

Dualidad en el Problema de Localización Discreto Simple.

José A. Moreno Pérez.

Departamento de Estadística, Investigación Operativa y Computación.
Universidad de La Laguna.

Para la revista de la
Academia Canaria de Las Ciencias.

Mayo de 1995.

ABSTRACT

In this paper we show how several duality tools are useful in the solution of the Simple Discrete Location Problem. This is the public-sector discrete version of the most important location-allocation problem, known by SPLP or UFLP. After a simple model introduction and using a very powerful standard notation, we establish the most useful formulations and linear relaxations. The standard, the reduced, the condensed and lagrangean dual problems are considered. We analyze the role of the duality in the main solution procedures: subgradient optimization, decomposition method and canonical reduction. We show how the dual ascent heuristic procedures, known as the most successful methods for these problems, allow us to consider the problem from the heuristic point of view. Finally we give some recommendations for the practical solving of a real problem with moderate size.

RESUMEN

En el presente trabajo se analiza la utilidad de las distintas formas de dualidad en la resolución del Problema de Localización Discreto Simple (PLDS). Este problema es la versión discreta para el sector público del problema de localización-asignación más importante, conocido internacionalmente por SPLP o UFLP. Tras una sencilla introducción del modelo simple y, utilizando una notación estándar muy potente, se establecen las formulaciones más útiles y las relajaciones lineales. Se establecen los distintos problemas duales: estándar, reducido, condensados y lagrangiano. Se analiza el papel de la dualidad en los principales procedimientos de solución: el método del subgradiente, el método de descomposición y la reducción canónica. Se muestra como los procedimientos duales ascendentes, reconocidos como los de mayor éxito para estos problemas, permiten abordar el PLDS desde la perspectiva heurística. Finalmente se ofrecen algunas recomendaciones al afrontar la resolución práctica de un problema real de dimensiones moderadas.

Palabras Clave: Localización, Optimización Combinatoria. Programación Matemática.

1. Introducción

Los **Problemas de Localización** aparecen al organizar sobre un determinado espacio la prestación de un servicio. Son problemas en los que hay que optimizar la localización o ubicación de los puntos de servicio y algún otro aspecto de la organización del servicio. En los problemas de localización-asignación hay que determinar exclusivamente la ubicación de los puntos de servicio y la forma en que se asignan cada una de las posibles demandas a tales puntos. En los problemas que surgen en el sector público, el objetivo consiste en minimizar los costes globales que se producen (ver Thisse y Zoller (1983)). En los modelos más simples las asignaciones se pueden realizar de forma directa, asignando toda demanda al punto de servicio que lo atiende con menor coste. El problema simple estándar aparece con Kuehn y Hamburger (1963) y Balinski (1965) y es conocido en la literatura internacional por *Simple Plant Location Problem* (SPLP), o por *Uncapacitated Facility Location Problem* (UFLP) (ver Krarup y Pruzan (1983) y Cornuejols et al. (1990)). Si las posibilidades se encuentran dentro de un conjunto finito se trata de un modelo discreto. Al problema de optimización combinatoria correspondiente al modelo simple discreto lo denominamos **Problema de Localización Discreto Simple** (PLDS).

1.1. El modelo discreto simple.

Los elementos básicos del PLDS son: el conjunto $L = \{l_j : j \in J\}$ de m **puntos de localización**, y el conjunto $D = \{d_i : i \in I\}$ de n **puntos de demanda**. Los gastos que origina la prestación de un servicio suelen tener dos componentes, una cantidad fija inicial, que depende de la ubicación del servicio, y una cantidad proporcional a la demanda atendida, donde la constante de proporcionalidad depende de la situación relativa entre el punto donde se produce la demanda y de la ubicación del punto desde donde se le presta el servicio. A partir del estudio de estos gastos se obtienen el coste c_j asociado a la ubicación de un punto de servicio en el punto de localización l_j y el coste c_{ij} asociado a la asignación de toda la demanda producida en el punto de demanda d_i a un punto de servicio ubicado en l_j . Por tanto, los datos que especifican el problema vienen dados por dos conjuntos de costes no negativos: los **costes de localización** $c_J = (c_j : j \in J)$, y los **costes de asignación** $c_{IJ} = (c_{ij} : i \in I, j \in J)^*$. Desde el punto de vista formal, los datos del problema vienen dados por los costes: $c = (c_J, c_{IJ})$.

La hipótesis de que los puntos de servicio pueden tener una capacidad ilimitada permite hacer dos observaciones que simplifican las alternativas: (i) cada punto de demanda se asigna a un único punto de servicio que atenderá toda su demanda, y (ii) una vez establecida la ubicación de los puntos de servicio, cada punto de demanda se asigna al punto de servicio de menor coste. Por tanto, cada una de las soluciones **alternativas** se reduce a seleccionar el conjunto S de los índices de los puntos de localización donde ubicar puntos de servicio. El coste global asociado a cada una de estas alternativas es:

$$c(S) = \sum_{j \in S} c_j + \sum_{i \in I} \min_{j \in S} c_{ij}.$$

El PLDS consiste en encontrar la solución alternativa **óptima** determinada por el conjunto $S^* \subseteq J$ tal que:

$$(1) \quad c(S^*) = \min_{S \subseteq J} c(S).$$

El PLDS es un problema de optimización combinatoria NP-**duro** al tener como caso particular el problema del cubrimiento de un grafo por vértices (ver, por ejemplo, Cornuejols et al. (1990)).

1.2. Formulación del problema.

Para expresar el problema (1) como un problema de optimización combinatoria o problema de programación 0-1 las soluciones alternativas se especifican por las variables de decisión binarias. En el PLDS se consideran las variables de decisión binarias $x_j, \forall j \in J$, que especifican los puntos de localización seleccionados

* El emplear una sola letra c para ambos tipos de costes simplifica la notación y no tiene por que dar lugar a confusión puesto que el número de índices diferencia ambos costes: c_j, c_{ij} . Mantener la letra c para todos los costes, la letra x para todas las variables de decisión y la letra u para todas las variables duales permite ajustarnos a la notación estándar en los problemas de programación lineal. Al poner como subíndice un conjunto de índices se denota al conjunto o vector de los correspondientes elementos subindicados por cada uno de los índices del conjunto; $a_I = (a_i : i \in I)$.

para ubicar puntos de servicio en ellos. Estas variables, llamadas **variables de localización**, vienen definidas por ^{*}:

$$x_j := [j \in S] = [l_j \text{ es seleccionado}], \forall j \in J.$$

Entonces, se tiene la **formulación no lineal** del PLDS como el siguiente problema de programación 0-1 no lineal o problema combinatorio no lineal:

$$(2) \quad \begin{cases} \text{Minimizar:} & Z = \sum_{j \in J} c_j x_j + \sum_{i \in I} \min_{x_j=1} c_{ij} \\ \text{sujeto a:} & x_j \in \{0, 1\}, \quad \forall j \in J. \end{cases}$$

El espacio de soluciones consta de $2^m - 1$ elementos que son todas soluciones factibles.

