar que un vector de multiplicadores es tanto mejor en la medida que proporcione un valor más alto de la función objetivo.

En particular, la mejor elección será la de tomar el vector u como una solución óptima del problema dual lagrangiano

$$(D)\quad \max_{u\geq 0}\; v(PRu)$$

que coincide con el problema dual de la siguiente Relajacion de (P)

$$(P^*)\quad \min cx$$

$$Ax \geq b$$

$$x \in \text{Conv}\,\{\, x\geq 0 \,/\, Bx\geq d,\ x_j \text{ entero } j\in J \,\}$$

Las relaciones básicas entre (P), (PRu), (D), (P^*) y ($\overline{P}$) son bien conocidas y se enuncian en el siguiente

Teorema 1: $\forall u \geq 0$

1) $F(P) \subseteq F(P^*) \subseteq F(\overline{P})$, $F(P) \subseteq F(PRu)$

$v(P) \geq v(P^*) \geq v(\overline{P})$, $v(P) \geq v(PRu)$

2) Si $(\overline{P})$ es factible, entonces $v(\overline{P}) \leq v(\overline{PRu})$.

3) Si para un vector dado u, se tiene que un vector x cumple

i) x es óptimo en (PRu)
ii) $Ax \geq b$
iii) $u(b-Ax) = 0$,

entonces, x es una solución óptima para (P).

Si x cumple i),ii) pero no iii) entonces x es una solución ε-óptima para (P) con $\varepsilon = u(Ax-b)$.

4) Si (P^*) es factible, entonces

$$v(D) = \max_{u \geq 0} v(PRu) = v(PRu^*) = v(P^*)$$

El apartado 2) afirma que la Relajación Lagrangiana asociada al vector $\overline{u}$, solución del problema dual de ($\overline{P}$), proporciona una cota inferior que es cuando menos tan buena como la que proporciona ($\overline{P}$).

El apartado 3) establece las condiciones para que la solución de una relajación lagrangiana sea también óptima o cuasi-óptima en (P). Ello refuerza el interés de la utilización de la relajación lagrangiana en el sentido de que no sólo podemos obtener cotas inferiores, sino que eventualmente podremos encontrar una solución óptima o cuasi óptima para (P).

El apartado 4) asegura que en ningún caso la cota obtenida mediante una relajación lagrangiana puede ser mejor que la obtenida por (P^*). En este sentido, a la hora de utilizar la Relajación Lagrangiana para un tipo determinado de problemas, una cuestión que juega un papel primordial será la situación del valor v(P^*) en el intervalo [v($\overline{P}$),v(P)].

El siguiente teorema establece la relación entre v(P^*) y v($\overline{P}$). Ello será posible gracias a la siguiente *propiedad de integridad* :
El valor óptimo de (PRu) no cambia al eliminar las restricciones de integridad de las variables; es decir, v(PRu)=v($\overline{PRu}$) $\forall u \geq 0$

<u>Teorema 2</u>: Sean ($\overline{P}$) factible y (PRu) tal que cumple la condición de integridad. Entonces (P*) es factible y además

$$v(\overline{P}) = v(\overline{PRu}) = v(D) = v(PRu^*) = v(P^*)$$

Teniendo en cuenta que para relajaciones lagrangianas que cumplan la propiedad de integridad no vamos a poder obtener una cota mejor que la de ($\overline{P}$), el teorema anterior limita la utilización de relajaciones lagrangianas que cumplan dicha condición, al caso en que se disponga de un método específico para calcular una solución óptima o cuasi-óptima para (D) más rápido que la resolución de la relajación lineal ($\overline{P}$). En este sentido, parece más adecuada la utilización de relajaciones lagrangianas que no cumplan la mencionada condición.

Otra relación que es interesante conocer, es la existente entre v(P*) (y por tanto v(D)) y v(P). Una condición suficiente para que v(P) = v(P*) es que F(P*) = conv [F(P)]. La verificación de la condición anterior entra de lleno en el terreno de la combinatoria poliédrica, y es normalmente muy difícil, dada la complejidad de los poliedros de los distintos problemas enteros.

El siguiente teorema establece, sin embargo, condiciones equivalentes a la de asegurar que v(P) = v(D). Algunas de ellas vienen dadas en términos de la *función de b-parturbación* definida como

$$\Phi_b(y) = \inf_{x \geq 0} cx$$

$$Ax \geq b-y$$

$$Bx \geq d$$

$$x_j \text{ entero}, j \in J.$$

<u>Teorema 3</u>: Supongamos que (P) factible (y por lo tanto tiene soluciones óptimas ya que todas las variables están acotadas)

1) Las siguientes afirmaciones son equivalentes:

i) v(P) = v(D)

ii) Existe un subgradiente global de $\Phi_b(y)$ en y = 0

iii) Existe un par (x,u) tal que $u \geq 0$ y además cumple las condiciones i), ii), iii) del Teorema 1.3.

2) Si v(P) = v(D), entonces:

i) Una solución u es óptima para (D) $\Leftrightarrow$ es el negativo de un subgradiente global para $\Phi_b(y)$ en $y = 0$.

ii) Dada una solución u^*, óptima para (D), entonces el conjunto de soluciones óptimas para (P) vendrá dado por todos los vectores x que satisfagan las condiciones i), ii), iii) del Teorema 1.3. para $u = u^*$.

De todo lo anterior se deduce que, a la hora de utilizar la relajación lagrangiana, la primera decisión que se debe tomar es la de elegir una formulación equivalente del problema original que favorezca, en virtud de su estructura, formular un problema relajado que persiga dos objetivos:

1) Obtener problemas duales lagrangianos que sean relativamente fáciles de resolver, para que de esta manera el coste computacional necesario para obtener una mejor cota inferior no sea excesivo.

2) Conseguir que la estructura del problema relajado que se plantee refleje en lo posible la estructura del problema original para poder aprovechar la información de la solución que se obtenga en la construcción de soluciones posibles y, además, asegurar la calidad de la cota inferior resultante.

No es tarea fácil alcanzar los objetivos anteriores ya que en gran medida son contradictorios; una formulación que consiga un equilibrio entre ambos puede ser, sin embargo, la clave del éxito.

III.2 Método BISA de refuerzo dual

En esta sección se expone el algoritmo BISA que se obtiene a partir de una teoría general para métodos duales constructivos en programación discreta [Barci85a, Barci85b]. Esta técnica se basa en la reducción del conjunto de soluciones posibles para obtener programas duales que sean fáciles de resolver y para los que no exista gap de dualidad o para los que éste sea lo menor posible. Mientras un problema dual concreto no consigua resolver el problema primal, se construye sucesivamente un problema dual más 'potente'. La relajación que se formula en cada iteración se hace cada vez más restringida hasta conseguir una solución óptima en un número finito de iteraciones.

Al final de la sección se propone una modificación del algoritmo propuesto por Barcia en su trabajo. La variante que se propone consiste en realizar una única iteración del algoritmo en la que se actualizará el conjunto de soluciones posibles para el problema dual. Esta modificación viene motivada por el hecho de que en el algoritmo propuesto por Barcia [Barci85a, Barci85b] el número de iteraciones necesario para alcanzar la convergencia, a pesar de estar acotado, puede resultar bastante grande con lo que computacionalmente el método puede resultar muy costoso. Veremos, además, una justificación teórica para dicha variante.

El método que se expone a continuación puede enmarcarse dentro de los métodos duales constructivos. Estos métodos se basan en la construcción de una secuencia de problemas duales que dan lugar a una sucesión de cotas inferiores cada vez más próximas al valor del óptimo del problema primal, de forma que en un número finito de iteraciones se obtiene el valor óptimo del problema primal. Estos métodos fueron inicialmente propuestos por Bell y Shapiro [BeSh77] utilizando la teoría de grupos abelianos. El inconveniente principal que plantea el método que proponían es que los algoritmos obtenidos resultan computacionalmente ineficientes. En este sentido, el trabajo desarrollado por Barcia [Barci85a, Barci85b] resulta de especial validez ya que resuelve dicha cuestión al plantear una sucesión de problemas duales que resultan sencillos de resolver computacionalmente ya que se trata de problemas de tipo knapsack.

Sea un problema entero puro de la forma

$$\text{(IP)} \quad z = \min cx$$
$$Ax \geq b$$
$$x \in X$$

siendo, como siempre A una matriz dada de dimensiones adecuadas y $c \in Z^{n+}$ y $b \in R^n$ dos vectores conocidos.

El problema dual lagrangiano de (IP) viene definido por

$$\text{(DIP)} \quad s = \max_{u \geq 0} \min_{x \in X} \; cx + u(b-Ax)$$

Sea x^* una solución óptima para (IP). Definiendo un conjunto X(k) tal que $x^* \in X(k) \subseteq X$, entonces (IP) será equivalente al problema (IPk) definido como

$$\begin{aligned} \text{(IPk)} \quad z = \; & \min \; cx \\ & Ax \geq b \\ & x \in X(k), \end{aligned}$$

en el sentido de que tanto (IP) como (IPk) tendrán el mismo valor óptimo.

