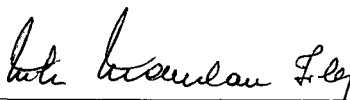


PROBLEMA DE LOCALIZAÇÃO CAPACITADO:  
FORMULAÇÕES, RELAXAÇÕES E SOLUÇÕES

**Hernaldo Reinoso Alarcón**

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS (D.Sc.) EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Aprovada por:



---

Prof. Nelson Maculan Filho  
(Presidente)



---

Prof<sup>a</sup> Ana Friedlander de Martinez Pérez



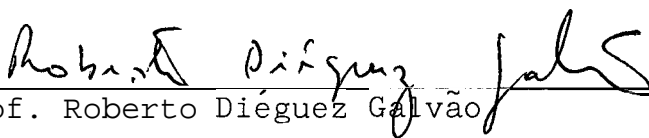
---

Prof. Celso Carneiro Ribeiro



---

Prof. Claudio Thomas Bornstein



---

Prof. Roberto Diéguez Galvão

RIO DE JANEIRO, RJ - BRASIL

Agosto DE 1988

REINOSO ALARCON, HERNALDO

Problema de Localização Capacitado: Formulações, relaxações e Soluções (Rio de Janeiro), 1988.

VII , 79 p. (COPPE/UFRJ, D.Sc., Engenharia de Sistemas e Computação, 1988).

Tese - Universidade Federal do Rio de Janeiro, COPPE.

1. Problemas de Localização Capacitados, Programação Linear Inteira. I. COPPE/UFRJ II. Título (Série).

AGRADECIMENTOS

Ao Professor Nelson Maculan pela sua orientação e apoio durante o desenvolvimento deste trabalho e nas atividades do meu curso.

À Ednéa Maria Nascimento cujo constante estímulo foi determinante no sucesso deste trabalho.

AO CNPq pelo importante apoio financeiro.

RESUMO DA TESE APRESENTADA A COPPE/UFRJ COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS (D.Sc.)

PROBLEMA DE LOCALIZAÇÃO CAPACITADO:  
FORMULAÇÕES, RELAXAÇÕES E SOLUÇÕES

Hernaldo Reinoso Alarcón  
Agosto de 1988

ORIENTADOR: Prof. Nelson Maculan Filho

PROGRAMA : Engenharia de Sistemas e Computação

Nesta tese propõe-se um algoritmo exato do tipo "branch and bound", para resolver um problema de localização capacitado onde existe a restrição adicional que cada cliente deve ser atendido por um Único armazém. Neste algoritmo é utilizado um método de ajuste de multiplicadores para calcular os limites. São propostos também, alguns métodos heurísticos para obter soluções aproximadas do problema.

Adicionalmente, são discutidas relaxações lagrangeanas baseadas em novas formulações. Em particular, descreve-se um esquema de decomposição lagrangeana que pode fornecer limites consideravelmente mais fortes que aqueles fornecidos por métodos lagrangeanos clássicos.

São apresentados também, alguns resultados computacionais que mostram a viabilidade dos métodos descritos.

ABSTRACT OF THESIS PRESENTED TO COPPE/UFRJ AS PARTIAL FULFILLMENT  
OF THE REQUIREMENTS FOR DEGREE OF DOCTOR OF SCIENCE (D.Sc.)

CAPACITATED LOCATION PROBLEM: FORMULATIONS,  
RELAXATIONS AND SOLUTIONS

**Hernaldo Reinoso Alarcón**  
August, 1988

CHAIRMAN : Prof. Nelson Maculan Filho  
DEPARTMENT: Systems Engineering and Computer Science

In this thesis we present a branch and bound algorithm for the capacitated location problem, where each customer must be supplied from one and only one warehouse. In this algorithm bounds are obtained using a multiplier adjustment method. We also present several heuristics to generate feasible solutions.

Furthermore are discussed Lagrangean relaxations based upon new formulations. In particular, we describe a model of Lagrangean decomposition which yield stronger bounds than the bounds obtained using classical Lagrangean relaxation.

Some computational results are presented which show reasonable performance of the proposed methods.