Para conseguir una formulación del PLDS como un problema lineal se introduce un nuevo conjunto de variables de decisión para especificar la forma en que se asigna la demanda producida en cada punto de demanda a los puntos de servicio. Estas variables, llamadas **variables de asignación**, se denotan por x_{ij} , $\forall i \in I, j \in J$ y especifican si la demanda del punto d_i se asigna al punto de servicio ubicado en l_j .

$$x_{ij} := [d_i \text{ es atendido por } l_j], \forall i \in I, \forall j \in J.$$

Las alternativas tienen entonces dos componentes claramente diferenciadas: las localizaciones $x_J = (x_j : j \in J)$ y las asignaciones $x_{IJ} = (x_{ij} : i \in I, j \in J)$, que constituyen la solución $x = (x_J, x_{IJ})$.

El **coste total** asociado a cada solución x se obtiene como la suma de los costes de localización y los de asignación por:

$$(3) \quad c(x) = \sum_{j \in J} c_j x_j + \sum_{j \in J} \sum_{i \in I} c_{ij} x_{ij}$$

Las **soluciones factibles** son aquellas en las que todo punto de demanda esté asignado a algún punto de localización seleccionado como puntos de servicio; es decir, en las que para cada $i \in I$ existe algún $j \in J$ con $x_j = 1$ y $x_{ij} = 1$. Estas soluciones vienen establecidas por las restricciones:

$$(4) \quad \sum_{j \in J} x_{ij} = 1, \quad \forall i \in I.$$

$$(5) \quad x_{ij} \leq x_j, \quad \forall i \in I, \forall j \in J.$$

Además de las restricciones implícitas de las variables binarias:

$$(6) \quad x_{ij}, x_j \in \{0, 1\}, \quad \forall i \in I, \forall j \in J.$$

Se obtiene entonces la llamada **formulación estándar** del PLDS al expresarlo como el problema de programación lineal 0-1 o problema de optimización combinatorio lineal consistente en minimizar (3) sujeto a (4), (5) y (6) dado por:

$$(7) \quad \begin{cases} \text{Minimizar:} & Z = \sum_{j \in J} c_j x_j + \sum_{j \in J} \sum_{i \in I} c_{ij} x_{ij} \\ \text{sujeto a:} & \sum_{j \in J} x_{ij} = 1, \quad \forall i \in I. \\ & x_{ij} \leq x_j, \quad \forall i \in I, \forall j \in J. \\ & x_{ij}, x_j \in \{0, 1\}, \quad \forall i \in I, \forall j \in J. \end{cases}$$

Esta es la formulación más utilizada del PLDS y consta de $m + nm$ variables binarias y $n + mn$ restricciones. El tamaño del conjunto de soluciones es $2^{m(n+1)}$ de las que son factibles $\binom{m}{1}1^n + \binom{m}{2}2^n + \dots + \binom{m}{m}m^n$.

Sumando las restricciones de (5) que tienen un mismo índice j se obtiene el conjunto de restricciones:

$$(8) \quad \sum_{i \in I} x_{ij} \leq n x_j, \quad \forall j \in J.$$

^{*}Las variables binarias o lógicas toman los valores verdadero (=1) o falso (=0) por lo que se les puede asignar el resultado lógico de cualquier aseveración que denotamos encerrada entre corchetes; cuando la aseveración es cierta la variable toma el valor 1 y toma el valor 0 en otro caso.

Teniendo en cuenta (6) se deduce que (8) también obliga a que $x_j = 1$ si para algún $i \in I$ es $x_{ij} = 1$. Se obtiene así la **formulación agregada** del PLDS como el siguiente problema de programación lineal 0-1:

$$(9) \quad \left\{ \begin{array}{l} \text{Minimizar:} \quad Z = \sum_{j \in J} c_j x_j + \sum_{j \in J} \sum_{i \in I} c_{ij} x_{ij} \\ \text{sujeto a:} \quad \sum_{j \in J} x_{ij} = 1, \quad \forall i \in I. \\ \quad \quad \quad \sum_{i \in I} x_{ij} \leq n x_j, \quad \forall j \in J. \\ \quad \quad \quad x_{ij}, x_j \in \{0, 1\}, \quad \forall i \in I, \forall j \in J. \end{array} \right.$$

El número de restricciones del problema combinatorio se reduce drásticamente desde las $n(m+1)$ restricciones de la formulación estándar hasta las $n+m$ restricciones de la formulación agregada.

Ejemplo: Consideremos el PLDS dado por los costes (c_J, c_{IJ}) siguientes:

$$c_J = [2 \quad 2 \quad 2 \quad 3 \quad 3 \quad 3] \quad c_{IJ} = \begin{bmatrix} 0 & 2 & 2 & 2 & 8 & 2 \\ 5 & 0 & 8 & 5 & 2 & 2 \\ 3 & 6 & 0 & 1 & 3 & 6 \\ 5 & 2 & 3 & 3 & 1 & 1 \end{bmatrix}$$

Por tanto el número de localizaciones posibles es $m = 6$ y el de puntos de demanda es $n = 4$. La formulación no lineal tiene sólo 6 variables binarias. Las formulaciones estándar y agregada constan de 30 variables binarias. El programa lineal estándar tiene 28 restricciones y el programa agregado sólo 10. En ambos casos existen $2^{30} \approx 10^9$ soluciones de las que sólo 10752 son factibles. \diamond

1.3. Relajación Lineal.

La **Relajación lineal** (continua) de un problema combinatorio se obtiene al sustituir en las restricciones implícitas el conjunto binario $\{0, 1\}$ por el intervalo $[0, 1]$. Esta relajación se aplica a las distintas formulaciones del PLDS reemplazando las restricciones implícitas (6) por el conjunto de restricciones:

$$(10) \quad 0 \leq x_{ij}, x_j \leq 1, \quad \forall i \in I, \forall j \in J.$$

Las relajaciones del PLDS en sus formulaciones estándar y agregada no coinciden, puesto que, aunque son equivalentes los problemas de minimizar (3) sujeto a (4), (5) y (6) y de minimizar (3) sujeto a (4), (8) y (6); no lo son las correspondientes relajaciones consistentes en minimizar (3) sujeto a (4), (5) y (10) y en minimizar (3) sujeto a (4), (8) y (10), ya que (5) es más fuerte que (8). Por tanto se tienen dos problemas de programación lineal como relajaciones del PLDS que se denominan, respectivamente, **Relajación Lineal Fuerte** (RLF) y **Relajación Lineal Débil** (RLD).

