

El objetivo del trabajo que ha dado origen a este artículo consistía en elaborar una formalización de los procedimientos de branch-and-bound que, por su generalidad, comprendiera como casos particulares a los detectados en la literatura y que, por ello, constituyera un marco teórico en el que inscribir las diversas implementaciones existentes y otras que pudieran derivarse, inclusive las de tipo heurístico, por relajación de alguna/s de las condiciones.

El artículo consta, fundamentalmente, de dos partes. La primera incluye una formalización pseudo-axiomática junto con una descripción informal. En la segunda se discute los problemas prácticos y las conclusiones de algunos experimentos.

1. INTRODUCCION

En la abundante literatura existente sobre los procedimientos llamados de branch-and-bound (conocidos también bajo otras denominaciones) se encuentra diversas formalizaciones teóricas y multitud de aplicaciones, no siempre ajustadas al planteamiento abstracto.

El objetivo que se han propuesto los autores, en el contexto de un proyecto de investigación más amplio desarrollado en la Universidad Politécnica de Barcelona, ha sido el elaborar una formalización de los mencionados procedimientos de branch-and-bound -- que comprendiera como casos particulares a las ya conocidas, superara algunas insuficiencias o faltas de generalidad detectadas en las mismas y constituyera un marco teórico en el que inscribir las diversas implementaciones existentes y las que pudieran derivarse, inclusive las de tipo heurístico por relajación de alguna o algunas de las condiciones.

Dicha formalización se ha alcanzado a través de sucesivas definiciones de complejidad creciente. En el presente artículo se expone los resultados alcanzados hasta el momento en dicho trabajo; consta de dos partes:

- La primera consiste en una descripción pseudo-axiomática de los procedimientos de branch-and-bound, junto con una descripción informal; algunas de las características más sobresalientes son:
 - . posibilidad de utilizar más de un procedimiento de examen de los nudos
 - . obtención, si se desea, de soluciones k-óptimas
 - . estrategia potencialmente dinámica.
- En la segunda parte se discute los problemas prácticos y se presenta brevemente -- las conclusiones de algunos experimentos, recogiendo aportaciones de los propios autores y de otros miembros del equipo de investigación de la Universidad Politécnica de Barcelona.

2. FORMALIZACION DE LOS PROCEDIMIENTOS DE BRANCH-AND-BOUND

2.1 Objetivos

En una primera fase del trabajo se llevó a cabo una revisión de la literatura existente; de las diversas formalizaciones detectadas se eligió las de AGIN /1/, HERVÉ /2/ y ROY /3/ para efectuar un estudio a fondo de

- A. Corominas de la Universidad Politécnica de Barcelona.
 - R. Companyys de SEAT y la Universidad Politécnica de Barcelona.
 - Article rebut el Setembre.

las mismas. La primera es un ejemplo de formalización simple en tanto que las otras -- dos están mucho más elaboradas y presentan una mayor generalidad.

Se pretendió que la formalización recogiera los aspectos incorporados en las tres mencionadas y además otros no tenidos en cuenta por lo menos en alguna de ellas.

En definitiva, los requerimientos fundamentales eran los siguientes:

- Definición general del problema a resolver mediante el algoritmo.
- Definición general del procedimiento de separación.
- Distinción entre restricciones implícitas (es decir, que se tienen en cuenta por la propia estructura del algoritmo) y explícitas.
- Posibilidad de tener en cuenta la información que se va obteniendo a medida que avanza el algoritmo para modificar las variables que definen el estado de los nodos ya examinados.
- Consideración dinámica del proceso de construcción de la arborescencia o grafo.
- Posibilidad de obtener más de una solución.
- Posibilidad de variar dinámicamente la estrategia, en función del tiempo transcurrido y/o de la información obtenida en el curso del propio desarrollo del algoritmo.

2.2 Formalización

2.2.1 Planteamiento del problema a resolver

Dados:

- Un conjunto F
- Un subconjunto E del mismo ($E \subset F$)
- Un conjunto totalmente ordenado, K
- Una aplicación f , de E en K
- Un conjunto totalmente ordenado K' y una aplicación g de $K \times K'$ en K tal que

$$g(u, \eta) \geq u \quad \forall u \in K, \forall \eta \in K'$$

$$u' \geq u \Rightarrow g(u', \eta) \geq g(u, \eta) \quad \forall u, u' \in K, \forall \eta \in K'$$

$$\eta' \geq \eta \Rightarrow g(u, \eta') \geq g(u, \eta) \quad \forall u \in K, \forall \eta, \eta' \in K'$$

- Un número natural n_s
- Unos elementos $\epsilon, \delta \in K'$ ($\delta > \epsilon$)

Se desea hallar, si existen, un conjunto de elementos de F

$$S_{\epsilon, \delta}^{n_s} = \{X_1, X_2, \dots, X_{n_s}\} \text{ tales que}$$

a) Pertenezcan a E

$$b) \forall Y \in E \quad f(X_1) \leq g \quad [\bar{f}(Y), \delta]$$

$$c) \forall Y \in E - S_{\epsilon, \delta}^{n_s} \quad f(X_1) \leq g \quad [\bar{f}(Y), \epsilon]$$

Explicación informal. Se trata de hallar -- hasta un máximo de n_s soluciones posibles δ -óptimas, tales que ninguna de ellas supere en ϵ a ninguna de las soluciones posibles no halladas, es decir sean ϵ -óptimas respecto al conjunto unión de las soluciones posibles no halladas con cada una de ellas¹.

2.2.2 Elementos previos y definiciones

Nodos

Sea $P(F)$ el conjunto de las partes de F y N el de los números naturales. El algoritmo operará en un conjunto $\{V_i\} \in P(F) \times N$ -- cuyos elementos se llamarán vértices o nodos.