## ÍNDICE

	<u>Pág.</u>
<b>CAPÍTULO I - INTRODUÇÃO.....</b>	<b>1</b>
1.1 - Definição e Formulação do Problema.....	1
1.2 - Notação.....	4
1.3 - Reformulação do Problema.....	
<b>CAPÍTULO II - OBTENÇÃO DE LIMITES SUPERIORES.....</b>	<b>6</b>
II.1 - Introdução.....	6
11.2 - Método de Ajuste de Multiplicadores.....	7
11.3 - Exemplos.....	17
<b>CAPÍTULO III - OBTENÇÃO DE LIMITES INFERIORES.....</b>	<b>23</b>
III.1 - Introdução.....	23
111.2 - Algoritmo Heurístico HRPLC1.....	24
111.3 - Algoritmo Heurístico HRPLC2.....	27
111.4 - Algoritmo Heurístico HRPLC3.....	29
<b>CAPÍTULO IV - UM MÉTODO EXATO.....</b>	<b>31</b>
IV.1 - Introdução.....	31
IV.2 - O Esquema "Branch and Bound".....	33
IV.3 - Metodos de Redução.....	37
IV.4 - Um Exemplo.....	39
IV.5 - Resultados Computacionais.....	43

<b>CAPÍTULO V – MÉTODOS DE DECOMPOSIÇÃO LAGRANGEANA PARA OBTEN- LIMITES SUPERIORES MAIS FORTES.....</b>	<b>46</b>
V.1 – Introdução.....	46
V.2 – Metodo 1.....	47
V.3 – Metodo 2.....	50
V.4 – Um Exemplo.....	57
V.5 – Aplicação ao Problema de Localização Capacitado.....	62
<b>CAPÍTULO VI – LIMITES SUPERIORES: OUTRAS RELAXAÇÕES.....</b>	<b>67</b>
VI.1 – Relaxações Lagrangeana Baseada num Problema de Local- ização não Capacitado.....	67
VI.2 – Relaxação Lagrangeana Baseada num Problema de Local- ização Capacitado.....	68
<b>CAPÍTULO VII – CONCLUSÕES.....</b>	<b>72</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>74</b>

## CAPÍTULO I

## INTRODUÇÃO

## I.1 - DEFINIÇÃO E FORMULAÇÃO DO PROBLEMA

O objetivo deste trabalho é o desenvolvimento de um método exato de solução de um problema de localização capacitado, que pode ser definido da seguinte maneira: seja  $I = \{1, 2, \dots, m\}$  um conjunto de  $m$  armazéns e  $J = \{1, 2, \dots, n\}$  um conjunto de  $n$  clientes. Cada cliente demanda  $q_j$  unidades de um certo tipo de mercadoria, para a qual, cada armazém tem capacidade  $b_i > 0$ . Para todo  $i \in I$  existe um custo fixo  $f_i$  de usar o armazém  $i$ , para atender a demanda de um ou mais clientes, e um custo variável  $c_{ij}$  de atender a demanda do cliente  $j$  no armazém  $i$ . O problema consiste em encontrar uma alocação de clientes a armazéns cujo custo total seja mínimo. Aqui será considerada a restrição em que a demanda de cada cliente deve ser totalmente atendida por um Único armazém.

O problema acima descrito pode ser colocado como um problema de programação linear inteira da seguinte forma:

$$(Q) \min \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i \quad (I.1)$$

$$\text{s.a} \quad \sum_{i \in I} x_{ij} = 1, \quad j \in J \quad (I.2)$$

$$\sum_{j \in J} q_j x_{ij} \leq b_i y_i, \quad i \in I \quad (I.3)$$

$$L \leq \sum_{i \in I} y_i \leq U, \quad (I.4)$$

$$x_{ij}, y_i \in \{0, 1\}, \quad i \in I, \quad j \in J. \quad (I.5)$$



onde

$$x_{ij} = \begin{cases} 1 & \text{se o cliente } j \text{ é alocado ao armazém } i, \\ 0 & \text{caso contrário,} \end{cases}$$

e

$$y_i = \begin{cases} 1 & \text{se o armazém } i \text{ é alocado a algum cliente } j, \\ 0 & \text{caso contrário.} \end{cases}$$

As restrições (1.2), junto com  $x_{ij} \in \{0,1\}$ , garantem que a demanda de cada cliente é totalmente satisfeita por um Único armazém. Por outro lado, as restrições (1.3) asseguram que a capacidade de cada armazém não é ultrapassada.