La RLF se expresa por:

$$(11) \quad \left\{ \begin{array}{l} \text{Minimizar:} \quad Z = \sum_{j \in J} c_j x_j + \sum_{j \in J} \sum_{i \in I} c_{ij} x_{ij} \\ \text{sujeto a:} \quad \sum_{j \in J} x_{ij} = 1, \quad \forall i \in I. \\ \quad \quad \quad 0 \leq x_{ij} \leq x_j \leq 1, \quad \forall i \in I, \forall j \in J. \end{array} \right.$$

En ambos problemas relajados, puesto que los costes son no negativos, las acotaciones por 1 de las variables continuas en (10) son innecesarias, por lo que estas restricciones pueden sustituirse por las restricciones implícitas de no negatividad de los problemas de programación lineal:

$$(12) \quad x_{ij}, x_j \geq 0, \quad \forall i \in I, \forall j \in J.$$

Por tanto, la RLF queda reformulada por:

$$(13) \quad \left\{ \begin{array}{l} \text{Minimizar:} \quad Z = \sum_{j \in J} c_j x_j + \sum_{j \in J} \sum_{i \in I} c_{ij} x_{ij} \\ \text{sujeto a:} \quad \sum_{j \in J} x_{ij} = 1, \quad \forall i \in I. \\ \quad \quad \quad x_{ij} \leq x_j, \quad \forall i \in I, \forall j \in J. \\ \quad \quad \quad 0 \leq x_{ij}, x_j, \quad \forall i \in I, \forall j \in J. \end{array} \right.$$

y la RLD queda como un problema con sólo nm variables no negativas y $n + m$ restricciones dado por:

$$(14) \quad \left\{ \begin{array}{l} \text{Minimizar:} \quad Z = \sum_{j \in J} c_j x_j + \sum_{j \in J} \sum_{i \in I} c_{ij} x_{ij} \\ \text{sujeto a:} \quad \sum_{j \in J} x_{ij} = 1, \quad \forall i \in I. \\ \quad \quad \quad \sum_{i \in I} x_{ij} \leq n x_j, \quad \forall j \in J. \\ \quad \quad \quad x_{ij}, x_j \geq 0, \quad \forall i \in I, \forall j \in J. \end{array} \right.$$

Aunque la RLD presenta la ventaja de que tiene una dimensión mucho menor que la RLF, suele ser mucho más útil la RLF porque aporta más frecuentemente soluciones enteras que resuelven el PLDS y en caso contrario aportan una cota mucho más ajustada del valor óptimo del PLDS.

2. Dualidad.

Para aplicar la dualidad a la RLF (11) del PLDS se pone en la forma estándar de la programación lineal (con desigualdades de \geq) separando el segundo conjunto de restricciones. La RLF se formula por:

$$(15) \quad \left\{ \begin{array}{l} \text{Minimizar:} \quad Z = \sum_{j \in J} c_j x_j + \sum_{j \in J} \sum_{i \in I} c_{ij} x_{ij} \\ \text{sujeto a:} \quad \sum_{j \in J} x_{ij} = 1, \quad \forall i \in I. \\ \quad \quad \quad x_j - x_{ij} \geq 0, \quad \forall i \in I, \forall j \in J. \\ \quad \quad \quad -x_j \geq -1, \quad \forall j \in J. \\ \quad \quad \quad x_{ij}, x_j \geq 0, \quad \forall i \in I, \forall j \in J. \end{array} \right.$$

El **problema dual** de la RLF es ^{*}:

$$(16) \quad \left\{ \begin{array}{l} \text{Maximizar:} \quad W = \sum_{i \in I} u_i - \sum_{j \in J} u_j. \\ \text{sujeto a:} \quad -u_{ij} + u_i \leq c_{ij}, \quad \forall i \in I, \forall j \in J. \\ \quad \quad \quad \sum_{i \in I} u_{ij} - u_j \leq c_j, \quad \forall j \in J. \\ \quad \quad \quad 0 \leq u_{ij}, u_j, \quad \forall i \in I, \forall j \in J. \end{array} \right.$$

El dual **reducido** es el problema dual de la formulación (13) de la RLF que viene dado por:

$$(17) \quad \left\{ \begin{array}{l} \text{Maximizar:} \quad W = \sum_{i \in I} u_i \\ \text{sujeto a:} \quad -u_{ij} + u_i \leq c_{ij}, \quad \forall i \in I, \forall j \in J. \\ \quad \quad \quad \sum_{i \in I} u_{ij} \leq c_j, \quad \forall j \in J. \\ \quad \quad \quad 0 \leq u_{ij}, \quad \forall i \in I, \forall j \in J. \end{array} \right.$$

2.1. Duales condensados.

En el problema dual estándar (16) se pueden realizar algunas simplificaciones. Dado u_{IJ} , para maximizar W se minimiza u_J , entonces por el segundo conjunto de restricciones (denotando por $(a)^+ = \max\{a, 0\}$) es: $u_j = (\sum_{i \in I} u_{ij} - c_j)^+$, $\forall j \in J$. Además, dado u_I , para minimizar u_J se minimizan los u_{IJ} , entonces por

^{*}En la notación de los duales continuamos ajustándonos a la notación estándar al mantener exclusivamente la letra u para designar las variables duales. Aparecen, por ahora tres tipos de variables duales también distinguidas por sus índices u_i , u_j y u_{ij} . Aunque ahora dos de estos grupos de variables distintas con un sólo subíndice no hay lugar a confusión si continuamos con la exclusividad de las letras i y j para los subíndices correspondientes.

el primer conjunto de restricciones de (16) es: $u_{ij} = (u_i - c_{ij})^+, \forall i \in I, j \in J$. Por tanto, $\forall j \in J$ es: $u_j = \left(\sum_{i \in I} (u_i - c_{ij})^+ - c_j\right)^+$. Se obtiene entonces el problema **dual condensado** que queda formulado como:

$$(18) \quad \max_{u_I} \left\{ \sum_{i \in I} u_i - \sum_{j \in J} \left(\sum_{i \in I} (u_i - c_{ij})^+ - c_j \right)^+ \right\}.$$

En el problema dual (16) de la RLF se tiene que, si para algún $j \in J$ es $u_j = \left(\sum_{i \in I} (u_i - c_{ij})^+ - c_j\right)^+ > 0$ entonces existe un $i \in I$ con $u_i - c_{ij} > 0$. Por tanto, si disminuye u_i una cantidad Δ entonces al menos u_j también disminuye en Δ , con lo que el objetivo W en (16) no disminuye. Por tanto, existe una solución óptima del dual con $u_j = 0, \forall j \in J$. De aquí, se obtiene también la formulación reducida del dual (17). En este problema dual, si para algún $i \in I$ es $u_i < c_{ij}, \forall j \in J$, entonces u_i puede aumentar manteniendo el segundo conjunto de restricciones de (17) y en beneficio del objetivo W . Por tanto, existe una solución óptima donde, para cada $i \in I$ es $u_i \geq \min_{j \in J} c_{ij}$ y además, $u_{ij} = (u_i - c_{ij})^+, \forall j \in J$. De donde se obtiene la **segunda forma condensada del dual** formulado por:

$$(19) \quad \begin{cases} \text{Maximizar:} & W(u_I) = \sum_{i \in I} u_i \\ \text{sujeito a:} & u_i \geq \min_{j \in J} c_{ij}, \quad \forall i \in I. \\ & \sum_{i \in I} (u_i - c_{ij})^+ - c_j \leq 0, \quad \forall j \in J. \end{cases}$$

Hay que destacar que las dos formas condensadas son problemas no lineales pero con sólo n variables no restringidas en signo.