Definición 1: Llamaremos nudo o vértice a los elementos de un conjunto $\{V_i\}$

$$V_i = (S_i, i), \quad S_i \in P(F), \quad i \in N$$

Definición 2: Diremos que el vértice V_i es vacio si

$$S_i \cap E = \emptyset$$

Definición 3: Llamaremos principio de separación primitivo M a una correspondencia multívoca de $P(F)$ en $P(F)$ que cumple las -- condiciones siguientes:

$$\text{Si } M_0 S \neq \emptyset \text{ para } S \subset F$$

$$a) Y \in M_0 S \Rightarrow Y \neq S$$

b) $UY = S ; Y \in M_0S$

c) $Y, Z \in M_0S \Rightarrow Y \cap Z \neq Y$

Definición 4: Diremos que un vértice V_i es terminal si S_i contiene un solo elemento de E .

Procedimiento de acotación

Existe un conjunto finito de procedimientos $P = \{p_e\}$ tales que cada uno de ellos permite asociar a cada nudo un elemento de $K: C_e(V_i)$ (donde e indica el procedimiento). Se cumplirán las siguientes propiedades siendo $V_i = (S_i, i)$

a) $C_e(V_i) \leq f(X) \quad \forall X \in S_i \quad \forall p_e \in P$

b) Existe por lo menos un procedimiento p_M tal que en los nudos terminales se cumple

$C_M(V_i) = f(X')$ siendo X' la única solución de F contenida en el nudo.

En lo sucesivo se escribirá $C(V_i)$ (sin subíndice) la mejor (es decir la mayor) de las cotas conocidas.

En un nudo vacío se puede hacer $C(V_i) = +\infty$

En un nudo en el que no se posee ninguna información $C(V_i) = -\infty$.

c) Existe un orden entre los procedimientos definido por

$p_e \leq p_{e'} \Rightarrow C_e(V_i) \leq C_{e'}(V_i) \quad \forall V_i$

Este orden es tal que cualquier par de procedimientos posee extremo superior (P posee la estructura de un semi-retículo superior). Por lo dicho puede considerarse

$\sup P = p_M$

Procedimiento de separación

Del principio de separación primitivo extraeremos un principio de separación M tal que

Definición 5: M es una correspondencia multívoca de $\{V_i\}$ en $\{V_i\}$. Diremos que V_j es descendiente o sucesor inmediato de V_i si $V_j \in M(V_i)$.

Diremos que V_i es ascendente o antecesor inmediato de V_j .

A los elementos de $M^k(V_i)$ los llamaremos descendientes de V_i y a los de $M^{-k}(V_i)$ ascendentes, siendo k un número natural no nulo.

Definición 6: Llamaremos separar un nudo o vértice al proceso de hallar uno o varios de sus descendientes inmediatos, identificarlos, y en su caso inicializarlos.

Diremos que los descendientes así hallados han sido generados en el proceso de separación.

Las propiedades de M son:

a) Si V_i es vacío o terminal, $M(V_i) = \emptyset$, es decir V_i no tiene descendientes.

b) Finitud. El conjunto de los descendientes de $V_1 = (E, 1)$ es finito

$|V_1 \cup M(V_1) \cup M^2(V_1) \cup \dots| = |M^*(V_1)| = \text{finito}$

c) Conservación. Siendo $V_i = (S_i, i)$ un nudo no terminal y $\{V_j\} = \{(S_j, j)\} = M(V_i)$ el conjunto de sus descendientes inmediatos se debe cumplir

c1) $S_i = \bigcup_{V_j \in M(V_i)} S_j$

c2) $S_j \cap S_{j'} = \emptyset$

Propiedades de M con relación a P

d) Si V_j es descendiente de V_i

$C(V_j) \geq C(V_i)$

e) Si V_i es el ascendente inmediato de V_j

$C(V_i) = \min \{C(V_j)\}$

$V_j \in M(V_i)$

Definición 7: Llamaremos examinar o acotar un nudo a la operación de aplicarle uno de los procedimientos mencionados y obtener la correspondiente cota².

Estados de un nudo

Definición 8: Diremos que una solución posible es conocida cuando hemos determinado el valor de la función objetivo para la misma, $f(X)$.

Definición 9: Una solución posible ha sido hallada cuando hemos determinado que forma parte de $S_{\epsilon, \delta}^{ns}$.

Llamaremos Q a la cola de las n mejores soluciones conocidas hasta el momento, ordenadas según los valores crecientes de f . Un subconjunto de Q , las primeras soluciones, forman las soluciones halladas de T .

En un momento dado:

$$|T| = n_T \quad |Q| = n_Q$$

Sea $Q = (Y_1, Y_2, \dots)$ llamaremos

$$VBSF = f(Y_1) \quad \text{si } n_Q \geq 1$$

$$VBSF = +\infty \quad \text{si } n_Q = 0$$

$$VFOQ = f(Y_{n_T+1}) \quad \text{si } n_Q - n_T \geq 1$$

$$VFOQ = +\infty \quad \text{si } n_Q - n_T = 0$$

$$VLOQ = f(Y_{n_Q}) \quad \text{si } n_Q = n_s$$

$$VLOQ = +\infty \quad \text{si } n_Q < n_s$$

Definición 10: Diremos que un nudo V_i está definitivamente descartado si se cumple una o varias de las condiciones siguientes:

- a) es vacío
- b) es terminal y se conoce el valor $f(X)$ de la única solución posible X que contiene
- c) $C(V_i) > g[\bar{VBSF}, \bar{\delta}]$
- d) $g[\bar{C}(V_i), \bar{\epsilon}] \geq VLOQ$

(puede substituirse d) por d'): $C(V_i) \geq VLOQ$.

Diremos que V está provisionalmente descartado si no está definitivamente descartado y

$$g[\bar{C}(V_i), \bar{\epsilon}] \geq VFOQ$$

Un vértice que no está ni definitiva, ni --provisionalmente descartado está no descartado.