Em (1.4) é estabelecida uma restrição no número mínimo ( $L$ ), e máximo ( $u$ ) de armazéns à serem utilizados. No entanto, se rá suposto que esses números ( $L$  e  $U$ ) são tais que a solução do problema (Q) sem (1.4) sempre satisfará essa restrição.

Não é difícil mostrar que o problema (Q) é NP-árduo. Para isto é suficiente observar que o problema de localização não-capacitado, o qual é NP-árduo (CORNEJUOLS et alii (9)), é um caso particular de (Q).

Em relação à viabilidade do problema, uma condição necessária, mas não suficiente, para a existência de alguma solução viável é que a capacidade total dos armazéns seja maior ou igual a demanda total dos clientes, ou seja, que  $\sum_{i \in I} b_i \geq \sum_{j \in J} q_j$ . Esta condição não é suficiente devido à restrição  $x_{ij} \in \{0,1\}$ , a qual implica que cada cliente é atendido por um Único armazém.

Se em (1.4) se estabelece que  $0 \leq x_{ij} \leq 1$ ,  $i \in I$ ,  $j \in J$ , então (Q) se reduz ao clássico problema de localização capacitado, o qual tem sido amplamente estudado na literatura. Ver por exemplo (3), (5), (6), (8) e (39). Por outro lado, se  $f_i = 0$ , para todo  $i \in I$ , então (Q) corresponderá a um Problema de Alocação Generalizado (PAG), para o qual existem métodos eficientes devidos a ROSS e SOLAND (35), MARTELLO e TOTH (26) e (27); e FISHER, JAIKUMAR e VAN WASSENHOVE (14).

Métodos aproximados de solução de (Q) tem sido desenvolvidos por BARCELÓ e CASANOVAS (4) e KLINCEWICZ e LUSS (24). Ambos métodos são construídos na base de relaxações lagrangeanas de (Q). No primeiro caso são relaxadas as restrições de atendimento dos clientes (1.2) para tentar achar uma solução viável a partir do problema lagrangeano, o qual separa em  $m$  problemas da mochila 0-1 independentes. No segundo caso, são relaxadas as restrições de capacidade dos armazéns, obtendo-se como subproblema um problema de localização não capacitado; heurísticas são utilizadas para gerar soluções viáveis a partir dos subproblemas lagrangeanos.

Um método exato é descrito por NEEBE e RAO (32). O método é do tipo "branch and bound", onde os limites inferiores são calculados formulando o problema como um problema de particionamento com muitas colunas. O limite inferior usado é o valor da solução da relaxação de programação linear deste problema de particionamento, obtida utilizando um método de geração de colunas.

O método exato que será proposto nesta tese consiste, também, de um esquema de "branch and bound", onde os limites para o valor da função objetivo são calculados a partir da relaxação lagrangeana das restrições (1.2), e onde o problema dual la-

grangeano é resolvido por um método de ajuste de multiplicadores, derivado daquele usado em (14) para o problema de alocação generalizado.

A estrutura deste trabalho é a seguinte: ainda neste capítulo, na seção 1.2 descreve-se parte da notação que será utilizada. Na seção 1.3 o problema (Q) é colocado sob a forma de maximização; isto facilitará a descrição de alguns dos métodos que serão apresentados. Nos Capítulos II e III serão propostos métodos para calcular limites superiores e inferiores, respectivamente. Estes métodos são incorporados ao esquema de "branch and bound" do Capítulo IV. No Capítulo V é apresentado um novo esquema, chamado decomposição lagrangeana, que pode fornecer limites superiores substancialmente mais fortes (melhores) que na relaxação lagrangeana clássica. Finalmente, no Capítulo VI são discutidos outros métodos de cálculo de limites superiores, e no capítulo VII, as conclusões.