2.2. Dual Lagrangiano.

El problema **dual lagrangiano** del PLDS se obtiene al aplicar la dualidad lagrangiana (Geoffrion, 1974) a la formulación estándar (7) del PLDS mediante la dualización las restricciones de igualdad (4) multiplicándolas por las correspondientes variables duales e incorporándolas a la función objetivo. Se obtiene el problema combinatorio o problema de programación lineal 0-1:

$$(20) \quad \begin{cases} \text{Minimizar:} & \sum_{j \in J} c_j x_j + \sum_{j \in J} \sum_{i \in I} c_{ij} x_{ij} + \sum_{i \in I} u_i (1 - \sum_{j \in J} x_{ij}) \\ \text{sujeito a:} & x_{ij} \leq x_j, \quad \forall i \in I, \forall j \in J. \\ & x_{ij}, x_j \in \{0, 1\}, \quad \forall i \in I, \forall j \in J. \end{cases}$$

De la segunda forma condensada del dual (19) se tiene, para cada conjunto de valores u_I , una cota del primal (11), la RLF del PLDS, dada por $W(u_I) = \sum_{i \in I} u_i$. El dual lagrangiano (20) proporciona también una cota del PLDS, para cada u_I , que se puede expresar por:

$$(21) \quad L(u_I) = \sum_{i \in I} u_i + \min_{x_{IJ} \leq x_J} \left\{ \sum_{j \in J} c_j x_j + \sum_{j \in J} \sum_{i \in I} (c_{ij} - u_i) x_{ij} \right\}$$

Si las variables duales u_I satisfacen el primer conjunto de restricciones de (19), $u_i \geq \min_{j \in J} c_{ij}$, entonces el problema lagrangiano (20) correspondiente es un problema combinatorio resoluble directamente teniendo en cuenta el signo de los coeficientes de las variables en la función objetivo de (21) expresada como en (21). Si la solución viene dada por:

$$\begin{aligned} \forall i \in I, j \in J : & \quad x_{ij} := [u_i > c_{ij}]; \\ \forall j \in J : & \quad x_j := \left[\sum_{i \in I} (u_i - c_{ij})^+ = c_j \right], \end{aligned}$$

entonces la cota vuelve a ser: $L(u_I) = \sum_{i \in I} u_i$. Esta solución satisface las condiciones de holguras complementarias:

$$\left(\sum_{i \in I} (u_i - c_{ij})^+ - c_j \right) \cdot x_j = 0, \quad \forall j \in J.$$

Por tanto, una buena solución del primal se obtendría de la solución óptima del dual por:

$$x_j := \left[\sum_{i \in I} (u_i - c_{ij})^+ = c_j \right], \quad \forall j \in J.$$

3. Métodos Duales de Solución.

La mayoría de los métodos directos (no basados en la dualidad) se centran en la formulación estándar (7) del PLDS. La cuestión principal en estos procedimientos es que, al analizar la RLF (11) del PLDS, se observa que, aunque el poliedro de las soluciones factibles tiene vértices fraccionarios, el resolución del problema da muy frecuentemente soluciones óptimas enteras (ReVelle y Swaim, 1970). Además, aunque dicha solución sea fraccionaria, da una cota muy ajustada del PLDS, por lo que, la ramificación y acotación terminará en pocos nodos de enumeración. Sin embargo, tiene el inconveniente de que hay un gran número de restricciones (5). Los procedimientos más comunes de solución consisten en ramificar en las x_j y buscar acotaciones por la RLD (14) (ver Khumawala (1972)). Por otro lado, las restricciones (5) generalizan las de acotación de variables, ya que las cotas de las variables vienen dadas por otras variables. Por tanto se puede adaptar la aplicación del método del simplex para variables acotadas (ver Schorage (1975)) o introducirlas como cortes al ser violadas (ver Morris (1978)). Sin embargo, en los procedimientos de mayor éxito se emplea la dualidad.

3.1. El subgradiente.

Al expresar el **dual lagrangiano** (20) del PLDS como la función en las variables duales u_I dada por:

$$(22) \quad L(u_I) = \min_{x_{IJ} \leq x_J} \left\{ \sum_{j \in J} c_j x_j + \sum_{j \in J} \sum_{i \in I} c_{ij} x_{ij} + \sum_{i \in I} u_i \left(1 - \sum_{j \in J} x_{ij} \right) \right\},$$

resulta el mínimo entre un conjunto finito de funciones lineales, por tanto es una función lineal a trozos y convexa. Por ello se ha propuesto aplicar el algoritmo del subgradiente que es eficiente para este tipo de función objetivo (Cornuejols et al. (1990)). Se trata de una generalización del método del gradiente para minimizar funciones convexas no lineales suaves; la dirección del gradiente es sustituida por la del subgradiente en los puntos de no diferenciabilidad (Held et al. (1974)).

Dada la solución dual u_I^k en la iteración k del método del subgradiente, se genera una nueva solución dual por $u_I^{k+1} = u_I^k + \gamma^k \partial L(u_I^k)$, donde γ^k es la amplitud de paso y $\partial L(u_I)$ es el subgradiente de $L(\cdot)$ en el punto u_I . El subgradiente está constituido por el cono generado por las soluciones óptimas del lagrangiano; es decir, la direcciones de componentes dadas por

$$\partial L(u_I)_i = 1 - \sum_{j \in J} x_{ij}^*, \quad \forall i \in I,$$

para cada una de las soluciones óptimas (x_J^*, x_{IJ}^*) del correspondiente problema dual lagrangiano (20).

3.2. Descomposición.

Una vez establecidas las localizaciones por x_J , el problema de determinar la asignación óptima de la demanda de cada punto d_i se puede formular como el problema lineal 0-1 en las variables x_{iJ} dado por:

$$(23) \quad \left\{ \begin{array}{l} \text{Minimizar:} \quad \sum_{j \in J} c_{ij} x_{ij} \\ \text{sujeto a:} \quad \sum_{j \in J} x_{ij} = 1, \\ \quad \quad \quad x_{ij} \leq x_j, \quad \forall j \in J. \\ \quad \quad \quad x_{ij} \in \{0, 1\}, \quad \forall j \in J. \end{array} \right.$$

Nótese que en este problema combinatorio el índice i está fijo y los valores x_J son constantes, las únicas variables de decisión son x_{iJ} .

La relajación lineal de (23) se obtiene al sustituir las restricciones implícitas binarias de los problemas 0-1 por las restricciones implícitas de no negatividad de los problemas de programación lineal consistente en:

$$(24) \quad \left\{ \begin{array}{l} \text{Minimizar:} \quad V_i = \sum_{j \in J} c_{ij} x_{ij} \\ \text{sujeto a:} \quad \sum_{j \in J} x_{ij} = 1, \\ \quad \quad \quad x_{ij} \leq x_j, \quad \forall j \in J. \\ \quad \quad \quad x_{ij} \geq 0, \quad \forall j \in J. \end{array} \right.$$

En la relajación lineal continua del PLDS, y de los problemas derivados como en (24), el significado de un valor fraccionario en una variable es el de una proporción de la demanda respectiva. Si la variable x_j no es entera, a l_j sólo se le puede asignar hasta una proporción x_j de la demanda de cualquier punto d_i .