El problema (IPk) permite, además, otener una cota inferior mejor del valor de (IP) puesto que si consideramos el problema dual de (IPk)

$$\text{(DIPk)} \quad s(k) = \max_{u \geq 0} \min_{x \in X(k)} \; cx + u(b-Ax)$$

podremos mejorar la cota inferior s proporcionada por (DIP) ya que se tiene la relación $s \leq s(k) \leq z$.

La propiedad anterior resulta esencial en el diseño de algoritmos cuyo objetivo sea obtener una familia de cotas inferiores sucesivamente mejores para el valor de (IP) y que terminen en un número finito de iteraciones. Para ello, bastará con considerar en cada iteración k conjuntos X(k) cada vez más restringidos hasta conseguir eliminar el gap de dualidad, es decir hasta que $s(k) = z$.

El siguiente teorema proporciona además una condición de terminación para un algoritmo de este tipo.

<u>Teorema 1</u>: Sea u^* el vector de multiplicadores óptimo para el problema (DIPk), y sea $\{x(i)\}_{i \in I(k)}$, $I(k)=\{1,...,p(k)\}$ el conjunto de soluciones óptimas del problema asociado

$$\min_{x \in X(k)} \; cx + u^*(b-Ax)$$

Si $\forall i \in I(k)$, x(i) es una solución posible para (IP), entonces por lo menos una de ellas es óptima para (IP).

Por lo tanto, en una iteración determinada se sabrá que se ha alcanzado el óptimo del problema original (IP) cuando sean posibles para el mismo todas las soluciones óptimas asociadas al vector de multiplicadores u* óptimo para el problema (DIPk),

En consecuencia, un esquema general de un algoritmo de refuerzo dual será el siguiente:

Inicialización

k=0

X(0)=X

Fin = falso

Fin Inicialización

Mientras no fin **hacer**

Resolver (DIPk)

Sean u^* el multiplicador óptimo y

$x(i), i \in I(k)=\{ 1,...,p(k) \}$ el conjunto de soluciones óptimas para

$$\min_{x \in X(k)} cx + u^*(b-Ax)$$

Si $\forall i \in I(k)$ x(i) es factible para (IP) **entonces**

fin= cierto *Por lo menos una de ellas es óptima*

en otro caso

Sea $x(k^*)$, $k^* \in I(k)$ una solución no factible para (IP).

Tomar $X(k+1) \subset X(k) \setminus \{ x(k^*) \}$ t. q. $x^* \in X(k+1)$.

Hacer $k \leftarrow k+1$

Fin ***si***

Fin mientras

Es evidente que, si la cardinalidad de X es finita, un algoritmo del tipo anterior terminará en un número finito de iteraciones.

El esquema general anterior presenta, sin embargo, el incoveniente de que a pesar de que la sucesión de cotas que se obtiene es no decreciente, el número de iteraciones necesario para alcanzar el óptimo puede ser arbitrariamente grande. Por ese motivo, interesa definir los conjuntos X(k) de forma que pueda asegurarse que la sucesión de cotas s(k) que se obtenga sea estrictamente monótona creciente.

En ese sentido, resulta fundamental obtener condiciones que aseguren que la sucesión de cotas obtenida es estrictamente monótona creciente puesto que ello permitirá demostrar que el número de iteraciones necesario para alcanzar el óptimo será reducido.

En [Barci85a, Barci85b] se dan condiciones necesarias y suficientes para conseguir este objetivo. Aquí las enunciamos para una mejor comprensión del algoritmo que se expondrá posteriormente.

Teorema 2: La condición necesaria y suficiente para que un algoritmo del tipo anterior genere una sucesión de cotas estrictamente monótona creciente es que el conjunto $X(k+1) \subset X(k)$ sea tal que $\text{Conv } X(k+1) \cap O(k) = \emptyset$, siendo O(k) el conjunto de soluciones óptimas para CIP(k) (problema IP(k) convexificado).

Proposición 1: Una condición necesaria para que $\text{Conv } X(k+1) \cap O(k) = \emptyset$, es que X(k+1) sea tal que cumpla las 3 siguientes propiedades:

i) $x^* \in X(k+1)$
ii) $X(k+1) \subset X(k) \setminus \{x(k^*)\}$
iii) $\text{Conv } X(k+1) \cap O(k) = \emptyset$

La Proposición 1 proporciona una herramienta potente para obtener conjuntos X(k) de forma que la sucesión de cotas que se obtenga sea estrictamente monótona creciente. En particular, en el caso de que el problema que se quiera resolver tenga la estructura de (IP) se puede demostrar que definiendo

$X(k+1) = \{ x \in X \,/\, cx \geq \lceil s(k) \rceil \}$, donde $\lceil y \rceil$ es el entero inmediatamente posterior a y,

se cumplen las propiedades i), ii) y iii).

Además, definiendo de esta forma los conjuntos X(k) el siguiente teorema proporciona una condición necesaria y suficiente para conocer que se ha alcanzado el óptimo del problema (IP).

Teorema 3: Dado un entero $s(k) \leq z$ que sea un punto fijo para el problema

$$s(k) = \max_{u \geq 0} \min_{\substack{cx \geq s(k) \\ x \in X}} cx + u(b-Ax),$$

la condición necesaria y suficiente para que s(k) sea el valor óptimo para (IP) (s(k) = z) es que exista una solución óptima para el problema anterior de la forma $(x^*,0)$ siendo x^* factible para (IP).

Teniendo en cuenta los resultados anteriores es fácil ver que el siguiente algoritmo genera una sucesión de cotas estrictamente monótona creciente y alcanza el óptimo en como mucho $z - \lceil s(0) \rceil$ iteraciones.

Algoritmo BISA (Bound Improving Sequence Algorithm)

Inicialización

calcular $s(0) = \max_{u \geq 0} \min_{x \in X(k)} cx + u(b-Ax)$

hacer k=0

fin = falso

Fin Inicialización

Mientras no fin **hacer**

Mientras s(k) sea no entero **hacer**

calcular $s(k+1) = \max_{u \geq 0} \min_{\substack{cx \geq \lceil s(k) \rceil \\ x \in X}} cx + u(b-Ax)$

hacer $k \leftarrow k+1$

Fin Mientras

Si existe $(x^*,0)$ tal que $Ax^* \geq b$ y sea óptimo para el problema de punto fijo

$$s(k) = \max_{u \geq 0} \min_{\substack{cx \geq s(k) \\ x \in X}} cx + u(b-Ax)$$

entonces fin = cierto

Fin Mientras

Hay que hacer notar que en cada iteración del algoritmo anterior hay que resolver un subproblema del tipo

$$s(k+1) = \max_{u \geq 0} \min_{\substack{cx \geq \lceil s(k) \rceil \\ x \in X}} cx + u(b-Ax)$$

Estos problemas, que no son sino relajaciones Lagrangianas de los problemas (IPk), pueden resolverse eficientemente utilizando optimización subgradiente [HeWoCr74] o cualquier variante de este tipo de procedimientos [CaFrMa75, BeMi73].

En las iteraciones internas del algoritmo de optimización subgradiente habrá que resolver subproblemas con una estructura de tipo knapsack. Estos subproblemas podrán resolverse eficientemente utilizando alguno de los distintos algoritmos que existen para su resolución como, por ejemplo, el de Martello y Toth [MaTo79] o alguno de los que utiliza programación dinámica [GiGo61, MaTo79].

El algoritmo BISA resulta teóricamente de gran interés y la convergencia del mismo está garantizada en un número finito de iteraciones. Presenta, sin embargo, inconvenientes en términos de eficiencia a la hora de su implementación. Ello se debe a que, si tenemos en cuenta que en cada iteración debemos resolver un problema por optimización subgradiente, a de pesar que la convergencia de algoritmos de este tipo está suficientemente probada bajo condiciones adecuadas y que en cada una de las iteraciones internas se puede conseguir un nivel satisfactorio de eficiencia, la repetición del mismo en varias iteraciones puede dar lugar a que los resultados computacionales obtenidos no resulten muy rentables.

Lo anterior resultará especialmente inconveniente en problemas en los que el gap inicial, $z-\lceil s(0)\rceil$, no sea suficientemente reducido; es decir en problemas para los que la relajación Lagrangiana ordinaria produzca cotas inferiores poco ajustadas (veremos más adelante que en problemas de Set Partitioning esto ocurre a menudo).

Barcia [Barci85a, Barci85b], consciente de estos inconvenientes, sugiere una mejora del BISA en la que en cada iteración del algoritmo no se exige la resolución al óptimo del subproblema (DIPk). Esto es posible gracias a la teoría de la relajación lagrangiana [Geof74] que nos asegura que para cualquier par de vectores (u,x) tal que $u \geq 0$ y $x \in X(k+1)$, el valor de la función objetivo de (DIPk) para (u,x) siempre es una cota inferior del valor $s(k+1)$.

Por lo tanto, teniendo en cuenta que el objetivo del algoritmo es el de incrementar el valor de la cota inferior $\lceil s(k)\rceil$, podremos dar por finalizada una iteración no ya en el momento en el que se consiga la convergencia, sino en el momento en el que para un par posible (u,x) se obtenga un valor de la función objetivo para (DIPk) que supere al de la cota inferior actual $\lceil s(k)\rceil$.