## I.2 - NOTAÇÃO

Será utilizada a seguinte notação, além da que será introduzida, na medida que for necessário, nos próximos capítulos:

(i) O símbolo  $\sum_i$  será usado para indicar  $\sum_{i \in I}$  ou  $\sum_{i=1}^m$ .

(ii) Dado um problema ( $\cdot$ ),  $V(\cdot)$  indicará o valor Ótimo da função objetivo do problema.  $SV(\cdot)$  e  $SO(\cdot)$  denotarão o conjunto de soluções viáveis e o conjunto de soluções Ótimas do problema, respectivamente. O valor de uma solução viável  $x$  não pertencente a  $SO(\cdot)$  será denotado por  $v(x)$ .

### I.3 - REFORMULAÇÃO DO PROBLEMA

Por comodidade, daqui em diante será considerada a versão de maximização de (Q). Para isto é suficiente multiplicar por -1 todos os coeficientes de (1.1) em (Q), e definir

$$p_{ij} = r - c_{ij} \text{ para todo } i \in I \text{ e } j \in J.$$

onde  $r > \max\{c_{ij} \mid i \in I, j \in J\}$ .

Assim o problema pode ser colocado sob a forma seguinte:

$$(P) \quad \max \sum_i \sum_j p_{ij} x_{ij} - \sum_i f_i y_i,$$

s.a. (I.2), (I.3) e (I.4).

Obviamente  $SV(P) = SV(Q)$ . Por outro lado, dado  $(x, y) \in SO(P)$  então

$$V(Q) = nr - V(P).$$

Nesta formulação,  $p_{ij}$  poderia ser interpretado como o "lucro" por atender a demanda do cliente  $j$  no armazém  $i$ . Portanto, agora o problema consiste em encontrar uma alocação de clientes aos armazéns, cujo lucro total (lucro líquido, lucro menos custo fixo total) seja máximo.

## CAPÍTULO II

## OBTENÇÃO DE LIMITES SUPERIORES

## II.1 - INTRODUÇÃO

Entre os métodos que poderiam ser usados para o cálculo de limites superiores, aqui foi escolhida a técnica de relaxação lagrangeana. Uma excelente exposição da teoria e prática desta técnica e seu uso em métodos de "branch and bound" pode ser encontrada em GEOFFRION (16), FISHER (12) e (13) e SHAPIRO (37).

De um modo geral, dado um problema (R) da forma:

$$\begin{aligned} (R) \quad & \max \quad fx \\ & \text{s.a.} \quad Ax \leq b \\ & \quad \quad Cx \leq d \\ & \quad \quad x \geq 0 \text{ e inteiro,} \end{aligned}$$

onde os vetores  $x$ ,  $f$ ,  $b$ ,  $d$  e as matrizes  $A$  e  $C$  tem dimensões apropriadas, pode-se definir a relaxação lagrangeana de (R) com respeito a  $Ax \leq b$  e a um vetor  $\lambda \geq 0$ , como

$$\begin{aligned} (RL_{\lambda}) \quad & \max \quad fx + \lambda(b - Ax) \\ & \text{s.a.} \quad Cx \leq d \\ & \quad \quad x \geq 0 \text{ é inteiro.} \end{aligned}$$

Pode-se demonstrar que  $V(RL_{\lambda}) \geq V(R)$ , para todo  $\lambda \geq 0$  (ver (16)). Esta propriedade implica que o ideal é escolher o  $\lambda$  que resolve o seguinte problema dual lagrangeano

$$(DR) \quad \min_{h \geq 0} V(LR_h)$$

Também é possível demonstrar que  $V(DR) \leq V(\bar{R})$ , onde  $(\bar{R})$  é a relaxação de programação linear de  $(R)$ . Uma outra propriedade importante é que se o problema  $(RL_A)$  satisfaz a Propriedade de Integralidade então  $V(DR) = V(\bar{R})$  (ver (16)).