Denotemos por $V_i(x_J)$ al correspondiente valor óptimo del problema (24) que da el coste de atender, con la asignación óptima, la demanda del punto d_i con las localizaciones x_J , aún cuando las variables x_J tengan valores fraccionarios. Si las variables de localización x_J tienen valores enteros ($x_j \in \{0, 1\}, \forall j \in J$), las asignaciones óptimas x_{iJ} se obtienen asignando toda la demanda de d_i al punto de servicio más económico por

$$x_{ij} := [c_{ij} = \min_{x_j=1} c_{ij}].$$

Sin embargo, la solución óptima de la relajación lineal (24), aún cuando x_J sea fraccionario, se puede seguir calculando de forma directa al asignar toda la proporción de la demanda posible al punto de servicio más económico hasta agotar la demanda. Entonces, $V_i(x_J)$ como función de x_J en $[0, 1]^J$ es convexa y lineal a trozos. Por tanto, la RLF (11) del PLDS se reformula como

$$(25) \quad \min_{0 \leq x_J} Z(x_J) = \min_{0 \leq x_J} \left\{ \sum_{i \in I} V_i(x_J) + \sum_{j \in J} c_j x_j \right\}$$

que es un problema de minimizar una función convexa lineal a trozos con restricciones simples.

Este problema se resuelve por el método del subgradiente. La j -ésima componente del subgradiente es:

$$(26) \quad \partial_j Z(x_J) = \sum_{i \in I} \partial_j V_i(x_J) + c_j$$

Para obtener $\partial_j V_i(x_J)$ hay que hacer un sencillo análisis paramétrico del problema de programación lineal (24) en el término de la derecha, que es el que depende de x_J . Sea c_{ij_0} el coste de la variable x_{ij} en el que se agota la demanda de d_i al construir la asignación óptima que da $V_i(x_J)$. Entonces $\partial_j V_i(x_J) = (c_{ij} - c_{ij_0})^+$.

La reformulación (25) de la RLF del PLDS se puede exponer como el problema en las variables V_I y x_J siguiente:

$$(27) \quad \left\{ \begin{array}{l} \text{Minimizar:} \quad Z = \sum_{i \in I} V_i + \sum_{j \in J} c_j x_j \\ \text{sujeto a:} \quad V_i \geq V_i(x_J), \quad \forall i \in I. \\ \quad \quad \quad 0 \leq x_j, \quad \forall j \in J. \end{array} \right.$$

Cada $V_i(x_J)$ se puede calcular por el dual de (24) que es:

$$(28) \quad \begin{cases} \text{Maximizar:} & W_i = u_i - \sum_{j \in J} x_j u_{ij} \\ \text{sujeto a:} & u_i - u_{ij} \leq c_{ij}, \quad \forall j \in J. \\ & u_{ij} \geq 0, \quad \forall j \in J. \end{cases}$$

Donde la solución óptima verifica $u_{ij} = (u_i - c_{ij})^+$, lo que lo convierte en un problema sólo en la variable dual libre u_i . Entonces:

$$V_i(x_J) = \max_{u_i} \left\{ u_i - \sum_{j \in J} x_j (u_i - c_{ij})^+ \right\},$$

que es el máximo de una función lineal a trozos. Los cambios de pendiente de esta función están en los valores $u_i = c_{ij}$, para $j \in J$. Por tanto:

$$(29) \quad V_i(x_J) = \max_{k \in J} \left\{ c_{ik} - \sum_{j \in J} x_j (c_{ik} - c_{ij})^+ \right\}.$$

La formulación de la RLF del PLDS por (27) se reformula con la igualdad (29) como el problema en las variables V_I y x_J dado por:

$$(30) \quad \begin{cases} \text{Minimizar:} & Z = \sum_{i \in I} V_i + \sum_{j \in J} c_j x_j \\ \text{sujeto a:} & V_i + \sum_{j \in J} x_j (c_{ik} - c_{ij})^+ \geq c_{ik}, \quad \forall i \in I, \forall k \in J. \\ & 0 \leq x_j, \quad \forall j \in J. \end{cases}$$

Se tienen $n + m$ variables y mn restricciones que se pueden interpretar como cortes a introducir sólo cuando sean violadas. Se pueden desarrollar desigualdades aun más fuerte (ver Magnanti y Wong (1981)).

3.3. Reducción Canónica.

Se realiza una reformulación del PLDS por agregación y disgregación de la demanda de la forma siguiente. La **agregación** de los puntos de demanda d_s y d_t consiste en sustituirlos por d_i con costes: $c_{ij} = c_{sj} + c_{tj}$, $\forall j \in J$. La **disgregación** del punto de demanda d_i en d_s y d_t consiste en sustituirlo por d_s y d_t con: $c_{ij} = c_{sj} + c_{tj}$, $\forall j \in J$. La agregación de los puntos de demanda d_s y d_t supone que ambos deben atenderse obligatoriamente desde el mismo punto de servicio, la disgregación de d_i permite que parte de su demanda pueda ser atendida por puntos de servicio distintos. Si la demanda tiene que fraccionarse ($0 < x_j < 1$) la disgregación abre más posibilidades. Entonces

$$(31) \quad V_i(x_J) \geq V_s(x_J) + V_t(x_J), \quad \forall x_J \in [0, 1]^J.$$

Por tanto, la agregación produce una supraestimación del coste y la disgregación produce una subestimación. En general se verifica (31); se dice que la agregación o disgregación es **válida** si:

$$(32) \quad V_i(x_J) = V_s(x_J) + V_t(x_J), \quad \forall x_J \in [0, 1]^J.$$

La asignación óptima de la demanda de cada punto d_i se realiza eligiendo sucesivamente el punto de localización l_j de menor coste al que se le asigna toda la demanda posible de d_i hasta agotarla. Si estas elecciones para los puntos d_s y d_t pueden hacerse coincidir entonces su agregación es válida. Si un punto d_i se disgrega en los puntos d_s y d_t de forma que coincidan estas elecciones, la disgregación es válida. Para cada $i \in I$, sea J^i el conjunto de las reordenaciones (j_1, j_2, \dots, j_n) de $J = \{1, 2, \dots, n\}$ tales que: $c_{ij_1} \leq c_{ij_2} \leq \dots \leq c_{ij_n}$. El conjunto J^i está constituido por las permutaciones de índices resultantes al ordenar de menor a mayor la fila i -ésima de la matriz de costes de asignación; ordenando de forma arbitraria los costes iguales. Estas reordenaciones dan las posibles secuencias de elecciones óptimas de los puntos de servicio para atender el punto de demanda d_i . Entonces se tienen las siguientes agregaciones y disgregaciones válidas: (i) si $J^s \cap J^t \neq \emptyset$ entonces la agregación de d_s y d_t es válida, y (ii) si $J^i \cap J^s \cap J^t \neq \emptyset$ entonces la disgregación de d_i en d_s y d_t es válida.