La modificación propuesta por Barcia [Barci85a, Barci85b] para su algoritmo sólo reduce parcialmente los inconvenientes anteriormente citados; ello es debido a que, a pesar de que ahora cada una de las iteraciones resultará más rápida, en ningún caso se reduce el número total de iteraciones a ejecutar por el algoritmo; por el contrario, en general el número de éstas aumentará. Bastará para ello con que en alguna iteración el valor del óptimo supere en más de una unidad el valor $\lceil s(k) \rceil$.

Este inconveniente puede, sin embargo, superarse diseñando un algoritmo similar al anterior, pero que conste de una única iteración global. En esta iteración se irá actualizando el valor de la cota inferior $\lceil s(k) \rceil$ cada vez que se tenga un valor de la función objetivo para (DIPk) que supere este valor.

Teniendo en cuenta la teoría de la relajación lagrangiana, si $\lceil s(k) \rceil$ es una cota inferior del valor de z, siempre se irán obteniendo valores de la función objetivo que serán cotas inferiores de z, y por lo tanto no existe ningún inconveniente para realizar la actualización sucesivamente dentro de una misma iteración.

Resulta fundamental poder asegurar la convergencia de este nuevo procedimiento. Ello resulta evidente, teniendo en cuenta que la utilización de la optimización subgradiente sólo exige en este punto [Pol67, Pol69, Gof77] que la sucesión de los parámetros que definen las longitudes de paso esté acotada superiormente por el valor 2, que converja a cero y que la serie formada por la suma de todas ellas sea divergente.

No será necesario inicializar el valor de este parámetro para garantizar las condiciones anteriores en la iteración subsiguiente a la actualización del valor de $\lceil s(k) \rceil$, puesto que el valor del mismo en dicha iteración ya satisface la condición de acotación que se acaba de mencionar. Además, a partir de esa iteración, la sucesión de los valores de los parámetros puede considerarse como una subsucesión de la inicial. Por ese motivo, la convergencia a 0 de la misma está asegurada y la suma de la serie que formen sus elementos será divergente, ya que, si de una serie divergente se elimina un número finito de sumandos, la serie resultante sigue siendo divergente.

III.3 Relajación subrogada

Otra técnica, también bien conocida, que hemos utilizado es la de la subrogación [Geo69, Glo75, Dye80]. Al igual que la de la relajación lagrangiana esta técnica se utiliza para obtener cotas inferiores para los problemas y eventualmente para encontrar soluciones óptimas de los mismos. A diferencia de la relajación lagrangiana, que incorpora a la función objetivo parte del conjunto de restricciones originales, la estrategia en la subrogación consiste en sustituir el conjunto de restricciones originales por una nueva restricción que se llama restricción subrogada. Evidentemente, a la hora de utilizar esta técnica, deberemos centrar nuestra atención en la obtención de la 'mejor' restricción subrogada.

Sea un problema primal de programación matemática

$$(P) \quad \min_{x \in X} \; f(x)$$
$$h(x) \leq 0,$$

donde f(x) y h(x) son funciones reales definidas sobre X.

Una restricción subrogada para (P) es una desigualdad de la forma $uh(x) \leq 0$, que es una combinación lineal de las restricciones $h(x) \leq 0$ asociando a cada componente $h_i(x)$ un multiplicador (coeficiente) u_i. Por definición, siempre que $u \geq 0$, una restricción subrogada vendrá implicada por el conjunto de restricciones $h(x) \leq 0$.

Se define el problema subrogado como

$$(PSu) \; s(u) = \min_{x \in X} \; f(x)$$
$$uh(x) \leq 0.$$

Su valor óptimo s(u) vendrá dado por

$$s(u) = \inf_{x \in X(u)} f(x), \text{ siendo } X(u) = \{ x \in X \,/\, uh(x) \leq 0 \}$$

Para cualquier vector u no negativo, (SPu) es una relajación del problema original (P) ya que:

1) El conjunto X de soluciones posibles de (P) está contenido en el conjunto X(u) de soluciones posibles de (PSu), y

2) El valor de la función objetivo de (PSu) es menor o igual que el valor de la función objetivo de (P) para cualquier solución posible de F(P); en particular será igual ya que la función objetivo es la misma para los dos problemas.

El valor de la cota que se obtenga mediante (SPu) será más próximo al valor óptimo del problema (P), v(P), en la medida que la restricción $uh(x) \leq 0$ refleje mejor la estructura, y en consecuencia la información, contenida en el conjunto de restricciones $h(x) \leq 0$. Tenemos, por tanto, un criterio para la elección del mejor vector de multiplicadores, que será aquel que proporcione una cota lo más ajustada posible. En este sentido, deberemos elegir u de forma que sea una solución óptima del problema dual subrogado

$$\text{(DS)} \quad s^* = \max_{u \geq 0} \ v(PSu)$$

En el caso particular de problemas enteros puros de minimización de la forma

$$\text{(IP)} \quad \begin{array}{l} \min \quad cx \\ Ax \geq e, \end{array}$$

la observación anterior se refleja de forma más concreta en el siguiente

Teorema 1: Sean z^* el valor de la relajación lineal asociada a (IP) y u^* la solución óptima del problema dual continuo asociado a dicha relajación. Entonces la mejor restricción subrogada para (IP) es $u^*Ax \geq z^*$.

Este resultado permite teóricamente conocer cual es el vector de multiplicadores óptimo a utilizar. Sin embargo, en el caso en el que no se vaya a utilizar la relajación lineal deberemos recurrir a algún otro método para hacer una estimación del conjunto de coeficientes óptimo o cuasi-óptimo. Este es sin duda el principal inconveniente que presenta la subrogación medido en términos de eficiencia a la hora de conseguir una estimación de los coeficientes.

En otro orden de cosas, es fácil observar que la relajación lagrangiana proporcionará cotas inferiores que nunca superaran las de la relajación subrogada ya que el problema

$$\text{(PRu)} \quad \inf_{x \in X} \{ f(x) + uh(x) \} \text{ es una relajación de (PSu) puesto que:}$$

1) El conjunto X de soluciones posibles de (PRu) está contenido en el conjunto X(u) de soluciones posibles de (PSu), y

2) El valor de la función objetivo de (PRu) es menor o igual que el valor de la función objetivo de (PSu) para cualquier solución posible de F(PSu), ya que

$$uh(x) \leq 0 \Rightarrow f(x) + uh(x) \leq f(x).$$

Debemos, además, tener en cuenta que si una solución óptima para (PSu) es solución posible para (P), dicha solución será óptima para (P).

Las dos observaciones anteriores permiten afirmar que la relajación subrogada es una herramienta útil a la hora de resolver problemas, no sólo porque suministra cotas inferiores que, como ya hemos visto, serán por lo menos tan buenas como las proporcionadas por la relajación lagrangiana, sino porque también puede, eventualmente, conducirnos al óptimo del problema original.

Una aplicación interesante de la relajación subrogada es su utilización para reforzar las formulaciones de distintas relajaciones lagrangianas. Obtendremos así un compromiso entre estos dos tipos de relajaciones que nos parece interesante, ya que podremos sacar partido de las ventajas de ambos métodos, eludiendo parcialmente los inconvenientes de los mismos.

En concreto, uno de los principales inconvenientes de las relajaciones lagrangianas, es el de encontrar un equilibrio entre el conjunto de restricciones a incorporar a la función objetivo y conseguir que el problema resultante tenga una formulación que contenga suficiente información sobre la estructura original del problema. Hay que tener en cuenta que cuando se relajan muchas restricciones el problema obtenido es fácil de resolver pero no proporciona cotas suficientemente ajustadas ya que es 'demasiado' relajado. Cuando esto sea así, podremos reforzar la formulación de la relajación lagrangiana añadiendo a la misma una restricción subrogada, de forma que el problema resultante siga siendo relativamente fácil de resolver, pero contenga, gracias a la incorporación de la restricción subrogada, suficiente información sobre la estructura del problema original. Además podremos utilizar cualquiera de los métodos conocidos, como por ejemplo optimización subgradiente, a la hora de resolver el nuevo problema.

Hay que de resaltar que una utilización en esta línea de la relajación subrogada abre, además, un amplio abanico de posibilidades a la hora de decidir cual debe ser la restricción subrogada a incorporar a la formulación de la relajación lagrangiana. Evidentemente la incorporación de la 'mejor' restricción subrogada nos llevaría a la obtención de los mejores resultados, pero el cálculo de esta restricción puede resultar computacionalmente elevado. No obstante, pueden encontrar distintas restricciones que, siendo subrogadas en el sentido de que vengan implicadas por el problema original, sin ser computacionalmente costosas de generar, incorporen una valiosa información al problema relajado.

Dos ejemplos de lo anterior son la utilización de los métodos duales propuestos por Barcia en el algoritmo BISA y la modificación propuesta a dicho algoritmo en la sección anterior ya que hemos de tener en cuenta que la restricción que se impone en las relajaciones lagrangianas que se formulan en ambos algoritmos es una restricción subrogada, no ya porque venga implicada por el conjunto original de restricciones, sino porque viene implicada por las condiciones lógicas que se derivan del hecho de saber que cualquier solución posible debe tener un valor de la función objetivo mayor o igual que el de la mejor cota inferior que se conoce.