As propriedades estabelecidas acima fornecem alguns critérios para escolher entre as possíveis relaxações do problema. No caso do problema estudado nesta tese será definida uma relaxação lagrangeana das restrições (I.2).

Alguns dos métodos que podem ser utilizados para resolver o dual lagrangeano são os seguintes:

- (i) método do subgradiente;
- (ii) métodos de programação linear, e
- (iii) métodos de ajuste de multiplicadores.

O sucesso do método (iii) aplicado ao problema de alocação generalizado por FISHER et alii (14), e a relação deste problema com o problema de localização capacitado formulado aqui, tem motivado seu uso no método exato que será desenvolvido nos próximos capítulos.

## II.2 - MÉTODO DE AJUSTE DE MULTIPLICADORES

Antes de desenvolver o método de ajuste de multiplicadores que se utilizará para obter limites superiores, será necessária uma reformulação do problema. Para isto pode-se definir para todo  $i \in I$

$$z_i = 1 - y_i,$$

de modo que

$$z_i = \begin{cases} 1 & \text{se } y_i = 0 \\ 0 & \text{se } y_i = 1. \end{cases}$$

Substituindo em (P) obtêm-se o problema

$$(P') \max \sum_i \sum_j p_{ij} x_{ij} + \sum_i f_i z_i - \sum_i f_i \quad (\text{II.1})$$

$$\text{s.a.} \quad \sum_i x_{ij} = 1, \quad j \in J \quad (\text{II.2})$$

$$m - U \leq \sum_i z_i \leq m - L \quad (\text{II.3})$$

$$\sum_j q_j x_{ij} + b_i z_i \leq b_i, \quad i \in I \quad (\text{II.4})$$

$$x_{ij}, z_i \in \{0,1\}, \quad i \in I, \quad j \in I \quad (\text{II.5})$$

O problema (P') pode ser interpretado como um problema de alocação generalizado onde existem  $n+1$  "clientes" que devem ser alocados a  $m$  "armazéns" e onde um dos clientes (o representado pelas variáveis  $z_i$ ) pode ser alocado à  $h$  armazéns ( $m-U \leq h \leq m-L$ ). Neste caso o lucro de alocar este cliente ao armazém  $i$  é  $f_i$ , sendo  $b_i$  a correspondente demanda.

De agora em diante será suposto que  $U = m$  e que  $L$  é o número mínimo de armazéns que devem ser usados para satisfazer a demanda total dos clientes. Ou seja  $L$  é o valor de uma solução Ótima do problema seguinte:

$$\begin{aligned}
 \text{(S)} \quad & \min \sum_i \delta_i \\
 \text{s.a} \quad & \sum_i b_i \delta_i \geq \sum_j q_j \\
 & \delta_i \in \{0,1\}.
 \end{aligned}$$

Este problema é um problema da mochila 0-1 que pode ser resolvido por inspeção.

A seguir define-se uma relaxação lagrangeana de (P') dualizando as restrições (II.2) e (II.3), associando-lhes multiplicadores  $\{\lambda_j \geq 0 | j \in J\}$ , e  $\mu \geq 0$ , respectivamente. O problema resultante é

$$\begin{aligned}
 \text{(LR}_{\mu, \lambda}^i) \quad & \max \sum_i \sum_j (p_{ij} - \lambda_j) x_{ij} + \sum_i (f_i - \mu) z_i + \sum_j \lambda_j + (m-L) \mu - \\
 & - \sum_i f_i \\
 \text{s.a} \quad & \text{(II.4) e (II.5)}.
 \end{aligned}$$

Dados  $\mu$  e  $\lambda$ , este problema se decompõe em  $m$  problemas de mochila 0-1 independentes, da forma

$$\begin{aligned}
 \text{(K}_{\mu, \lambda}^i) \quad & \max \sum_j (p_{ij} - \lambda_j) x_{ij} + (f_i - \mu) z_i \\
 \text{s.a} \quad & \sum_j q_j x_{ij} + b_i z_i \leq b_i \\
 & x_{ij}, z_i \in \{0,1\}, j \in J.
 \end{aligned}$$

Pelas propriedades enunciadas no começo deste capítulo, tem-se que, para quaisquer  $\lambda_j \geq 0$ ,  $j \in J$  e  $\mu \geq 0$ , a quantidade



$$\sum_i V(K_{\mu, \lambda}^i) + \sum_j \lambda_j + (m-L) \mu - \sum_i f_i$$

é um limite superior para  $V(P')$  ou, equivalentemente, para  $V(P)$ .