Cada punto de demanda d_i , $i \in I$, en cuya fila existan al menos dos costes distintos no nulos se disgrega iterativamente en dos puntos de demanda d_s y d_t de la forma siguiente. Sea $r = \min_{c_{ij} > 0} c_{ij}$. Entonces se toman los costes de asignación de d_s dados por: $c_{sj} := r \cdot [c_{ij} > 0]$, y los costes de asignación de d_t dados por: $c_{tj} := (c_{ij} - r) \cdot [c_{ij} > 0]$. Nótese que $J_i \subseteq J^s \cap J^t$ por lo que las disgregaciones son válidas. Esta disgregación se aplica recursivamente hasta descomponer cada punto de demanda d_i en un conjunto D_i de, a lo sumo, $m - 1$ filas. Para cada una de estas filas i existe un conjunto de índices $\kappa \subseteq J$ y un valor $r_\kappa > 0$ tales que:

$$c_{ij} = r_\kappa \cdot [j \notin T] = \begin{cases} 0 & \text{si } j \in T. \\ r_T & \text{si } j \notin T. \end{cases}$$

Ejemplo. Si la fila i , reordenada de acuerdo a J^i es $(c_{i1}, c_{i2}, c_{i3}, c_{i4}) = (1, 4, 4, 6)$ entonces el punto de demanda d_i se disgrega en d_s y d_t con las filas

$$(c_{s1}, c_{s2}, c_{s3}, c_{s4}) = (1, 1, 1, 1) \quad \text{y} \quad (c_{t1}, c_{t2}, c_{t3}, c_{t4}) = (0, 3, 3, 5).$$

Y esta última fila se disgrega en $(0,3,3,3)$ y $(0,0,0,2)$. Por tanto, el punto de demanda d_i con costes $(1, 4, 4, 6)$ se puede sustituir por tres puntos de demanda con costes $(1, 1, 1, 1)$, $(0, 3, 3, 3)$ y $(0, 0, 0, 2)$. \diamond

Una vez disgregados todos los puntos de demanda, se agregan las filas o puntos de demanda d_s y d_t con valores nulos en las mismas columnas; es decir, tales que $c_{sj} = 0 \Leftrightarrow c_{tj} = 0$, $\forall j \in J$. Esta agregación es válida. Además, todo punto de demanda con coste constante puede eliminarse.

La matriz de costes de asignación $C = c_{IJ}$ se reduce a la matriz R llamada **reducción canónica** de C . La matriz R tiene, a lo sumo, $m(n - 1)$ filas y cada fila es un punto de demanda ficticio d_κ que representa a un conjunto κ , $\emptyset \subset \kappa \subset J$, con costes: $r_{\kappa j} = r_\kappa \cdot [j \notin \kappa]$.

Ejemplo: La matriz C de los costes de asignación del ejemplo se transforma en la matriz canónica R por:

$$C = \begin{bmatrix} 0 & 2 & 2 & 2 & 8 & 2 \\ 5 & 0 & 8 & 5 & 2 & 2 \\ 3 & 6 & 0 & 1 & 3 & 6 \\ 5 & 2 & 3 & 3 & 1 & 1 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & 2 & 2 & 2 & 2 & 2 \\ 0 & 0 & 0 & 0 & 6 & 0 \\ 2 & 0 & 2 & 2 & 2 & 2 \\ 3 & 0 & 3 & 3 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 2 & 2 & 0 & 0 & 2 & 2 \\ 0 & 3 & 0 & 0 & 0 & 3 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & 2 & 2 & 2 & 2 & 2 \\ 0 & 0 & 0 & 0 & 6 & 0 \\ 2 & 0 & 2 & 2 & 2 & 2 \\ 4 & 0 & 4 & 4 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 2 & 2 & 0 & 0 & 2 & 2 \\ 0 & 3 & 0 & 0 & 0 & 3 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} = R.$$

\diamond

Cada punto de demanda d_κ se puede asignar con coste nulo a cualquier punto de localización l_j con $j \in \kappa$. Por tanto, dado $x_J \in [0, 1]^J$, el coste de la asignación de d_κ es sólo la parte que no se puede atender con los puntos de servicio l_j con $j \in \kappa$. Luego, si $1 \leq \sum_{j \in \kappa} x_j$ entonces la demanda de d_κ es atendida gratis (con coste 0). En otro caso, la cantidad $1 - \sum_{j \in \kappa} x_j$ es la fracción de la demanda de d_κ no satisfecha. Por tanto,

$V_\kappa(x_J) = r_\kappa(1 - \sum_{j \in \kappa} x_j)^+$. De donde, denotando $x_\kappa = (1 - \sum_{j \in \kappa} x_j)^+$, se puede calcular $V_\kappa(x_J)$, para cada punto de demanda d_κ , por el problema:

$$(33) \quad \begin{cases} \text{Minimizar:} & Z = r_\kappa x_\kappa \\ \text{sujeto a:} & x_\kappa \geq \sum_{j \in \kappa} x_j - 1. \\ & x_\kappa \geq 0. \end{cases}$$

Sea \mathcal{K} la familia de subconjuntos κ de J representados en la matriz de costes canónica R . Entonces la RLF (11) del PLDS se puede formular por:

$$(34) \quad \left\{ \begin{array}{l} \text{Minimizar:} \\ \text{sujeto a:} \end{array} \right. \quad \begin{array}{l} Z = \sum_{\kappa \in \mathcal{K}} r_{\kappa} x_{\kappa} + \sum_{j \in J} c_j x_j \\ x_{\kappa} - \sum_{j \in \kappa} x_j \geq -1, \quad \forall \kappa \in \mathcal{K}. \\ x_{\kappa}, x_j \geq 0, \quad \forall \kappa \in \mathcal{K}, \forall j \in J. \end{array}$$

El dual de este problema de programación lineal es:

$$(35) \quad \left\{ \begin{array}{l} \text{Maximizar:} \\ \text{sujeto a:} \end{array} \right. \quad \begin{array}{l} W = - \sum_{\kappa \in \mathcal{K}} u_{\kappa} \\ \sum_{\kappa \ni j} u_{\kappa} \leq c_j, \quad \forall j \in J. \\ 0 \leq u_{\kappa} \leq r_{\kappa}, \quad \forall \kappa \in \mathcal{K}. \end{array}$$

Este programa dual tiene, a lo sumo, del orden de mn variables y m restricciones (además de las de acotación de variables). Se trata de la formulación más compacta conocida del PLDS. Además es la formulación más rápida para aplicar el simplex con variables acotadas (Cornuejols et al. 1980). Tiene ventajas computacionales respecto a otras reformulaciones (Simao y Thizy, 1986)

4. Heurísticas.

4.1. Heurísticas Primitives.

Para la aplicación de heurísticas primales se considera la formulación del PLDS como el problema combinatorio (1). Existen diversas **estrategias heurísticas de búsqueda** fundamentales para ir modificando una solución S de un problema de localización-asignación en busca de una solución óptima (ver, Moreno et al. (1995) o Moreno (1995)). Estas estrategias de búsqueda están basadas en movimientos elementales sobre el espacio de soluciones que, para el PLDS son principalmente de tres tipos: apertura, cierre e traslado. La **apertura** de un punto de servicio en l_j , $j \notin S$, consiste en hacer $S \leftarrow S + \{j\}$. El **cierre** del punto de servicio ubicado de l_k , $k \in S$, es: $S \leftarrow S - \{k\}$. El **traslado** de un punto de servicio desde l_k hasta l_j , $j \notin S$, $k \in S$, consiste en: hacer $S \leftarrow S + \{j\} - \{k\}$.