Como ya se ha mencionado anteriormente, el problema que surge a la hora utilizar en un contexto algorítmico más general restricciones subrogadas es, como en el caso de la relajación lagrangiana, el cálculo de los multiplicadores que generen la mejor restricción. Las aportaciones en este sentido no han sido muy extensas, aunque ya en uno de los primeros trabajos sobre la teoría general Greenberg y Pierskalla [GrPi70] sugieren un método de programación generalizada. Existen además distintos estudios para problemas con una estructura determinada como los de Freville y Plateau [FrPl86] para problemas de Knapsack. En un contexto general, un trabajo de Dyer [Dye80] sugiere dos algoritmos diferentes para el cálculo de coeficientes de la mejor restricción subrogada. Los dos algoritmos se basan en las propiedades que se derivan de la teoría general de la subrogación; el primero de ellos se enmarca en los de programación generalizada, mientras que el segundo tiene una estructura semejante a los de tipo subgradiente.

A continuación exponemos un resumen de dicho trabajo [Dye80] en el que detallamos el segundo de dichos algoritmos que es el que se ha utilizado posteriormente.

Recordando la notación introducida anteriormente, sean los problemas

$$(P) \qquad f^* = \min_{x \in X} f(x)$$

$$h(x) \leq 0,$$

$$(PSu) \quad s(u) = \min_{x \in X(u)} f(x), \qquad \text{siendo } X(u)=\{ x \in X \ / \ uh(x) \leq 0 \}, \text{ y}$$

(DS) $\quad s^* = \sup \{ s(u) / u \geq 0, u \in R^m \}$.

Teniendo en cuenta que $s(ku)=s(u)$ $\forall k>0$ podemos considerar sólamente vectores normalizados u con la ventaja de que pertenecerán a un conjunto compacto. Por lo tanto podemos considerar

(DS) $\quad s^* = \sup \{ s(u) / u \geq 0, ||u||=1, u \in R^m \}$.

En adelante consideraremos que $||.||$ es la norma L^1 aunque los siguientes resultados pueden extenderse fácilmente a cualquier norma.

Definamos también el conjunto compacto $B = \{ u \geq 0 / ||u||=1 \}$

Una primera observación es que aunque siempre se van a alcanzar los valores f^* y $s(u)$, puede ocurrir que s^* no sea igual a ningún valor $s(u)$ como puede fácilmente verse con distintos ejemplos. De hecho, si los problemas no son convexos se necesitan unas condiciones muy fuertes para asegurar que se alcanza s^*.

A continuación se enuncian sin demostración algunas propiedades y resultados que se utilizarán en el diseño del algoritmo para obtener un conjunto de multiplicadores.

Proposición 1: Si (P) es factible, entonces $\forall \varepsilon>0$, el conjunto $W(\varepsilon) = \{ u \in B / s(u) > s^*-\varepsilon \}$ es un subconjunto no vacío, abierto y convexo de B.

La diferencia $f^* - s^*$ se llama gap de dualidad subrogada. Evidentemente la mejor situación es aquella en la que no existe este gap, es decir $f^* = s^*$.

Sea $Y = \{x \in X / h(x) \leq 0\}$ el conjunto de soluciones posibles de (P). El siguiente resultado es inmediato teniendo en cuenta que $Y \subseteq X(u)$ $\forall u$.

Proposición 2 : Se alcanza s^* y es igual a f^* si y sólo si $s(u) = f(x)$ para algún $u \in B$ y $x \in Y$. Entonces, x es la solución óptima de (P) y u la de (DS).

En virtud del resultado anterior podemos considerar el caso de gap infinitesimal como aquel en que $f^* = s^*$, pero s^* no se alcanza para ningún u.

Como ya se ha comentado anteriormente puede ocurrir que s^* no se alcance en B incluso para problemas para los que no hay gap de dualidad. Este no va a ser, sin embargo, el caso de los problemas enteros, que son un caso particular de programas discretos, entendiendo por programa discreto aquel para el que el conjunto $\{f(x) / x \in X\}$ no tiene puntos de acumulación.

Proposición 3: Si (P) es un programa discreto, entonces se alcanza s^* en algún subconjunto abierto y convexo de B.

Notación:

Sea $F(\alpha) = \{x \in X / f(x) \leq \alpha\}$; evidentemente $F(\alpha) \cap X(u) \neq 0$ si y sólo si $s(u) \leq \alpha$.

Sea M(u) el conjunto de soluciones óptimas del problema surrogado (SPu) para u:

$$M(u) = X(u) \cap F(s(u)).$$

Sean $H(\alpha) = h(F(\alpha))$ y $Z(u) = h(M(u))$.

Sean $H^*(\alpha) = \{t \in B / h.t \geq 0 \ \forall h \in G(\alpha)\}$ y $Z^*(u) = \{t \in B / u.t \geq 0 \ \forall u \in Z(u)\}$. Estos conjuntos son los 'polares' de $H(\alpha)$ y $Z(u)$.

Los siguientes resultados dan condiciones de optimalidad para el problema subrogado dual (DS).

Proposición 4: $s(u) > \alpha$ si y sólo si $u \in \text{int } H^*(\alpha)$.

Proposición 5: $\text{int } H^*(\alpha) = x$ si y sólo si existe $v \in \text{conv } \{H(\alpha)\}$ tal que $v \leq 0$.

Proposición 6: s^* es el menor número α para el que existe $v \leq 0$ tal que $v \in \text{conv } \{H(\alpha)\}$.

Proposición 7: Si se alcanza s^* entonces u es un punto máximo de s si y sólo si existe $v \leq 0$ tal que $v \in \text{conv } [H(s(u))]$.

Las proposiciones anteriores proporcionan las condiciones bajo las que puede asegurarse que no va a existir ningún vector de multiplicadores que proporcione un valor de la función s más alto. Deberán, por lo tanto utilizarse como condición de terminación en los algoritmos para el cálculo de multiplicadores.

El algoritmo que se expone a continuación es un algoritmo cuya estructura es semejante a los del tipo subgradiente. La motivación para utilizar una estructura de esta clase es fácil de comprender, dada la eficiencia demostrada de estos algoritmos. El inconveniente que aparece es el de poder construir en cada punto un vector subgradiente, ya que que para la función cuasi-cóncava s(u) que se utiliza en relajación subrogada, dicho vector no tiene por qué existir.

Ahora bien, en cada punto siempre es posible construir un vector, que se denomina cuasi-subgradiente, que es el vector normal al hiperplano de soporte de los conjuntos de nivel; dicho vector será el que se utilice en el algoritmo.

La norma que se utiliza es la norma euclídea $\|.\|_2$. En este caso, el conjunto B'= $\{x\in R^n / \|x\|\leq 1, x\neq 0\}$ es un conjunto convexo contenido en el octante positivo, pero el conjunto B" = $\{x\in R^n / \|x\|=1, x\neq 0\}$ claramente no lo es. A pesar de ello, los resultados teóricos que se han expuesto anteriormente pueden adaptarse para el caso en el que se utilice esta norma, sin más que sustituir la condición de convexidad por la de convexidad esférica.

El siguiente algoritmo genera una sucesión de multiplicadores que pertenecen a B".

Notación:

$$\hat{d}_k = \frac{d_k}{\|d_k\|}$$

Algoritmo

Inicialización

Elegir $u_1 \in$ B"

Hacer $k \leftarrow 1$

$\bar{s} = -\infty$

Fin = falso

Fin Inicialización

Mientras no fin **hacer**

Determinar $x_k \in M(u_k)$

Sean $h_k = h(x_k)$

$f_k = f(x_k) = s(u_k)$

Si $f_k > \bar{w}$ **hacer** $\bar{s} \leftarrow f_k$

Si $h_k \leq 0$ fin=cierto

Calcular u_{k+1}

Hacer $k \leftarrow k+1$

Fin Mientras

En una iteración k determinada el cálculo del vector u_{k+1} es como sigue

Hacer $d_k' \leftarrow h_k - (u_k.h_k)\, u_k$

$d_k \leftarrow \hat{d}_k'$

$u_k' \leftarrow u_k + t_k\, d_k$

$u_k'' \leftarrow u_k' + p_k$

$$\text{donde } p_{ki} = \begin{cases} -2(u_k')_i & \text{si } (u_k')_i < 0 \\ 0 & \text{en otro caso} \end{cases}$$

$u_{k+1} \leftarrow \hat{u}_k''$

Fin

En primer lugar hemos de observar que un algoritmo de este tipo no tiene por qué terminar, ya que nada asegura que vaya a encontrar un vector h_k tal que $h_k \leq 0$. Por este motivo, en la práctica tendremos que establecer algún criterio de parada. El que se ha utilizado es no haber mejorado el valor de $\bar{s}$, mejor valor de $s(u_k)$, en una serie de l iteraciones, siendo l un parámetro fijado arbitrariamente.

El cálculo del vector u_{k+1} tiene una interpretación geométrica que permite asegurar que $u_{k+1} \in B''$, que es la siguiente:

El vector d_k es la dirección cuasi-subgradiente. Este vector permite obtener la dirección de movimiento del vector u_k proyectando el vector h_k en el plano tangente a u_k y normalizando. Hay que notar que esta normalización siempre es posible puesto que si $d_k'=0$ esto implicaría que $h_k \leq 0$. Por lo tanto, primero se modifica u_k con un paso de longitud t_k en la dirección d_k. Esto asegura que u_k' está fuera de la esfera unidad, lo cual es necesario en las demostraciones de convergencia. Al igual que con los procedimientos de tipo subgradiente, es necesario para estas demostraciones que la sucesión de longitudes de paso $\{t_k\}$ esté formada por números reales y sea tal que $\sum t_k = \infty$ y $t_k \rightarrow 0$.