Agora o objetivo será resolver o problema dual lagrangeano de  $(P')$ , ou seja, encontrar valores  $h$  e  $\mu$  tais que sejam uma solução Ótima do problema

$$(D) \min_{\mu, \lambda} V(LR_{\mu, \lambda}) .$$

Para resolver (D) propõe-se o seguinte método de ajuste de multiplicadores. Este método consiste basicamente de duas partes:

(i) Inicializar  $h_j$  como o segundo máximo dos  $p_{ij}$ ,  $i \in I$ , e  $\mu$  como o  $(m-L+1)$ -ésimo máximo dos  $f_i$ ,  $i \in I$ . Assim se terá no máximo um  $i \in I$  tal que  $p_{ij} - h_j > 0$ , e, no máximo  $(m-L)$   $i \in I$  tal que  $f_i - \mu > 0$ .

Portanto uma solução Ótima do problema lagrangeano para estes valores de  $\mu$  e  $h$ , satisfará as condições

$$\sum_i x_{ij} \leq 1, \text{ para todo } j \in J, \text{ e } \sum_i z_i \leq m-L.$$

Se  $\sum_i x_{ij} = 1$ , para todo  $j \in J$ , e  $\sum_i z_i = m-L$  esta solução é uma solução Ótima de  $(P')$ .

(ii) Se em (i) não é possível conseguir uma solução Ótima, sob certas condições pode-se escolher, para decrescer algum multiplicador  $h_{j^*}$  (ou  $\mu$ ) de tal maneira que a nova solução lagrangeana satisfaz

$$\sum_{\mathbf{i}} x_{\mathbf{i}j^*} = 1, \quad \sum_{\mathbf{i}} z_{\mathbf{i}} \leq m-L \quad \text{e} \quad \sum_{\mathbf{i}} x_{\mathbf{i}j} \leq 1 \quad \text{para todo } j \in J - \{j^*\}$$

$$\text{(ou } \sum_{\mathbf{i}} z_{\mathbf{i}} \leq m-L \quad \text{e} \quad \sum_{\mathbf{i}} x_{\mathbf{i}j} \leq 1 \quad \text{para todo } j \in J \text{)}.$$

Se as condições especificadas se satisfazem, o multiplicador escolhido é diminuído e se continuam as iterações procurando um novo  $\lambda_{j^*}$ . O método termina quando as condições requeridas não podem ser satisfeitas.

Na descrição formal do algoritmo será necessária alguma notação. Dado um conjunto de números  $\{g_{\mathbf{i}} | \mathbf{i} \in I\}$ , define-se

$$\arg \max \{g_{\mathbf{i}}\} = \text{o menor índice } \mathbf{i}_0 \in I$$

$$\text{tal que } g_{\mathbf{i}_0} = \max \{g_{\mathbf{i}} | \mathbf{i} \in I\}$$

$$\max_2 \{g_{\mathbf{i}}\} = \max_{\mathbf{i}} \{g_{\mathbf{i}} | \mathbf{i} \in I, \mathbf{i} \neq \arg \max \{g_{\mathbf{i}}\}\}$$

$$\arg \max_2 \{g_{\mathbf{i}}\} = \text{o menor índice } \mathbf{i}_0 \in I, \mathbf{i}_0 \neq \arg \max \{g_{\mathbf{i}}\},$$

$$\text{tal que } g_{\mathbf{i}_0} = \max_2 \{g_{\mathbf{i}}\}.$$

Precisamos definir também

$$\max_P \{g_{\mathbf{i}}\} = \max \{g_{\mathbf{i}} | \mathbf{i} \in I'\},$$

$$\arg \max_P \{g_{\mathbf{i}}\} = \text{o menor índice } \mathbf{i}_0 \in I'$$

$$\text{tal que } g_{\mathbf{i}_0} = \max_P \{g_{\mathbf{i}}\},$$

onde  $p = m - L + 1$  e  $I' = I - \{i \mid i = \arg \max_2 g, \dots, \arg \max_{p-1} g\}$

As definições para  $\min_2 \{g_i\}$ ,  $\arg \min \{g_i\}$  e  $\arg \min_2 \{g_i\}$  são análogas às anteriores.