Las estrategias heurísticas voraces y ansiosas tratan de aplicar movimientos para mejorar la solución mientras sea posible. Ambas están basadas en un análisis de todos los movimientos aplicables a la solución actual. La estrategia **voraz** (*greedy*) aplica el movimiento que produce la mayor mejora, mientras que la estrategia **ansiosa** aplica el primer movimiento de mejora que encuentra. Si sólo se consideran movimientos de apertura se suele comenzar con la solución vacía ($S = \emptyset$) y si se consideran sólo movimientos de cierre con la solución completa ($S = J$), pero en general se pueden considerar los tres tipos de movimientos y partir de cualquier solución. La heurística voraz de apertura fue uno de las primeras formas que se consideraron para abordar estos problemas (Kuehn y Hamburger (1963)). Sin embargo, en la actualidad se están aplicando con éxito, al igual que a otros muchos problemas de optimización combinatoria difíciles, diversas metaheurísticas como los Redes Neuronales, Algoritmos Genéticos, búsquedas Tabú, Recristalización Simulada, e incluso procedimientos de integración entre ellas (ver Moreno (1995)).

4.2. Heurísticas Duales Ascendentes.

Los procedimientos heurísticos duales ascendentes, como el propuesto por Bilde y Krarup (1977) y, en especial el algoritmo DUALOC de Erlenkotter (1978), y las diferentes mejoras aparecidas, están reconocidos unánimemente como la herramienta de más éxito para este problema. Se considera la RLF (13) y su dual (17) que, tomando $u_{ij} = (u_i - c_{ij})^+$, queda en la segunda forma condensada (19). Denotando las holguras por:

$$(36) \quad s_j(u_I) = c_j - \sum_{i \in I} (u_i - c_{ij})^+ \geq 0, \quad \forall j \in J.$$

el dual condensado se puede reformular como:

$$\begin{cases} \text{Maximizar:} & W(u_I) = \sum_{i \in I} u_i \\ \text{sujeto a:} & s_j(u_I) \geq 0, \quad \forall j \in J. \end{cases}$$

Los procedimientos **duales ascendentes** consisten en incrementar repetidamente el valor de la función objetivo del dual aumentando variables u_i hasta que todas queden bloqueadas por las restricciones (36) al anular las holguras.

En el dual (17), dado $u_{IJ} \geq 0$, para maximizar $W(u_I)$ se toma, del primer conjunto de restricciones,

$$(37) \quad u_i := \min_{j \in J} \{c_{ij} + u_{ij}\}, \quad \forall i \in I.$$

Se dice que la asignación de d_i a l_j (o simplemente de i a j) es **válida** si es en j donde se alcanza el mínimo en esta ecuación; $u_i = c_{ij} + u_{ij}$. Sea, para cada $i \in I$, el conjunto de asignaciones válidas de i dado por: $J_i := \{j \in J : u_i = c_{ij} + u_{ij}\}$.

El procedimiento dual ascendente originalmente parte de $u_{ij} = 0, \forall i \in I, j \in J$. Por tanto, $u_i = \min_{j \in J} c_{ij}$, $\forall i \in I$. Sin embargo, se podría empezar con otros valores tales que con $u_i \geq \min_{j \in J} c_{ij}, \forall i \in I$. En cada iteración del procedimiento dual ascendente se incrementan variables duales u_i dadas por (37) aumentando algunas u_{ij} en una cantidad $\delta > 0$. Para aumentar la variable u_i hay que incrementar, al menos, los u_{ij} de las asignaciones válidas de i ; es decir, las de $j \in J_i$. A menos que $J_i = J$, aparece una nueva asignación válida de i si se toma como valor de δ la diferencia al valor c_{ij} inmediato superior a u_i , es decir, $\delta = \min_{u_i < c_{ij}} c_{ij} - u_i$. Este incremento se puede llevar a cabo siempre que se mantengan las restricciones de las holguras (36); en otro caso, el aumento queda bloqueado por alguna de estas restricciones. Entonces el incremento δ es el máximo permitido por ellas; es decir, $\delta = \min_{j \in J_i} \{c_j - \sum_{i \in I} u_{ij}\}$. Sea $J^+ := \{j \in J : \sum_{i \in I} u_{ij} = c_j\}$ el conjunto de las **localizaciones esenciales** que corresponden a las holguras anuladas. El proceso se detiene si $J^+ \cap J_i \neq \emptyset, \forall i \in I$ ya que en tal caso todas las u_i están bloqueadas.

Entonces, $\forall i \in I$, existe una asignación válida de J^+ , por tanto se tiene una solución del primal (la RLF del PLDS) y del PLDS seleccionando las localizaciones esenciales y realizando asignaciones válidas. Formalmente, las asignaciones válidas óptimas de cada $i \in I$ vienen dadas por: $J_i^+ := \{j \in J_i \cap J^+ : c_{ij} = \min_{j \in J_i} c_{ij}\}, \forall i \in I$.

Entonces la solución del PLDS dada por *:

$$\begin{aligned} x_j &:= [j \in J^+], & \forall j \in J. \\ x_{ij} &:= [j \in J_i^+], & \forall j \in J, \forall i \in I. \end{aligned}$$

verifica (por las definiciones de J^+ y J_i^+) las condiciones de holguras complementarias de la RLF (13) y el dual reducido (17):

$$(38) \quad (c_{ij} + u_{ij} - u_i)x_{ij} = 0, \quad \forall i \in I, \forall j \in J.$$

$$(39) \quad (c_j - \sum_{i \in I} u_{ij})x_j = 0, \quad \forall j \in J.$$

Si además verifica las condiciones de holguras complementarias:

$$(40) \quad (x_j - x_{ij})u_{ij} = 0, \quad \forall i \in I, \forall j \in J.$$

entonces se tiene una solución óptima para el PLDS. En otro caso, se produce una **abertura** entre el dual y el primal; el **ajuste dual** trata de cerrar esta abertura.

Para realizar este ajuste se observa lo siguiente. Si se viola la igualdad (40) para algún $i \in I$, entonces i debe tener al menos dos asignaciones válidas ya que $u_{ij} > 0$ sólo para asignaciones válidas y sólo se hace una asignación de i ; $x_{ij} = 1$ para un solo $j \in J$. Por tanto, se rebaja u_i al valor anterior disminuyendo simultáneamente todos los u_{ij} , para $j \in J_i$, en una misma cantidad δ (con lo que u_i decrecerá en δ). Entonces

* si hay empates se elige uno cualquiera

se produce una holgura en la correspondiente restricción $s_j = c_j - \sum_{i \in I} u_{ij} \geq 0$ de (36) y se vuelve entonces a aplicar el procedimiento dual ascendente (posiblemente con otra u_i) que vuelve a alcanzar o superar el valor del dual anterior W .

Si el dual ascendente y el ajuste no consiguen cerrar la abertura entonces se entra en un **proceso de ramificación** aprovechando las cotas aportadas.

4.3. Mejoras heurísticas de los duales ascendentes.

Por lo general casi siempre es posible una ejecución del dual ascendente que consigue cerrar la abertura. En estos procedimientos queda margen para realizar el proceso de distintas maneras y de aprovechar la información disponible para mejorar la eficiencia empírica del procedimiento dual ascendente y de la ramificación posterior. Por ejemplo, el algoritmo DUALOC original ramifica en el menor coste del punto de servicio de J_i^+ que contribuye a la violación de la condición de holgura complementaria.