Definición 11: Un nudo está virtualmente examinado por un procedimiento cuando se conoce una cota del mismo que no puede mejorarse mediante aplicación del procedimiento.

Cuando un nudo se halla virtualmente examinado por p_M se dirá que está totalmente examinado.

Definición 12: Un nudo está definitivamente cerrado si ha sido totalmente examinado y todos sus descendientes inmediatos están definitivamente descartados o totalmente examinados.

Diremos que un nudo está provisionalmente cerrado cuando ha sido totalmente examinado y sus descendientes inmediatos están totalmente examinados, definitivamente descartados o provisionalmente descartados, y existe por lo menos uno en este último estado.

Propiedades:

- a) Los nudos terminales no tienen descendientes, por lo que estarán definitivamente cerrados cuando estén totalmente examinados, y no pueden estar provisionalmente cerrados.
- b) Si un nudo está definitivamente descartado (o provisionalmente descartado) lo están todos sus descendientes.
- c) Si todos los descendientes inmediatos de un nudo están definitivamente descartados (o provisionalmente descartados) también lo está dicho nudo.
- d) Un nudo terminal examinado con el procedimiento p_M está provisionalmente descartado (o definitivamente descartado).
- e) Un nudo terminal cerrado está provisionalmente o definitivamente descartado.

Definición 13: Diremos que un nudo está totalmente separado cuando se ha generado todos sus descendientes inmediatos³.

2.2.3 Descripción del algoritmo

Definición 14: Llamaremos explorar un nudo al proceso de intentar aumentar nuestro conocimiento sobre las soluciones posibles -- que contiene por examen del mismo o de sus descendientes inmediatos.

a) INICIO DEL ALGORITMO: Situar el contador a 2. Colocar en Q las soluciones conocidas, e inicializar n_Q , VBSF, VFOQ, VLOQ ($n_T = 0$). Generar y examinar el nudo -- (E,1).

b) CUERPO DEL ALGORITMO:

1) Buscar un nudo generado y no cerrado a -- explorar. Si existe ir a 3), si no, estudiar la solución de orden n_T+1 de la cola; si existe ir a 2) si no (VFOQ = $+\infty$)
FIN DEL ALGORITMO.

2) Pasar la solución n_T+1 de la cola a T, -- aumentar n_T en 1 y cambiar VFOQ. Si $n_T = n_s$
FIN DEL ALGORITMO, si no, modificar los estados de los nudos y volver a 1).

3) Fijado el nudo a explorar:

3.i) si ha sido detectado como terminal, se examina. Ir a 5)

3.ii) si se ha generado ya todos sus -- descendientes, o bien se decide -- no generar más, ir a 4).

3.iii) Si se decide generar un nuevo descendiente (lo que sería necesario si en el conjunto formado por el nudo a explorar y sus descendientes inmediatos no hay ningún elemento no totalmente examinado) este descendiente puede ser el primero generado o no. Si es el primero, se concretará en primer lugar el paso de M_0 a M, por lo que después, solo faltará formalizar los otros descendientes no formalizados (o sea no generados). -- Cualquier descendiente generado --

será el nudo a examinar. En la generación de $V_j = (S_j, j)$; j recibe el valor del contador, al cual se añade una unidad a continuación. -- Ir a 5).

3.iv) Al intentar generar el primer descendiente se detecta (si no se ha hecho antes) si el nudo es vacío o terminal. Si es vacío, se cierra y descarta definitivamente, se deduce las consiguientes modificaciones de estado y cota e ir a 1). Si es terminal, elegirlo como nudo a examinar, e ir a 5).

4) En el conjunto formado por el nudo a explorar y sus descendientes inmediatos elegir un nudo no totalmente examinado.

5) Elegir el procedimiento de examen del nudo a examinar.

6) Examinar el nudo. Si el nudo es terminal y el examen proporciona la única solución posible X contenida en él, cerrarlo definitivamente, y si $f(X) < VLOQ$, colocar X en Q modificando n_Q (y eventualmente n_T).

En cualquier caso deducir las posibles variaciones de estado del nudo.

7) Deducir las posibles modificaciones en el estado y en las cotas de los demás nudos. Ir a 1).

c) FIN DEL ALGORITMO. Si $n_T = 0$ el problema no tiene solución. Si $n_T = n_s$ hemos hallado una solución completa $S_{\epsilon, \delta}^{n_s} = Q$.

Si $0 < n_T < n_s$ hemos hallado una solución parcial $S_{\epsilon, \delta}^{n_T} = T$.

2.2.4 Justificación del algoritmo

- El algoritmo es finito, puesto que son finitos la sucesión de nudos que pueden generarse en la separación y el conjunto de -- procedimientos de examen.

- Cuando se halla una solución, ésta es ϵ -óptima respecto a todas las soluciones no -- extraídas.

En efecto, al hallar una solución, todos -- los nudos generados están provisional o definitivamente descartados. En efecto, supon^{gamos} que hubiera un nudo no descartado; en este caso, en virtud de la propiedad b) del punto "Estados de un nudo", por lo menos -- uno de sus descendientes inmediatos no estaría descartado y podríamos aplicarle el mismo razonamiento; por consiguiente existiría un camino formado de nudos no descartados -- que llegaría a un nudo terminal, lo cual es absurdo, ya que un nudo terminal cerrado está provisional o definitivamente descartado (propiedad e) del mismo punto citado anteriormente). Luego, la solución hallada que es la mejor de las no halladas hasta el momento, es ϵ -óptima, ya que al estar todos -- los nudos provisional o definitivamente descartados se cumple, salvo para los hallados

$$f(x_0) \leq g_1 [C(v_i), \epsilon] \quad \forall v_i (= (s_i, i)) \in H \cap \bar{T}$$

siendo H el conjunto de los nudos generados

$$\text{Por ser } C(v_i) \leq f(x) \quad \forall x \in S_i$$

se cumple pues

$$f(x_0) \leq g [\bar{f}(x), \epsilon] \quad \forall x \in S_i \quad \forall v_i \in H \cap \bar{T}$$

es decir

$$f(x_0) \leq g [\bar{f}(x), \epsilon] \quad \forall x \in US_i \quad (v_i \in H \cap \bar{T})$$

$$\text{pero } US_i = E \cap \bar{T}$$

$$\text{luego } f(x_0) \leq g [\bar{f}(x), \epsilon] \quad \forall x \in E \cap \bar{T}$$