A notação  $KP(i, K)$  será usada para representar o seguinte problema da mochila:

$$\max \sum_{j \in K} (p_{ij} - \lambda_j) x_{ij}$$

$$\text{s.a.} \quad \sum_{j \in K} q_j x_{ij} \leq b_i$$

$$x_{ij} \in \{0, 1\}, \quad j \in K.$$

Por outro lado o par  $(\tilde{x}, z)$  denotará uma solução viável de  $(P')$ , e  $ZD_i$  o valor da solução do problema  $(K_{\mu, \lambda}^i)$ .

O algoritmo proposto é o seguinte:

**Algoritmo MAM:** Método de ajuste de multiplicadores

**Passo 1.** Inicializar

$$A_j := \max_{i \in I} \{c_{ij}\}, \quad x_{ij} := 0, \quad \text{para todo } i \in I, j \in J, \text{ e}$$

$$p := \max_{i \in I} \{f_i\}, \quad z_i := 0, \quad \text{para todo } i \in I.$$

**Passo 2.** Definir  $J_i^+ = \{j \in J \mid p_{ij} - \lambda_j > 0\}$ .

Para todo  $i \in I$  efetuar

$$(x_{ij}) := \text{solução do problema } KP(i, J_i^+).$$

Se  $V[\text{KP}(i, J_i^+)] < f_i - \mu$  então efetuar

$$z_i := 1;$$

$$x_{ij} := 0, \text{ para todo } j \in J;$$

$$ZD_i := f_i - \mu;$$

Caso contrário,  $z_i := 0$

$$ZD_i := V[\text{KP}(i, J_i^+)].$$

**Passo 3:** Teste de otimalidade

$$\text{Definir } \bar{J} = \{j \in J \mid \sum_i x_{ij} = 0\}$$

Se  $\bar{J} = \emptyset$ , se  $\sum_i z_i = m-L$ , parar (Ótimo)

se  $\sum_i z_i < m-L$ , atualizar  $(\hat{x}, \hat{z})$ . Ir ao passo 10.

Se  $\bar{J} \neq \emptyset$ , ir ao passo 4.

**Passo 4:** Aplicação de uma heurística.

$$\text{Definir } J_i^0 = \{j \in \bar{J} \mid p_{ij} - \lambda_j = 0\};$$

$$T_j^0 = \{i \in I \mid j \in J_i^0, z_i = 0\};$$

$$\bar{b}_i = \begin{cases} b_i - \sum_j q_j x_{ij}, & \text{se } z_i = 0, \\ 0 & , \text{ se } z_i = 1. \end{cases}$$

Efetuar

$(x_{ij}) :=$  solução heurística do problema

$$\text{máx } \sum_{i \in I_j^{\circ}} \sum_{j \in J_i^{\circ}} p_{ij} x_{ij}$$

$$\text{s.a } \sum_{i \in I_j^{\circ}} x_{ij} = 1, j \in \bar{J}$$

$$\sum_{j \in J_i^{\circ}} q_j x_{ij} \leq \bar{b}_i, i \in I$$

$$x_{ij} \in \{0,1\}, j \in J_i^{\circ}, i \in I_j^{\circ}$$

**Passo 5:** Teste de otimalidade

Definir  $\bar{J} = \{j \in J \mid \sum_i x_{ij} = 0\}$ .

Se  $\bar{J} = \emptyset$ , se  $\sum_i z_i = m-L$ , parar (Ótimo)

se  $\sum_i z_i < m-L$ , atualizar  $(\tilde{x}, \hat{z})$  e ir ao passo 10

Se  $\bar{J} \neq \emptyset$ , fazer  $J_v := \bar{J}$  e ir ao passo 6.