Existen diversas propuestas heurísticas para mejorar la eficiencia empírica de los procedimientos ascendentes del dual. En Körkel (1989) se propone un procedimiento **dual multi-ascendente** que acelera el cálculo de la primera cota inferior y permite empezar con un incremento mayor en el dual ascendente. Para ello, no se calculan las modificaciones en las holguras s_j sino que sólo se acumula la disminución total de los s_j . Se utiliza un criterio heurístico para detener el procedimiento con el que no se garantiza $s_j \geq 0$; sino que se mantiene cerca de la factibilidad y se completa con un proceso descendente en busca de la factibilidad. Se alterna en el orden de selección de las u_i que se elevan en cada iteración y se restringen a un subconjunto de I mediante un test que descarta las u_i bloqueadas a priori.

Para mejorar el **ajuste primal-dual**, Roy y Erlenkotter (1982) propusieron recalcular la solución del primal cada vez que se modifica la del dual. Körkel (1989) propone además un **ajuste múltiple primal-dual** que modifica varias variables u_i a la vez cuando se comienza muy lejos de la región factible.

En cuanto a las ramificaciones, se ha propuesto aplicar el criterio de elegir el subproblema de la **mínima cota inferior** (MLB), lo que produce un rápido aumento de la cota inferior. Otras propuestas implican ramificar en la localización que da lugar a una mayor holgura s_j prefiriendo variables esenciales.

5. Recomendaciones prácticas.

Al enfrentarse a un problema concreto de un tamaño moderado se puede intentar resolver la RLF por cualquier programa de programación lineal lo que puede proporcionar, si la solución es entera, la solución óptima del PLDS y en caso contrario una muy buena acotación. Esta resolución de la RLF se puede incluir en un esquema de ramificación y acotación para acortar considerablemente la enumeración de soluciones. Si el número de restricciones de la RLF lo hace inabordable por un programa estándar, se puede resolver la RLD e ir introduciendo las restricciones de la RLF violadas como cortes. Sin embargo, es conveniente obtener el valor óptimo de la RLF a través de alguno de los problemas duales que tienen unas dimensiones menores lo que facilitaría su resolución por programas estándares de programación lineal. Otra posibilidad consiste en realizar la reducción canónica para obtener un problema de dimensiones más moderadas.

Si no se dispone de un programa de programación lineal estándar que aplicar, uno de los procedimientos más fáciles de implementar es el método del subgradiente para la dualidad lagrangiana. En todo caso, sería importante tener la garantía de disponer pronto, al menos, de una buena solución. Para ello se podría aplicar una de las heurísticas primales que son fácilmente implementables. Sin embargo es más eficiente aplicar alguno de los procedimientos duales ascendentes que proporcionan también muy rápidamente una primera buena solución del PLDS.

Entre los procedimientos especiales, el algoritmo DUALOC ha conseguido una justificada fama y está generalmente disponible en FORTRAN. Para problemas de gran tamaño, una modificación heurística (como la de Körkel (1989)) puede resultar conveniente.

6. Referencias.

1. Balinski, M.L.(1965). *Integer programming: methods, uses, computation*. Management Science 12, 253-313.
2. Bilde, O. ; Krarup, J. (1977). *Sharp Lower Bounds and Efficient Algorithms for the Simple Plant Location Problem*. Annals of Discrete Mathematics 1, 79-97.
3. Cornuejols, G. ; Nemhauser, G. L. ; Wolsey, L. A. (1980). *A Canonical Representation of Simple Plant Location Problems and Its Applications*. SIAM J. of Algebraic and Discrete Methods 1, 261-272.
4. Cornuejols, G. ; Nemhauser, G. L. ; Wolsey, L. A. (1990). *The Uncapacited Facility Location Problem*. En *Discrete Location Theory*. (Francis and Mirchandani, eds.) Cap 3, 119-171.
5. Erlenkotter, D. (1978). *A Dual-Based Procedure for Uncapacitated Facility Location*. Operations Research 26, 992-1009.
6. Galvão, R.D. (1992). *The use of Lagrangean Relaxation in the solution of uncapacitated facility location problems*. Location Science 1, 57-79.
7. Geoffrion, A.M. (1974). *Lagrangean Relaxation for Integer Programming*. Mathematical Programming Study 2, 82-114.
8. Guignard, M. (1988). *A Lagrangean dual ascent algorithm for simple plant location problems*. European Journal of Operational Research 35, 193-200.
9. Held, M.; Wolfe, P. and Crowder, H.P. (1974). *Validation of Subgradient Optimization*. Mathematical Programming 6, 62-88.
10. Khumawala, B.M. (1972). *An Efficient Branch and Bound Algorithm for the Warehouse Location Problem*. Management Science 18, 718-731.
11. Körkel, M. (1989). *On the Exact Solution of Large-Scale Simple Plant Location Problems*. European Journal of Operational Research 39, 157-173.
12. Krarup, J. ; Pruzan, P. M. (1983). *The Simple Plant Location Problem: Survey and Synthesis*. European Journal of Operational Research 12, 36-81.
13. Kuehn, A. A. ; Hamburger, M. J. (1963). *A Heuristic Program for Locating Warehouses*. Management Science 9, 643-666.
14. Magnanti, T.L. and Wong, R.T. (1981). *Accelerated Benders decomposition: Algorithmic Enhancement and Model Selection Criteria*. Operations Research 29, 464-484.
15. Mirchandani, P. B.; Francis, R.L. (eds.) (1990). *Discrete Location Theory*. John Wiley.
16. Moreno, José A.; Mladenovic, N. and Moreno-Vega, J.Marcos (1995) *MultiStart heuristic searches for p-facility location-allocation problems*. Aparecerá en Studies in Locational Analysis, vol. 9 (1995).
17. Moreno, José A. (1995) *Heurísticas de Búsqueda para Problemas Discretos de Localización-Asignación*. Capítulo 6, 89-112, de *Lecturas sobre Teoría de Localización*. Justo Puerto y otros. Universidad de Sevilla, 1995.
18. Morris, J.G. (1978). *On the Extent to which certain fixed-charge depot location problems can be solved by LP*. Journal of the Operational Research Society 29, 71-76.
19. Revelle, C. ; Swain, R. (1970). *Central Facility Location*. Geographical Analysis 2 n.1, 30-42.
20. Schorage, L. (1975). *Implicit Representation of Variable Upper Bounds in Linear Programming*. Mathematical Programming Study 4, 118-132.
21. Simao, H.P. y Thizy J.M. (1986) *A Dual Simplex Algorithm for the Canonical Representation of the Uncapacitated Facility Location Problem*. Working Paper, Princeton.
22. Tcha, D.W.; Ro, H.B. y Yoo, C.B. (1988). *A dual-based add heuristic for uncapacitated facility location*. Journal of the Operational Research Society 39, 873-878.
23. Thisse, J.F.; Zoller, H.G. (eds.) (1983). *Location Analysis of Public Facilities*. North-Holland.