Computacionalmente u_k' es fácil de calcular ya que se trata de una combinación lineal de u_k y h_k. En el caso en el que el punto u_k' esté fuera del octante no negativo de R^m, lo cual es posible, se refleja u_k' en un punto u_k'' perteneciente a dicho octante. En este caso u_k'' será no negativo, pero no pertenecerá a B''. Por ello se normaliza u_k'' para obtener u_{k+1}.

Puede demostrarse que $\|u_k''\|=(1+t_k)^{1/2}$ con lo que la normalización es computacionalmente inmediata.

A continuación se enuncian una serie de resultados sobre la convergencia de la sucesión $\{u_k\}$ que justifican la utilización del algoritmo anterior para el cálculo de los multiplicadores de una restricción subrogada.

Proposición 8 : Lim sup $s(u_k) = s^*$

La proposición anterior proporciona un tipo de convergencia para la sucesión de valores de la función $s(u_k)$. Es natural estudiar también la convergencia, si es que existe alguna, de la sucesión de multiplicadores $\{u_k\}$.

Sea $W(\varepsilon) = \{u \in B \,/\, s(u) > s^* - \varepsilon\}$. Por analogía con la Proposición 1 es fácil ver que $W(\varepsilon)$ es convexo y contiene por lo menos un punto interior respecto a R.

Sea $W^* = (\bigcap_{\varepsilon>0} \mathrm{Cl}\, s(\varepsilon)) \cap B''$.

W^* es no vacío, puesto que los conjuntos $W(\varepsilon)$ están anidados y para cada ε el conjunto $\mathrm{Cl}\,[W(\varepsilon)]$ contiene puntos de B''. Además dado $u \in W^*$, $\forall\varepsilon$ se tiene que cualquier entorno de u contiene un punto que es ε-maximal para s. Las observaciones anteriores permiten demostrar la siguiente

Proposición 9: Todo punto de acumulación de la sucesión $\{u_k\}$ generada por el algoritmo anterior pertenece a W^*.

Corolario 1: Si se alcanza s^* y s es una función continua en cada punto u tal que $s(u) = s^*$, entonces cualquier punto de acumulación u^* de la sucesión $\{u_k\}$ generada por el algoritmo satisface $s(u^*) = s^*$

Corolario 2 : Si s tiene un único punto máximo u^*, entonces $u_k \rightarrow u^*$.

Los corolarios anteriores dan condiciones para que los puntos de acumulación de $\{u_k\}$ maximicen la función s. Sin embargo, en general, sólo se podrá asegurar que, cuando se alcanza s^*, esos puntos están infinitesimalmente próximos a los puntos máximos y, cuando s^* no se alcanza, a los puntos cuasi-óptimos.

Ya se ha comentado anteriormente la necesidad de establecer algún criterio de terminación para el algoritmo. El que se ha sugerido propone su terminación cuando no se haya producido ninguna mejora en un número fijo de iteraciones.

Es natural, por tanto, cuestionarse si existe algún criterio que permita determinar de forma teórica la mejora proporcionada por el algoritmo hacia su convergencia. En cualquier iteración $k \geq k_0$ puede definirse un problema lineal de forma que el valor del óptimo del mismo pueda utilizarse para determinar la proximidad de la convergencia. Sea $J_k = \{ k_0,...,k \}$

El problema lineal es el siguiente:

$$
\text{(I)} \qquad r_k^* = \max y \qquad
\begin{array}{l}
u.h_j - \|h_j\| y \geq 0 \quad j \in J_k \\
\|u\|=1, \\
u \geq 0
\end{array}
$$

En realidad este problema no es más que

$$
r_k^* = \max d(y) \qquad
\begin{array}{l}
d_j(y) \geq 0, \; j \in J_k \\
u \in B''
\end{array}
$$

donde $d_j(y)= d(y,H_j)$ siendo H_j el hiperplano $H_j = \{y \ / \ y.h_j = 0\}$ y $d(.,.)$ indica la distancia euclídea.

<u>Proposición 10</u>: $r_k^* \to 0$ cuando $k \to \infty$.

La proposición anterior nos proporciona un criterio para la terminación del algoritmo, ahora bien, a costa de resolver problemas lineales. Su utilización práctica no parece, por tanto, muy aconsejable.

III.4 Aplicaciones a problemas de Set Partitioning

En las seciones precedentes de este capítulo se han visto, desde una perspectiva general, distintos métodos que se pueden utilizar para reforzar la calidad de las soluciones duales. En esta sección se proponen distintas formulaciones de problemas duales asociados a los problemas (SP), en los que se centra este trabajo, y a los problemas (SPA) debido al interés primordial de los mismos en la resolución de los problemas (SP). Asimismo se estudia la aplicación para estos tipos de problemas de los métodos expuestos anteriormente.

Primero se estudia la aplicación de la técnica de la relajación subrogada tanto para (SP) como para (SPA) puesto que la restricción que así se obtenga se utilizará posteriormente para reforzar la formulación de algunas de las relajaciones lagrangianas que se proponen.

III.4.1 Relajación subrogada

En el caso de problemas enteros puros de set partitioning el problema primal viene dado por

$$
\begin{array}{ll}
(SP) & \min \; cx \\
 & Ax = e \\
 & x_j \in \{0,1\},\ j \in N,
\end{array}
$$

donde $c \in N^{n+}$, A es una matriz de dimensión mxn y e es el vector m-dimensional formado por todo unos.

Teniendo en cuenta que el algoritmo de Dyer [Dye80] para obtener una restricción subrogada que se ha expuesto en III.3 considera los problemas de optimización expresados en forma canónica en los que las restricciones aparecen siempre como desigualdades de signo "$\leq$" será preciso expresar (SP) mediante alguna formulación equivalente que permita aplicar el algoritmo anteriormente descrito. En particualr podemos reescribir (SP) como

$$
\begin{array}{ll}
(SP) & \min \; cx \\
 & Ax \geq e \\
 & nx \leq m \\
 & x_j \in \{0,1\},\ j \in N,
\end{array}
$$

que expresado en forma canónica resulta

$$
\begin{aligned}
(SP)\quad & \min\ cx \\
& -Ax + e \leq 0 \\
& nx - m \leq 0 \\
& x_j \in \{0,1\},\ j \in N,
\end{aligned}
$$

El problema subrogado para (SP), asociado a un vector de multiplicadores (u,v) será

$$
\begin{aligned}
(PSuv)\quad & \min\ cx \\
& u\,(-Ax+e) + v\,(nx-m) \leq 0 \\
& x_j \in \{0,1\},\ j \in N
\end{aligned}
$$

o equivalentemente

$$
\begin{aligned}
(PSuv)\quad & \min\ cx \\
& (uA-vn)\,x \geq ue - vm \\
& x_j \in \{0,1\},\ j \in N
\end{aligned}
$$

El problema dual subrogado será

$$
(DS)\qquad \max_{\substack{u\geq 0\\ v\geq 0}}\ v(PSu) = \max_{\substack{u\geq 0\\ v\geq 0}}\ \min_{x\in X(u)} cx
$$

siendo $X(u,v) = \{x \,/\, (uA-vn)x \geq ue-vm,\ x_j \in \{0,1\},\ j \in N\}$

En este caso, dado un vector (u,v) el conjunto M(u,v) vendrá dado por el conjunto de las soluciones óptimas del problema (PSuv).

En el caso de problemas de set partitioning ampliados con restricciones de tipo set covering

$$
\begin{aligned}
(SPA)\quad & \min\ cx \\
& Ax = e_m \\
& Gx \geq e_r \\
& x_j \in \{0,1\},\ j \in N,
\end{aligned}
$$

donde $c \in Z^{n+}$, A y G son matrices de dimensiones mxn y rxn, respectivamente, y $e_m \in R^m$ y

$e_r \in R^r$ son los vectores formados por todo unos.

La formulación equivalente que permite expresar estos problemas en forma canónica es la siguiente:

$$
\begin{array}{ll}
\text{(SPA)} & \min\ cx \\
& Ax \geq e_m \\
& Gx \geq e_r \\
& nx \leq m \\
& x_j \in \{0,1\},\ j \in N,
\end{array}
$$

que expresado en forma canónica resulta

$$
\begin{array}{ll}
\text{(SPA)} & \min\ cx \\
& -Ax + e_m \leq 0 \\
& -Gx + e_r \leq 0 \\
& nx - m \leq 0 \\
& x_j \in \{0,1\},\ j \in N,
\end{array}
$$

Por consiguiente, el problema subrogado para (SPA), asociado a un vector de multiplicadores (u,w,v) será

$$
\begin{array}{ll}
\text{(PASuwv)} & \min\ cx \\
& u(-Ax+e_m) + w(-Gx+e_r)+v(nx-m) \leq 0 \\
& x_j \in \{0,1\},\ j \in N
\end{array}
$$

o equivalentemente

$$
\begin{array}{ll}
\text{(PASuwv)} & \min\ cx \\
& (uA + wG - vn)\,x \geq ue_m + we_r - vm \\
& x_j \in \{0,1\},\ j \in N
\end{array}
$$

El problema dual subrogado será

$$
\text{(DSA)} \quad \max_{\substack{u\geq 0\\ w\geq 0\\ v\geq 0}} v(PSu) = \max_{\substack{u\geq 0\\ w\geq 0\\ v\geq 0}} \ \min_{x\in X(u)} cx
$$

siendo $X(u,w,v) = \{x \,/\, (uA + wG - vn)x \geq ue_m + we_r - vm,\ x_j \in \{0,1\},\ j \in N\}$

Ahora, dado un vector (u,w,v) el conjunto M(u,w,v) vendrá dado por el conjunto de las soluciones óptimas del problema (PASuwv).