- Cuando un algoritmo termina es por uno de los dos motivos siguientes:

- a) Se ha hallado n_s soluciones (todas ellas, sucesivamente, ϵ -óptimas respecto al conjunto de soluciones no extraídas).
- b) No existen soluciones posibles no halladas y todos los nudos están definitivamente descartados (no pueden estar provisionalmente descartados al no existir soluciones posibles no halladas).

Es decir, todos los nudos están:

- hallados, o
- no contienen soluciones posibles (son va-

rios), o

- se cumple que

$$\exists x_0 \in E \mid C(v_i) > g [\bar{f}(x_0), \delta]$$

Es decir, cuando el algoritmo termina, o se han encontrado, sucesivamente, todas las soluciones ϵ -óptimas deseadas, o se ha demostrado que sólo las encontradas hasta el momento cumplen las condiciones especificadas⁴.

3. CONSIDERACIONES PRACTICAS

3.1 Estudio de posibilidades

3.1.1 Procedimiento de separación

Empleando la misma notación que en el punto 2.2, el procedimiento de separación queda determinado al definir la correspondencia M, función del problema concreto que se quiere resolver.

La correspondencia M, en la práctica, determina casi siempre una partición del nudo.

Si un conjunto está definido por el valor adoptado por ciertas variables, permaneciendo libre el valor de las demás, dicho conjunto se puede partir en los subconjuntos -- que vienen definidos por cada uno de los valores que puede adoptar una de las variables libres; por supuesto, el procedimiento ha de comprender un método para la elección de la variable.

Es corriente la utilización de variables binarias exclusivamente (una variable discreta siempre se puede representar como combinación lineal de variables binarias); en este caso si M establece una partición es una partición binaria.

Otra forma de establecer una partición binaria, aunque las variables no lo sean, consiste en separar el conjunto en dos, en uno de los cuales una de las variables libres -- toma un determinado valor, en tanto que en el otro se le prohíbe que tome este valor. Este procedimiento coincide con el mencionado en el párrafo anterior si la variable es

binaria.

A primera vista, la elección de la variable que determina la partición parece tener una importancia fundamental.

Aunque en algunos casos la influencia sobre la eficacia del algoritmo no resulta tan -- grande como cabría esperar, en muchos otros es realmente importante.

AGIN /1/ enuncia dos posibles criterios para realizar la elección:

- 1 - Procurar que uno de los nudos resultantes tenga una buena cota (baja en el caso de mínimo). De esta forma se tienen ciertas esperanzas razonables de que el nudo contenga la solución óptima o, por lo menos, soluciones suficientemente -- buenas.
- 2 - Procurar que uno de los nudos resultantes tenga una mala evaluación (alta en el caso de mínimo). Así será muy difícil que el nudo contenga la solución óptima, por lo que es probable que no sea necesario seguirlo explorando.

ROY /3/ es partidario de este último criterio; en concreto, recomienda que el procedimiento de separación favorezca la aparición rápida de nudos de cota elevada (en el caso de mínimo). Recomienda asimismo elegir un -- procedimiento de separación que mantenga agrupadas las mejores soluciones (si se dise-- minan es probable que tengamos que explorar muchos nudos antes de aislar la solución óptima). De todas formas cabe pensar que esto es más un buen deseo que una regla que pueda aplicarse de forma general.

Desde un punto de vista práctico, es conveniente la generación de pocos vértices, por lo que, en el caso de que la separación se efectúa mediante una secuencia de procesos elementales, puede ser útil empezar por -- aquellos que den origen a un menor número -- de vértices siguientes.

3.1.2 Procedimiento de acotación o de examen

Puede consistir en uno o varios procedimien

tos para calcular de forma más o menos sencilla la cota del nudo y uno de varios procedimientos para detectar si el nudo es vacío, es decir, si no contiene soluciones posibles (EE).

Como señala ROY /3/ es deseable que el procedimiento de evaluación sea tal que la desviación entre la cota de cada nudo y el valor óptimo (entre las soluciones posibles -- que contiene) sea del mismo orden en todos los nudos, a fin de que el orden inducido -- por la evaluación en el conjunto de los nudos se aproxime lo más posible al orden que corresponde a los valores óptimos.

AGIN observa que en el caso de disponer de dos métodos de evaluación alternativos conviene aplicar el más preciso (y, por consiguiente, más costoso) en los nudos superiores de la arborescencia, lo cual presumiblemente permitirá descartar, en las exploraciones sucesivas, ramas importantes.

En la práctica no se trabaja casi nunca con muchos procedimientos de cálculo de la cota. Sea cual sea su número, se asociarán o no, en los nudos no terminales, a un "test" de vacío, en función de lo costoso que éste resulte, de acuerdo con criterios predefinidos (vg.: hacer el "test" sólo a determinados niveles, etc.).