Teniendo en cuenta que (SP) puede considerarse como el caso particular de (SPA) en el que la matriz G es nula, a continuación se expone el algoritmo de Dyer [Dye80] de cálculo de multiplicadores para una restricción subrogada para el caso general de problemas de la forma (SPA).

Algoritmo

Inicialización

Elegir (u_1,w_1,v_1) tal que

i) $u_1 \geq 0,\ w_1 \geq 0,\ v_1 \geq 0$

ii) $\|(u_1,w_1,v_1)\| = 1$

Fijar kmax *número máximo de iteraciones que se permite iterar sin mejorar $\bar{s}$ antes de terminar el algoritmo*

Hacer $k=1$

$\bar{s} = -\infty$

Fin = falso

Fin Inicialización

Mientras no fin **hacer**

Hallar $x_k \in M(u_k,w_k,v_k)$ resolviendo el problema

(PSuv) min cx

$(u_kA + w_kG - v_kn)\, x \geq u_ke_m + w_ke_r - v_km$

$x_j \in \{0,1\},\ j \in N$

Sean $h_k = h(x_k)$ cuyas componentes vienen dadas por

$$h_{ki} = \begin{cases} -a_i.x_k+1 & i=1,\ldots,m \\ -g_i.x_k+1 & i=m+1,\ldots,m+r \\ n.x_k-m & i=m+r+1 \end{cases}$$

$f_i = f(x_i) = s(u_i,w_i,u_i) = c.x_k$

Si $c.x_k > \bar{s}$ **entonces**

$\bar{s} \leftarrow c.x_k$

$\bar{k} \leftarrow k$

Fin si

Si $-Ax_k + e_m \leq 0$ **y**
$-Gx_k + e_r \leq 0$ **y**
$nx_k - m \leq 0$ **entonces**

fin = cierto *La solución es posible.*

En otro caso si $k - \bar{k} \geq kmax$ **entonces**

fin = cierto *Sobrepasa número máximo de iteraciones sin mejorar.*

Fin si

Calcular u_k+1

Hacer $k \leftarrow k+1$

Fin Mientras

En una iteración determinada el cálculo del vector u_{k+1} se realiza como ya se ha expuesto en el algoritmo general. La sucesión de longitudes de paso que tomamos es $\{1/k\}$ que cumple las condiciones impuestas anteriormente.

Queda abierta la cuestión de qué vector (u_1,w_1,v_1) inicial elegir. Una primera opción es la de tomar el vector unitario m+r+1-dimensional normalizado. Otra posible elección, que también se ha probado, es la de utilizar como m primeras componentes las provinientes de una solución dual (u,w) ya conocida (bien por la aplicación de una heurística, bien por la utilización de relajación lagrangiana), haciendo 0 la última componente y normalizando el vector (u,w,0) así obtenido.

Hay que resaltar, una vez más, que el algoritmo que se acaba de describir es aplicable de forma inmediata a problemas (SP) de set partitioning puros. Bastará con considerar r=0 con lo que no aparecerá la parte correspondiente a la componente w en el vector de multiplicadores (u,w,v), desapareciendo además todas las referencias a la matriz G.

III.4.2 Relajación Lagrangiana

En este apartado nos centraremos en algunas de las posibles relajaciones lagrangianas para problemas de minimización de set partitioning.

Sea el problema

$$
\begin{array}{ll}
(SP) & \min\ cx \\
 & Ax = e \\
 & x_j \in \{0,1\},\ j \in N,
\end{array}
$$

donde, una vez más, $c \in Z^{n+}$, A es una matriz de dimensión mxn y $e \in R^m$ es el vector formado por todo unos.

La primera relajación que se presenta es aquella que consiste en incorporar a la función objetivo todo el conjunto de restricciones, excepto las de integridad y de acotación, resultando un problema relajado de la forma

$$
(RL1u) \qquad \min_{x \in X}\ cx + u(e-Ax)
$$

siendo $X = \{x \in R^n / x_j \in \{0,1\}, j \in N\}$

Esta relajación lagrangiana, que es la ordinaria, genera problemas que tienen una estructura muy sencilla y cuya solución, dado un vector u de multiplicadores es inmediata y viene dada por

$$
x_j = \begin{cases} 1 & \text{si } (c-uA)_j \leq 0 \\ 0 & \text{si } (c-uA)_j > 0 \end{cases}
$$

Desgraciadamente, las cotas inferiores obtenidas con esta formulación, a pesar de que en todos los casos son mejores que las proporcionadas por la heurística, siguen estando excesivamente alejadas del valor del óptimo. Esto es así debido al hecho que la única información que contiene el problema (RL1) sobre el problema (P) es que las soluciones deben ser 0 ó 1. Claramente, y a pesar de que las restricciones originales están implícitamente consideradas en la función objetivo, esta formulación resulta excesivamente relajada.

Consideremos ahora la siguiente formulación para el problema (P)

(P1) $\min cx$

$Ax \geq e$

$nx \leq m$

$x_j \in \{0,1\}, j \in N$

donde n es un vector n-dimensional cuyas componentes n_j denotan el número de elementos de la columna j de la matriz A, es decir $n_j = |N_j|$ $\forall j$, y $m = |M|$ es el número total de restricciones.

Evidentemente se trata de una formulación equivalente para (P) ya que cualquier solución de (P) satisface las restricciones de (P1), en particular las cumple todas como estricta igualdad y, recíprocamente, cualquier solución de (P1) es solución para (P) ya que si $Ax > e \Rightarrow$ (sumando todas las desigualdades) $nx > m$, lo cual es absurdo puesto que $nx \leq m$.

A partir de esta formulación equivalente de (P) podemos construir la siguiente Relajación Lagrangiana

(RL2u) $\min\limits_{x \in X} cx + u(e-Ax)$

$nx \leq m$

siendo $X = \{x \in R^n \,/\, x_j \in \{0,1\}, j \in N\}$

En este caso el vector de multiplicadores u estará restringido en signo al tratarse de desigualdades las restricciones que hemos incorporado a la función objetivo.

Ahora bien, teniendo en cuenta que la restricción $nx \leq m$ resulta redundante respecto a la formulación original del problema (P) podemos considerar dicho problema como

(P) $\min cx$

$Ax = e$

$nx \leq m$

$x_j \in \{0,1\}, j \in N,$

con lo que si incorporamos a la función objetivo el conjunto original de restricciones de igualdad y manteniendo como restricción $nx \leq m$ obtenemos como relajación lagrangiana la siguiente:

$$\text{(RL3u)} \qquad \min_{x \in X} \; cx + u(e-Ax)$$

$$nx \leq m$$

siendo $X = \{ x \in R^n \,/\, x_j \in \{0,1\}, j \in N \}$

La única diferencia formal entre las relajaciones lagrangianas anteriores (RL2u) y (RL3u) es la formulación equivalente del problema original (P) a partir de la que se obtienen. Ello da lugar a que en (RL2u) lel vector de multiplicadores u esté restringido a ser no negativo ya que las restricciones incorporadas son $Ax \geq e$, mientras que en (RL3u) el vector de dichos multiplicadores u no esté restringido en signo ya que las restricciones incorporadas a la función objetivo son $Ax = e$. Sin embargo, veremos que esta diferencia en el signo del vector u no da lugar a resultados claramente diferentes.

Esto se debe al hecho de que al resolver los problemas duales Lagrangianos

$$\text{(D2)} \qquad \max_{u \geq 0} \min_{x \in X} \; cx + u(e-Ax)$$

$$nx \leq m,$$

$$\text{(D3)} \qquad \max_{u} \min_{x \in X} \; cx + u(e-Ax)$$

$$nx \leq m$$

los dos problemas penalizan en la función objetivo vectores x tales que $Ax<e$ (en (D3) también se penaliza $Ax>e$), mientras que por otro lado la restricción favorece que $Ax \leq e$.

Existe, por el contrario, una diferencia de interpretación en las formulaciones que dan lugar a las ralajaciones (RL2u) y (RL3u):

En el primer caso se trata de una relajación lagrangiana ordinaria en el sentido de que la restricción $nx \leq m$ es parte de las restricciones de 'caracterizan' el problema (P1) y simplemente decidimos no relajar dicha restricción. En el caso de (RL3u), la restricción $nx \leq m$ no es una de las restricciones que caracterizan (P), ya que se trata de una restricción redundante; es, por lo tanto, una restricción subrogada puesto que viene implicada por el conjunto original de restricciones.