En los nudos terminales puede ser interesante emplear un método más complejo que el utilizado en los demás nudos. En efecto, con mucha frecuencia el cálculo del valor exacto del valor óptimo en el nudo es muy costoso, por lo que deberemos tender a evitarlo en la medida de lo posible, lo cual se puede conseguir empleando, en dichos nudos terminales, una serie de procedimientos de evaluación, cada uno más potente que el anterior, en cascada; de esta forma, sólo una -- parte de los nudos terminales deberá ser -- examinada con el procedimiento más potente, es decir, el que da el valor exacto del óptimo. Por supuesto, según las condiciones -- que ha de reunir el procedimiento de examen, en los nudos terminales es obligado efectuar el test de vacío.

Es corriente que el procedimiento de acotación consista en calcular el valor óptimo -- de la función económica en un conjunto E^* --

tal que $E \subset E^*$ e incluso tal que $F \subset E^*$. E^* se obtiene a partir de E o de F suprimiendo alguna restricción. Vg.: En programas lineales en números enteros es frecuente tomar como cota el valor óptimo de la función económica que se obtiene suponiendo las variables continuas, o considerando menos restricciones que las efectivamente existentes.

3.1.3 Inicialización

Es muy conveniente, en la práctica, conocer una solución posible, aunque no esté muy próxima al óptimo, ya que esto puede contribuir a reducir fuertemente la memoria necesaria para almacenar los nudos generados y no cerrados.

En muchos casos se dispone de una solución posible debido a que el problema posee un método de resolución tradicional, y en la inicialización se evaluará su función económica. En otros existirá un método heurístico que generará dicha solución, que puede consistir en una estrategia de exploración rápida para llegar a un vértice terminal.

En cualquier caso, el algoritmo ha de comenzar examinando la raíz de la arborescencia.

3.1.4 Selección del nudo a explorar

Desde luego, en este punto las posibilidades son muy variadas.

La selección del nudo se efectúa de acuerdo con reglas tales como las siguientes:

- Elegir el nudo no cerrado más recientemente examinado.
- Elegir entre todos los nudos no cerrados el de mejor cota (menor en el caso de mínimo, mayor en el caso de máximo).

Este procedimiento es el que aparece con más frecuencia en la literatura (la mayoría de las veces se presenta sólo esta posibilidad, como si fuera la única) e intuitivamente parece ser el mejor, pero esto es indemostrable teóricamente y, de hecho, HERVÉ, por ejemplo, ha comprobado que no lo es en muchos casos. Además, habría que definir --

qué significa mejor, puesto que se puede adoptar distintos puntos de vista: rapidez en alcanzar el óptimo, rapidez en terminar el algoritmo, rapidez en alcanzar una solución posible, ocupación de memoria, etc.

- Elegir uno de los nudos aparecidos en la última separación (el de mejor cota) si entre ellos hay alguno cuya cota no sea superior a la menor conocida en más de una cantidad dada, η ; en caso contrario, elegir el nudo de cota mínima.

En general, los cálculos efectuados al explorar un nudo son útiles para el examen de sus descendientes; normalmente no es posible conservar el resultado de los cálculos más que para el último nudo examinado. Como observa ROY, la introducción del parámetro η fuerza, en la medida que nosotros deseamos, a proseguir la exploración a lo largo de una rama, permitiendo así aprovechar dichos resultados.

Puede decirse que haciendo $\eta=0$ efectuaremos una exploración lo más "inteligente" posible, a costa de desaprovechar el resultado de los cálculos anteriores y, desde luego, de ocupar más memoria; si no se quiere recurrir a memorias externas y no es posible mantener toda la información necesaria en memoria central, quizá no lleguemos a alcanzar realmente el óptimo.

Un enfoque más general, que comprende como casos particulares a las reglas anteriores, es:

- 1) Definir una función que dependa de la cota del nudo, de su nivel, de su accesibilidad (de su ubicación en memoria interna o externa, por ejemplo) de la disponibilidad de los resultados de los cálculos (normalmente, de si el nudo es el último que ha sido explorado o no) y de su grado de separación.
- 2) Elegir el nudo que optimice el valor de esta función.

Este enfoque coincide con el BENAYOUN, ROY y TERGNY /4/ en un trabajo por otra parte especialmente interesante ya que constituye una aplicación práctica de los principios teóricos expuestos en ROY /3/. De hecho el

que una estrategia de selección sea mejor - que otra dependerá fuertemente de la estructura del problema.

3.1.5 Elección del nudo a examinar

Normalmente, en la práctica, el procedimiento de examen tiene un solo nivel por lo que el nudo seleccionado para la exploración no puede ser objeto de examen posterior. Por ello casi siempre el nudo a examinar es un descendiente inmediato (que se ha de generar) del nudo a explorar.

Es corriente que las reglas de selección -- del nudo a explorar y del nudo a examinar -- se combinen de tal forma que el nudo a explorar se selecciona repetidamente hasta -- que queda totalmente separado. Esta forma -- de proceder se puede presentar también formalizando el algoritmo de modo algo distinto (básicamente diciendo que se selecciona para el examen un subconjunto de los descendientes inmediatos del nudo a explorar y -- que, en este caso, el conjunto está formado por todos los dichos descendientes inmediatos) y tiene la ventaja de que el conjunto de candidatos a ser explorados es un subconjunto de los nudos no separados de la parte de la arborescencia que ha sido examinada.

3.1.6 Elección del procedimiento de examen

En la práctica, cuando se examina un nudo -- se profundiza tanto como se puede en su conocimiento (aunque el examen puede realizarse en etapas sucesivas de tal forma que el resultado de una de ellas puede condicionar la siguiente).

Por ello, casi nunca presenta dificultades especiales este punto.

Puede ser útil disponer de distintos procedimientos de examen que se apliquen en función del nivel del nudo a examinar.

3.1.7 Re-evaluación y recalificación

El examen de un nudo puede mejorar la cota de otros nudos, si existe alguna relación -- conocida entre los mismos. Por ejemplo, pue

de mejorar la información sobre todos sus -- ascendientes y descendientes o, incluso, si se ha hallado una solución posible mejor, -- que la ya disponible, sobre otros.

En particular, cuando se conoce cotas de todos los descendientes inmediatos de un nudo, la máxima de todas ellas, si es menor que -- la del nudo en cuestión, puede sustituirla. Por otra parte, el examinar con un nuevo -- procedimiento un nudo puede conducirnos a -- modificar las cotas de sus descendientes inmediatos que no tienen por qué ser mayores que la de su antecesor (todo ello en el caso de maximización).

Además, si en el examen de un nudo se aísla una solución posible mejor que las conocidas hasta el momento, ello puede modificar el estado de los nudos, algunos de los cuales pueden haberse convertido en provisional o definitivamente descartados.

Después del examen, uno de los nudos puede pasar a provisional o definitivamente cerrado.

Por ello después de examinar nuevos nudos -- puede procederse a una revisión de la información almacenada y modificarla si procede.

3.1.8 Almacenamiento de la información sobre los nudos

Algunas estrategias permiten calcular "a -- priori" la ocupación de memoria central. En estos casos el almacenamiento de la información no presenta problemas especiales. Por ejemplo, para una estrategia tipo LIFO -- ("last in-first out") LAWLER y WOOD /5/ sugieren que la información se almacene en -- una pila; cada vez que se practica una separación en un nudo, los nuevos nudos se colocan en la parte superior de la pila; se explota siempre el nudo situado en la parte superior de la pila.

Otras estrategias, en cambio, como las basadas en la selección del nudo de mejor cota, exigen cantidades de memoria imposibles de determinar "a priori" pero que pueden llegar a ser muy grandes. En cualquier caso, a no ser que trabajemos exclusivamente con -- problemas de talla muy reducida, se ha de --

contar, en principio, con que la memoria -- central es insuficiente, por lo cual se ha de recurrir a memorias externas preferible y casi indispensablemente de acceso directo, organizando el archivo de forma adecuada, a provechando asimismo al máximo la memoria central. Sobre este punto, la información -- que puede encontrarse en la literatura es -- prácticamente nula, a pesar de que (o quizá a causa de que) su interés práctico es enorme.

Se ha de tener en cuenta que la solución de trabajar exclusivamente en memoria central y eliminar los nudos "peores" cuando se termina el espacio disponible es muy corriente, incluso en programas comercializados.

3.1.9 Dinamización de la estrategia

En cada uno de los puntos que se ha ido comentando, los procedimientos se pueden ir -- variando a medida que se va desarrollando -- el algoritmo, como consecuencia de:

- el tiempo transcurrido
- la información que se ha ido obteniendo -- en el curso del propio algoritmo
- una intervención humana
- un intercambio de información hombre-máquina.

Por ejemplo, en muchos casos, puede ser interesante la siguiente estrategia:

- Empezar con una estrategia adecuada para encontrar rápidamente soluciones posibles.
- Una vez encontrada una solución posible, seguir con una estrategia adecuada para -- buscar el óptimo.

Otra posibilidad interesante es la de cambiar los parámetros de la función de selección del nudo a explorar, a medida que avanza el tiempo, en el sentido de favorecer el examen de nudos terminales. Con ello aumentan las posibilidades de obtener una solución posible antes de que transcurra el -- tiempo máximo aceptable.

3.2 Experimentación

Los autores han desarrollado diversos expe-

rimentos dentro de un estudio general sobre la programación de inversiones interrelacionadas. Se ha utilizado para las mismas un -- modelo simplificado aunque en el modelo complejo, que incluye unos procedimientos de -- simulación para la evaluación de la función económica, los resultados han sido coherentes. El modelo simplificado era de la forma:

$$[\text{MAX}] z = \sum_{i,j} b_{ij} X_{ij} - \sum_{i,j,k,l} c_{ijkl} \cdot X_{ij} \cdot X_{kl}$$

$$\sum_j X_{ij} = 1 \quad \forall i$$

$$\sum_i d_i \cdot X_{ij} \leq P_j \quad \forall j$$

$$X_{ij} = 0 \text{ ó } 1 \quad \forall i, j$$

Se considera dos tipos de separación: generación de todos los siguientes, o sólo una cada vez y dos tipos de evaluación, uno más potente que el otro, junto con un test de -- vértice vacío, basado en las inecuaciones.

- E1. estrategia clásica, separación total y selección de la mejor cota.
- E2. igual que E1, pero con el procedimiento de evaluación más potente.
- E3. separación total y función de selección que favorece los vértices de mayor nivel.
- E4. estrategia "go-to-right"; siempre se -- parte del vértice últimamente examinado, hasta llegar a un terminal.
- E5. igual que E3 pero con mayor peso del nivel.
- E6. igual que E5 pero con un sólo descendiente.
- E7. igual que E1 pero con ordenación previa de las variables favoreciendo la aparición de vértices vacíos.

No habiendo sido posible realizar por limitaciones de tiempo máquina un plan completo de experiencias las conclusiones son meramente indicativas:

- a) E2 retrasa la aparición de la primera solución posible, y E4 y E5 la adelantan.
- b) La diferencia entre la primera solución posible hallada, que, en general, se alcanza rápidamente y la óptima es siempre inferior al 10% y habitualmente se halla

comprendida entre el 1 y el 2%.

La diferencia mayor se obtiene con E4 y E6.

- c) El número total de vértices examinados es menor con E2 y E7.
- d) El número de vértices terminales examinados es menor en E2 y E7.
- e) Con E4 la ocupación de memoria es el mínimo, con E6 prácticamente igual y con E2 - el máximo. Considerando la ocupación teórica máxima igual al número de vértices - del nivel inmediatamente inferior al máximo, se obtuvo como máximo una ocupación - del orden del 15% de este máximo teórico.
- f) La eficacia aumentó con el tamaño del problema.