En este sentido, la relajación (RL3u) puede entenderse como una relajación lagrangiana que viene reforzada por la incorporación de una restricción subrogada. La similitud de los resultados a los que dan lugar estas dos relajaciones hace que, en este caso, resulte meramente especulativo incidir en la diferencia de estas dos formulaciones.

Si consideramos de nuevo la relajación lagrangiana ordinaria (RL1u), podemos planteamos ahora qué restricciones subrogadas incorporar a esta relajación de forma se refuerce la misma, añadiendo información relevante sobre el problema original y de forma que el problema resultante resulte sencillo de resolver. Una primera observación, que hace esta aproximación sugerente, es que al añadir una única restricción en forma de desigualdad, los problemas resultantes serán del tipo knapsack para los que existen, como ya se ha dicho, distintos algoritmos muy conocidos y utilizados que los resuelven de forma muy eficiente [MaTo79, GiGo61].

En el contexto anterior, hemos estudiado tres tipos distintos de restricciones subrogadas. Como ya se ha mencionado, una de ellas, es la forma $nx \leq m$. Otra es la restricción $cx \geq s_0$ que se utiliza de una forma similar a la de los métodos de refuerzo dual sugeridos por Barcia {Barci85a, Barci85b] dando lugar al problema relajado

$$(\text{RL4u}) \qquad \min_{x \in X} cx + u(e-Ax)$$
$$cx \geq s_0$$

siendo s_0 el entero inmediatamente superior al de la mejor cota inferior que conocemos y $X=\{x \in R^n \,/\, x_j \in \{0,1\}, j \in N\}$.

La tercera de las restricciones estudiadas es la restricción subrogada $gx \geq g_0$ obtenida mediante el algoritmo de cálculo de los multiplicadores de Dyer [Dye80] que ya se ha expuesto, dando lugar a un problema de la forma

$$(\text{RL5u}) \qquad \min_{x \in X} cx + u(e-Ax)$$
$$gx \geq g_0$$

El criterio que ha llevado a elegir precisamente estas tres restricciones es que cada una de ellas resulta representativa de alguna de las propiedades que se consideran deseables a la hora de generar restricciones subrogadas.

La restricción $nx \leq m$ resulta inmediata de generar, con lo que computacionalmente su utilización resulta rentable y, además, la información parcial que proporciona sobre el conjunto original de restricciones resulta interesante.

En concreto, las restricciones $Ax = e$ significan que cualquier solución del sistema de ecuaciones debe ser tal que las componentes que estén fijadas a 1 tengan asociadas vectores columna a^j que deben de ser ortogonales a todos los vectores asociados a las demás componentes fijadas a 1. Es evidente que la restricción que se incorpora no implica esta condición de ortogonalidad, pero facilita la elección de soluciones que sí la cumplan. A pesar de que $Ax = e$ implica $nx = m$ que es una igualdad tan sencilla de generar como $nx \leq m$ y resulta más restrictiva, el motivo por el que se toma la restricción de desiguladad es que la estructura del problema resultante resulta así más fácil de resolver.

La restricción $cx \geq s_0$, donde s_0 es el entero inmediatamente superior a la mejor cota inferior de la que disponemos, es asimismo inmediata de generar. La información que suministra esta restricción no procede del conjunto original de restricciones, por lo que no refuerza la relajación lagrangiana en el sentido de aproximarnos a la estructura del problema original. En este caso, la información que se obtiene se deriva de las condiciones lógicas que pueden imponerse a partir del conocimiento actual del problema. En particular, la condición de superar el valor del entero inmediatamente superior a la mejor cota inferior conocida resulta intuitivamente evidente, pero su inclusión en una formulación impide la elección de soluciones de las que, a priori, sabemos no son factibles.

Al contrario de las dos restricciones anteriores, la restricción $gx \geq g_0$ obtenida mediante el algoritmo de Dyer resulta computacionalmente costosa de obtener, ya que para ello deberemos utilizar un algoritmo iterativo, bien hasta obtener una solución posible, bien hasta que no produzca mejora en un número determinado de iteraciones. Sin embargo, está plenamente justificada por tres motivos:

El primero es que, como ya se ha visto anteriormente, el algoritmo para obtenerla genera una sucesión de cotas inferiores cuyos valores pueden ser mejores, a partir de una cierta iteración, que las cotas que tengamos, con lo que conseguiremos reducir el gap.

El segundo, es que, en una de las iteraciones de dicho algoritmo, podemos obtener eventualmente una solución posible; si esto ocurre habremos encontrado el óptimo del problema original.

Finalmente, aunque no se consiga ninguna de las posibilidades anteriores, la restricción obtenida mediante este algoritmo, resume, por la teoría que lo sustenta, la mayor información posible sobre el conjunto original de restricciones.

A pesar de que las relajaciones lagrangianas anteriores han proporcionado resultados sucesivamente mejores, incluso con la más potente de todas ellas (RL6u) no se ha conseguido reducir el gap de forma notable. Por este motivo se ha estudiado la formulación de otras relajaciones lagrangianas que resulten aún más restringidas, incorporando simultáneamente dos restricciones subrogadas diferentes. Las dos restricciones a incorporar se han elegido de forma que una de ellas aporte información sobre la estructura original del problema y la segunda sobre las condiciones lógicas que se derivan del conocimiento actual del problema.

Este criterio nos ha llevado a la formulación de las dos siguientes relajaciones lagrangianas

(RL6u) $$\min_{x \in X} cx + u(e-Ax)$$
$$nx \leq m$$
$$cx \geq s_0$$

(RL7u) $$\min_{x \in X} cx + u(e-Ax)$$
$$gx \geq g_0$$
$$cx \geq s_0$$

En ambas, la segunda restricción impone que las soluciones tengan un valor de la función objetivo por lo menos tan alto como el de la mejor cota inferior que conocemos. En (RL6u) la primera restricción $nx \leq m$ pretende acercarnos a la condición de que las componentes de las soluciones del problema original sean ortogonales, mientras que en (RL7u) hemos impuesto la restricción subrogada obtenida a partir del algoritmo de Dyer [Dye80].

Es evidente que la obtención de la primera restricción es inmediata en el caso de (RL6u), mientras que en el caso de (RL7u) requiere un esfuerzo computacional considerable. También lo es el hecho de que la información suministrada por $gx \geq g_0$ es en general mucho más relevante del problema que queremos resolver debido a la teoría de la subrogación que sustenta su utilización.

Las dos relajaciones anteriores tienen una estructura que ya no es la de un problema clásico de knapsack. En concreto, tanto (RL6u) como (RL7u) son problemas de knapsack con dos restricciones. Ahora bien, mientras que en (RL6u) las dos restricciones tienen la desigualdad del mismo tipo en el caso de (RL5u) los signos de las desigualdades son diferentes. La estructura resultante de las dos relajaciones anteriores nos llevará, en el siguiente capítulo, a estudiar con detalle los problemas de knapsack con dos restricciones e intentar diseñar algoritmos para resolverlos eficientemente.

Una vez consideradas algunas de las posibles relajaciones lagrangianas para problemas de set partitioning, estamos interesados en considerar posibles relajaciones lagrangianas para los problemas ampliados de la forma (SPA). Esto es así, ya que, como se ha visto en II.3, el problema (SPA) que se obtiene al incorporar de forma sucesiva a (SP) las desigualdades válidas que se obtienen a partir de las cotas condicionales resulta equivalente, en cuanto a las soluciones óptimas, al problema original.

Además, si la información añadida a los problemas originales mediante la incorporación de estas desigualdades de tipo set covering resulta valiosa, no debe resultar sorprendente que dicha información se refleje de forma satisfactoria al tenerse en cuenta en las distintas relajaciones lagrangianas que se formulen para el problema original.

En particular dado un problema ampliado de la forma

$$
\text{(SPA)} \qquad \begin{array}{l} \min\ cx \\ Ax = e_m \\ Gx \geq e_r \\ x_j \in \{0,1\},\ j \in N \end{array}
$$

donde, como siempre, $c \in R^n$, dimensiones mxn y rxn respectivamente y $e_m \in R^m$ y $e_r \in R^r$ son vectores formados por todo unos.

Para un conjunto de multiplicadores (u,v), donde u es un vector m-dimensional y v un vector no negativo r-dimensional, la relajación lagrangiana ordinaria asociada al problema (SPA) viene dada por:

$$
\text{(RL8uv)} \qquad \min_{x \in X}\ cx + u(e_m - Ax) + v(e_r - Gx)
$$

siendo $X = \{x \in R^n \,/\, x_j \in \{0,1\},\ j \in N$.

Ahora, parte del vector de multiplicadores estará restringido en signo ya que viene asociado a las restricciones de tipo set covering que se han incorporado a (SP) cuyo signo es "$\geq$". Los multiplicadores asociados al conjunto original de restricciones, como antes, no están restringidas en signo.

Podemos reforzar esta formulación de relajación lagrangiana básica asociada a problemas del tipo (SPA) teniendo en cuenta todas las consideraciones que se han hecho previamente para los problemas de (SP) puros. Las restricciones que se incorporen a esta formulación serán las que se han mencionado previamente: $nx \leq m$, la restricción subrogada $gx \geq g_0$ y $cx \geq s_0$. Los criterios que se utilicen para incorporar una o varias de estas restricciones a la formulación básica seguirán siendo los mismos puesto que no varía el significado de las mismas.