En el caso del modelo complejo, se ha resuelto ocho problemas distintos, empleando en todos ellos estrategias parecidas, de tipo clásico. Tal como se ha dicho, los resultados - son consistentes con los obtenidos en el caso del modelo simplificado; en general, en el modelo complejo:

- la primera solución posible aparece en un tiempo no superior al 10% del empleado para terminar el algoritmo.
- la diferencia entre el valor de la función económica en esta primera solución posible y en el óptimo es asimismo inferior al 10%
- el tiempo requerido para alcanzar el óptimo es generalmente del orden del 15-25% -- del total.

Estos resultados coinciden sustancialmente - con los obtenidos por GINSBURG y VAN PETERSEN /6/ en sus experimentos con un problema concreto:

- la primera solución posible encontrada no difiere en ningún caso (de los probados - por dichos autores) en más de un 5% de la solución óptima; los autores observan, además, que esto es común a la mayoría de algoritmos de este tipo.
- el número total de iteraciones es por lo

menos igual al doble del número de nudos explorados para obtener la primera solución posible.

3.3 Transformación en un procedimiento heurístico

3.3.1 Justificación

De lo expuesto anteriormente se deduce que la eficacia fundamental de los métodos de - exploración dirigida estriba en la facilidad de obtención de soluciones posibles -- "buenas". El óptimo aparece generalmente -- pronto, pero lo difícil y costoso es saber que el óptimo es "óptimo". Parece más interesante, pues, plantear estrategias que vayan a buscar soluciones posibles que estrategias que se orienten únicamente a buscar el óptimo. Si, por otra parte, a las naturales dificultades del método se añade limitaciones de tiempo de tratamiento, o espacio de memoria, se deberá incumplir alguno de - los condicionamientos que aseguran que el - procedimiento encuentra el óptimo, por lo - que habremos transformado el mismo en un -- procedimiento heurístico.

3.3.2 Posibilidades

Para que cuando se interrumpa el algoritmo dispongamos por lo menos de soluciones presumiblemente "buenas" puede apuntarse los - procedimientos siguientes:

- Cada vez que se encuentre una solución posible mejor que las ya conocidas, exhibirla, o, simplemente exhibir la mejor solución posible obtenida hasta el momento en que se quiere interrumpir los cálculos. En este caso son preferibles los algoritmos que conducen rápidamente a una solución posible (en este sentido, la aplicación de la regla del nudo de evaluación mínima puede ser peor -- que la regla LIFO, que conduce rápidamente a nudos terminales).

La comparación con la mejor cota existente mostrará la posible diferencia con el óptimo. Desde luego, cabe el peligro de que no se haya encontrado ninguna solución posible en el momento de interrumpir el algoritmo o

bien de que la obtenida difiera mucho de la cota sin que podamos saber si lo mediocre es la solución o la cota.

Por supuesto, dado el escaso coste que ello lleva asociado, es preferible que se exhiba toda solución posible mejor que las precedentes, el instante en que aparece (contado desde el comienzo del algoritmo) y la mejor cota presente en aquel momento; de estos datos podrá obtenerse una idea sobre lo que podría haber ocurrido de no existir la limitación de tiempo.

- Introducir reglas de eliminación de nudos basadas en consideraciones razonables: Se puede, por ejemplo, limitar la memoria destinada a los vértices. Si un vértice sobrepasa dicha capacidad, se elimina el de peor cota entre los existentes y se coloca el nuevo vértice, si no es el eliminado, en su lugar. ROY y BENAYOUN /7/ hablan de una capacidad de 1.500 vértices y eliminación de los sobrantes. En las eliminaciones será preciso conservar la mejor cota eliminada, a fin de determinar al final del algoritmo si disponemos del óptimo con toda seguridad o no.

Otro procedimiento consistirá en la eliminación de los peores vértices en cada separación, o bien en la conservación de sólo los mejores vértices en cada separación. Normalmente este procedimiento obligará a trabajar de forma distinta en los niveles altos del árbol y en los bajos, ya que en los altos los procedimientos de evaluación acostumbra a ser especialmente ineficaces. Una aplicación de este método consiste, por ejemplo, en conservar en cada una de las separaciones el vértice de cota siguiente a la mejor, prosiguiendo la exploración por el vértice de mejor cota. Una vez hallada la primera solución posible (que corresponde a una regla heurística de óptimos locales) un examen de las cotas permite proseguir la exploración en otras direcciones, si parece prometedor (ver por ejemplo OLIVELLA /8/).

Un refinamiento en la eliminación de vértices se produce cuando el vértice candidato a la eliminación es dotado de sus siguientes hasta cierto nivel relativo (diferencia de niveles) y la decisión definitiva se --

adopta a partir de las cotas de dichos siguientes.

Otra variante se produce cuando la función de evaluación de los vértices no es estrictamente una cota, sino el valor de una solución posible obtenida a partir del vértice mediante reglas heurísticas muy simples -- (ver CONWAY, MAXWELL y MILLER /9/), o bien un indicador de la eficacia del algoritmo -- hasta el momento. (Ver ELIAS, GARRIGA y KIRCHNER /10/).

También cabe la posibilidad de utilizar un valor de ϵ que sea función del nivel de los nudos (grande para niveles bajos, es decir, los más cercanos a la raíz y menor en los altos); ello permite eliminar, en general, muchos nudos, pero se ha de tener en cuenta que sólo se puede garantizar que la solución así obtenida es ϵ_M -óptima, siendo ϵ_M el máximo de los valores de ϵ que se haya utilizado.

- Otra solución, en principio poco interesante, para no sobrepasar un tiempo prefijado, T , es la que sugieren LAWLER y WOOD /5/: aumento, con el tiempo, de ϵ en la siguiente forma (expresando ϵ en valor relativo):

Hasta $1/2 T$	$\epsilon = 0$
De $1/2 T$ a $3/4 T$	$\epsilon = 0,05$
De $3/4 T$ a $7/8 T$	$\epsilon = 0,10$
De $7/8 T$ a $15/16 T$	$\epsilon = 0,15$
.....

Así se garantiza que el algoritmo se interrumpe dentro del periodo T , (aunque quizá no obtengamos ninguna solución posible).

4. REFERENCIAS

- /1/ AGIN, N. "Optimum seeking with branch & bound". Management Science V.13, N°4, - 1966.
- /2/ HERVE, PH. "Les procédures arborescentes d'optimisation". RFIRO, V.3, 1968.
- /3/ ROY, B. "Procédures d'exploration par séparation et evaluation (P.S.E.P. et P.S.E.S.)" RFIRO, V.1, 1969.
- /4/ BENAYOUN, R., ROY, B., TERGNY, S. "De -

la procédure S.E.P. au programme OPHE--
LIE mixte". Metra, Vol. IX, N°1, 1970.

- /5/ LAWLER, E.L., WOOD, D.E. "Branch and --
bound methods: a survey". Operations Re-
search, V. 14, N°4, 1966.
- /6/ GINSBURGH, V., VAN PETERSEN, A. "un al-
gorithme de programmation quadratique -
en variables binaires". RFIRO, V.1, 1969.
- /7/ ROY, B., BENAYOUN, R. "Programmation li-
néaire en variables bivalentes et conti-
nues sur un graphe (le programme POLIGA-
MI)". Metra. V.VI, N°4, 1967.
- /8/ OLIVELLA, L. "El problema del taller me-
cánico". Proyecto fin de estudios.
- /9/ CONWAY, R.W., MAXWELL, W.L., MILLER, L.
W. Theory of Scheduling. Addison Wes-
ley, 1967.
- /10/ ELIAS, A., GARRIGA, J., KIRCHNER, X. -
"Ordenación de trabajos en un ordenador
con sistema operativo en discos (D.O.S)"
Proyecto fin de estudios, 1972.

5. NOTAS

¹Planteado así es un problema de mínimo. Pa-
ra convertirlo en uno de máximo bastaría -
cambiar el signo de \leq por \geq y \geq por \leq en -
las desigualdades anteriores y en todas --
las que aparezcan en lo sucesivo, y con --
cambios de detalle en algunas definiciones.

En la práctica K , será casi siempre el con-
junto de los números reales y K' el de los
reales no negativos.

Se prescinde de la consideración de crite-
rios secundarios. Obsérvese que, si única-
mente imponemos la condición de que estén
comprendidos dentro de determinados lími-
tes, el tenerlos en cuenta equivale, en de-
finitiva, a reducir el espacio E de las so-
luciones posibles, con lo cual, en este ca-
so, el planteamiento sigue siendo el mismo.

Tal como se ha dicho, la aplicación f sólo
está definida en E . De todas formas puede
considerarse definida en F haciendo:

$$f(X) = +\infty$$

$$\forall X \notin E$$

²En la práctica, en general, el proceso de -
separación constituye una partición de E . -
Es decir, además de la condición de conser-
vación se verifica, siendo $V_i=(S_i, i)$ y --
 $V_j=(S_j, j)$ dos descendientes inmediatos de -
otro cualquiera, que $S_i \cap S_j \neq \emptyset$. Aunque, como
hemos dicho, esta condición se cumple casi
siempre, no es necesaria. No obstante, si -
no se cumple, se ha de tomar algunas precau-
ciones adicionales, puesto que en este caso
las soluciones que se van hallando pueden -
ser repetidas; para salvar la dificultad --
basta comparar cada solución que se halla -
con las ya halladas.

Si llamamos nivel de un nudo al número de -
sus ascendientes, generalmente los nudos --
terminales son los que tienen el máximo ni-
vel. Sin embargo no se excluye la possibili-
dad de que aparezca un nudo terminal en ni-
veles bajos, lo cual normalmente será una -
casualidad que puede producirse como conse-
cuencia del proceso de acotación (vg.: en -
un algoritmo de resolución de programas li-
neales en variables binarias el procedimien-
to de acotación puede consistir en resolver
un programa lineal en variables continuas;
evidentemente puede darse el caso de que en
la solución todas las variables sean cero o
uno: entonces podemos aislar esta solución
y generar un nudo que la contenga).

³Cuando un nudo pasa a ser definitivamente -
descartado permanece indefinidamente en es-
te estado (ya que los valores de las solu-
ciones conocidas no pueden aumentar y la co-
ta no puede disminuir). En cambio, un nudo
provisionalmente descartado puede pasar a -
no descartado al hallar una solución (o, --
por supuesto, a definitivamente descartado,
si, por ejemplo, se llega a conocer una nue-
va solución suficientemente mejor que las -
ya conocidas).

Cuando un nudo está definitivamente cerrado
permanece indefinidamente en este estado. -
En cambio un nudo provisionalmente cerrado
puede pasar a definitivamente cerrado y tam-
bién a no cerrado, al hallar una solución.

Que un nudo esté provisionalmente (o defini-
tivamente) cerrado puede interpretarse como
que se sabe que provisionalmente (o defini-

tivamente) no interesa profundizar más en el conocimiento del nudo o que no es posible hacerlo más que a través del examen de sus descendientes no inmediatos.

⁴Cuando $n_s=1$ las cosas se simplifican bastante, ya que desaparece la distinción entre provisional y definitivo (puesto que todo nudo que no está descartado está definitivamente descartado) y además no hay soluciones halladas (si se halla una solución el algoritmo se puede dar ya por finalizado).

Obsérvese que, tal como está planteado el algoritmo, las soluciones que, sucesivamente, se va extrayendo son ϵ -óptimas respecto a las no extraídas, lo cual no implica que el orden de extracción sea el que correspondería ordenando las soluciones por el valor de su función objetivo.