A continuación se propone una relajación lagrangiana asociada a los problemas (SPA) que resulta especialmente sugerente y que no es una generalización de las ya mencionadas anteriormente. Se trata de mantener de forma explícita la información referente al último plano secante obtenido por el procedimiento de generación de cortes, mientras que se incorporan a la función objetivo el resto de las restricciones del problema.

Consideremos por lo tanto un problema de tipo (SPA) para el que se ha obtenido una desigualdad válida de la forma $\pi x \geq 1$. Dado un vector de multiplicadores (u,v) tal que $v \geq 0$, una posible relajación lagrangiana para el problema resultante es la siguiente:

$$(RL9uv) \qquad \min_{x \in X} cx + u(e_m - Ax) + v(e_r - Gx)$$
$$\pi x \geq 1$$

Teniendo en cuenta que este problema puede escribirse como

$$(RL9uv) \qquad ue_m + ve_r + \min_{x \in X} (c - uA - vG)x$$
$$\pi x \geq 1$$

Sean $J_1 = \{j \in N \,/\, \pi_j = 1\}$ y $J_2 = \{j \in N \,/\, (c - uA - vG)_j \leq 0\}$.

La solución a (RL9uv) es inmediata y viene dada por

Si $J_1 \cap J_2 \neq \emptyset$ entonces,

$$x_j = \begin{cases} 1, & \text{si } (c - uA - vG)_j \leq 0 \\ 0, & \text{en otro caso} \end{cases}$$

Si $J_1 \cap J_2 = \emptyset$, sea $j_1 \in J_1$ t.q. $(c - uA - vG)_{j1} = \min_{j \in J_1} (c - uA - vG)_j$. Entonces,

$$x_j = \begin{cases} 1, & \text{si } (c - uA - vG)_j \leq 0 \\ 1, & \text{si } j = j_1 \\ 0, & \text{en otro caso} \end{cases}$$

El interés de esta relajación radica en el hecho de que los problemas resultantes resultan muy sencillos de resolver y además en que mantiene de forma explícita en la formulación el último de los planos secantes generados, que puede considerarse como una restricción subrogada en el sentido de que viene implicada por las condiciones lógicas que se derivan del conocimiento actual del problema. En ese sentido, puede considerarse que se alcanzan los dos objetivos que se han perseguido en las formulaciones de posibles relajaciones lagrangianas. Evidentemente, esta formulación resultará tanto más adecuada en la medida que el corte añadido resulte más potente.

CAPÍTULO IV

PROBLEMAS DE KNAPSACK CON DOS RESTRICCIONES

En este capítulo se estudian problemas de knapsack con dos restricciones. Estos problemas son un caso particular de programas de multi-knapsack (programas enteros puros en los que el conjunto de restricciones está formado por una familia de inecuaciones), en los que el conjunto de restricciones consta de dos desigualdades. Los problemas de multi-knapsack reciben esta denominación por considerarse una generalización de los, ya clásicos, problemas de knapsack en los que el conjunto de restricciones está formado por una única inecuación. A diferencia de los clásicos problemas de knapsack, que han sido ampliamente estudiados, tanto desde un enfoque poliédrico [CrJo83] como desde un punto de vista algorítmico [MaTo79, GiGo61], el interés que han suscitado los problemas de knapsack con más de una restricción ha sido bastante limitado, aunque existen distintos trabajos en los que se estudian estos problemas en el caso que todas las desigualdades tengan el mismo signo desde una perspectiva general [Ema86, FrPl86, GaPi85, Shih79].

Este estudio, en el que nos limitaremos a algunos problemas con dos restricciones, viene motivado, como ya se ha mencionado anteriormente, por la aparición de los mismos en algunas de las relajaciones lagrangianas que se han formulado en el capítulo anterior.

El interés de los problemas de knapsack con dos restricciones no se limita al contexto en el que han aparecido en este trabajo. Por ejemplo, en los problemas de localización de plantas con restricciones de capacidad [BaFeJö86,BaJöMi87], algunas de las relajaciones lagrangianas, que se obtienen al incorporar a la función objetivo distintas familias de restricciones, tienen una estructura de problemas de knapsack con dos restricciones, que coincide con alguno de los dos tipos de problemas que se han estudiado.

El objetivo que se persigue es el diseño de algún algoritmo eficiente para su resolución, que aproveche al máximo la estructura particular de los mismos. En particular, se van a estudiar dos clases de problemas; en la primera clase cada una de las restricciones es una desigualdad de distinto sentido, mientras que en la segunda, las dos inecuaciones son del mismo sentido.

Los trabajos existentes que estudian los problemas de multi-knapsack no resultan adecuados en el contexto de los problemas que queremos resolver fundamentalmente por dos motivos: el primero es de carácter numérico mientras que el segundo incide en la importancia de considerar la estructura particular de cada tipo de problema.

Desde una perspectiva general los trabajos que estudian los problemas de multi-knapsack imponen condiciones sobre los valores numéricos de los coeficientes que resultan excesivamente restrictivas en el contexto de este trabajo. En concreto suelen imponerse restricciones de integridad sobre los coeficientes, así como restricciones de no negatividad sobre los mismos. Además, se supone que las restricciones que aparecen son todas del mismo sentido. Esta restricción numérica en los coeficientes de los problemas y en el tipo de desigualdades que aparecen no es superable, como se explicará posteriormente, en el contexto de los problemas que queremos resolver.

En otro orden de cosas, y debido a que el conjunto de restricciones en los problemas que se estudian a continuación está formado por dos únicas desigualdades, parece además conveniente intentar obtener el mayor rendimiento posible de la estructura específica que presentan estos dos tipos de problemas. Ello se debe a que los procedimientos generales que existen para problemas enteros de programación matemática resultan excesivamente vagos en este contexto.

De los dos tipos de problemas que estudiaremos en este capítulo los de la primera clase son de la forma

$$
\begin{array}{ll}
\text{(KP1)} & \min dx \\
 & cx \geq c_0 \\
 & gx \leq g_0 \\
 & x_j \in \{0,1\},\ j \in N,
\end{array}
$$

donde c, g $\in R^{n+}$, d $\in R^n$; mientras que los de la segunda son

$$
\begin{array}{ll}
\text{(KP2)} & \min dx \\
 & cx \geq c_0 \\
 & gx \geq g_0 \\
 & x_j \in \{0,1\},\ j \in N,
\end{array}
$$

donde c,g $\in R^n$, d $\in R^{n+}$ y c_0, $g_0 \in R^+$, siendo N el conjunto de índices de las variables.

La estructura de las relajaciones lagrangianas mencionadas era

$$(RL6u) \quad \min \; cx + u(e-Ax)$$
$$nx \leq m$$
$$cx \geq s_0$$
$$x_j \in \{0,1\},\; j \in N$$

$$(RL7u) \quad \min \; cx + u(e-Ax)$$
$$gx \geq g_0$$
$$cx \geq s_0$$
$$x_j \in \{0,1\},\; j \in N.$$

Veremos más adelante que (RL6u) es un caso particular de (KP1) y que (RL7u) puede transformarse fácilmente en un caso particular de (KP2). Por lo tanto, la clase de problemas que se estudia es suficiente para el objetivo de las relajaciones que se pretenden resolver.

Como ya se ha comentado anteriormente, los problemas de knapsack con varias restricciones en los que todas ellas son desigualdades del mismo signo han sido ya estudiados [Ema86, FrPl86, GaPi85, Shih79].

Las restricciones de integridad que suelen imponerse sobre los coeficientes pueden teóricamente superarse multiplicando todos ellos por una potencia de 10 suficientemente grande. Sin embargo, se trata de una solución que presenta inconvenientes, ya que cuando se requiera una precisión de varios decimales para la parte no entera de los coeficientes, el factor que deber aplicarse a los mismos resulta excesivamente grande dando lugar a problemas de almacenamiento. Si, por el contrario, limitamos la magnitud del factor a aplicar, el problema resultante no es un problema equivalente al problema original dando lugar a resultados erróneos, especialmente cuando, como en este caso, se resuelve iterativamente una sucesión de problemas de este tipo en los que, de una iteración a otra, algunos coeficientes variarán sólo ligeramente y además el reconocimiento de ciertas propiedades numéricas resulta determinante para la identificación de la solución óptima.

Las restricciones de no negatividad que se imponen a los coeficientes resultan no superables en el ámbito de este trabajo. Tengamos en cuenta que, cuando una de las variables tenga un coeficiente negativo, sólo será posible hacer una transformación a un problema equivalente en el caso de que los otros dos coeficientes que aparecen en el problema también sean negativos. Evidentemente, se trata de una restricción excesiva que en este caso no se puede garantizar.

Este capítulo se estructura en 4 partes; en la primera se justifica el enfoque algorítmico utilizado, en la segunda y tercera, se estudian los problemas (KP1) y (KP2) respectivamente, presentando cotas inferiores y algoritmos de enumeración implícita para cada uno de dichos problemas. Finalmente, en la cuarta, se estudia la aplicación de estas técnicas a la resolución de las relajaciones lagrangianas formuladas anteriormente y se definen heurísticas ligadas a dichas formulaciones.