**Passo 6:** Escolher algum  $j^* \in J_v$  e para todo  $i \in I$  efetuar

$$\Delta_i := ZD_i - (p_{ij^*} - \lambda_{j^*}) - V[KP(i, J - \{j^*\})];$$

$$J_v := J_v - \{j^*\};$$

Passo 7: Se  $\min_2 \{\Delta_i\} \leq 0$ , ir ao passo 8,

Caso contrário efetuar

$$\lambda_{j^*} := \lambda_{j^*} - \min_2 \{\Delta_i\};$$

$$i^* := \arg \min \{\Delta_i\};$$

$$x_{i^*j} := \begin{cases} 1 & \text{se } j = j^*, \\ \text{solução de KP}(i^*, J - \{j^*\}), & \text{se } j \neq j^*; \end{cases}$$

$$z_{i^*} := 0;$$

$$ZD_{i^*} := (p_{i^*j^*} - \lambda_{j^*}) + V[\text{KP}(i^*, J - \{j^*\})].$$

Passo 8: Se  $\sum_i x_{ij} \leq 1$  para todo  $j \in J$ , e  $\sum_i z_i \leq m-L$ , então voltar ao passo 3. (Neste caso o limite superior diminuiu em  $A_{i^*}$ ). Senão, restabelecer  $\lambda_{j^*}$ ,  $(x_{i^*j})$  e  $ZD_{i^*}$  aos valores que tinham no começo do passo 7.

Passo 9: Se  $J_V \neq \emptyset$  voltar ao passo 4;

Se  $J_V = \emptyset$  se  $\sum_i z_i = m-L$ , parar  
se  $\sum_i z_i < m-L$ , ir ao passo 10.

Passo 10: Definir  $I_0 = \{i \in I \mid z_i = 0\}$ ;

Para todo  $i \in I_0$  efetuar

$$\Delta_i := ZD_i - (f_i - \mu).$$

Se  $\min_2 \{\Delta_i | i \in I_0\} \leq 0$ , parar.

Caso contrário efetuar

$$\mu := \mu - \min_2 \{\Delta_i | i \in J_0\};$$

$$i^* := \arg \min \{\Delta_i | i \in I_0\};$$

$$x_{i^*j} := \text{para todo } j \in J;$$

$$z_i^* := 1;$$

$$ZD_i := \begin{cases} f_i - \mu & \text{para } i = i^* \text{ e para todo } i \text{ tal que } z_i = 1, \\ 0 & \text{caso contrário.} \end{cases}$$

**Passo 11:** Definir  $J = \{j \in J | \sum_i x_{ij} = 0\}$ . Voltar ao passo 4.

A heurística aplicada no passo 4 consiste em colocar os índices  $ij$  na ordem não-crescente dos  $p_{ij}$  e, se  $j \in J_i^0$  fazer  $x_{ij} := 1$  se com isto não ultrapassar a capacidade da "mochila"  $i$ . Outras heurísticas podem ser encontradas em MARTELLO e TOTH (26) e (27).

Os problemas da mochila 0-1 dos passos 2 e 6 foram resolvidos usando o método de MARTELLO e TOTH descrito em (25). Outros métodos eficientes podem ser encontrados em FAYARD e PLATEAU (11) e MARTELLO e TOTH (28).

## II.3 - EXEMPLOS

## II.3.1 - Exemplo 1

Seja um problema de localização capacitado com os seguintes dados:

$$I = \{1, 2, 3, 4\}, \quad J = \{1, 2, 3, 4, 5\}$$

$$(p_{ij}) = \begin{bmatrix} 24 & 16 & 19 & 15 & 20 \\ 19 & 23 & 5 & 22 & 20 \\ 24 & 23 & 12 & 1 & 17 \\ 24 & 13 & 22 & 11 & 15 \end{bmatrix}, \quad (b_i) = \begin{bmatrix} 20 \\ 20 \\ 15 \\ 20 \end{bmatrix}, \quad (f_i) = \begin{bmatrix} 10 \\ 30 \\ 40 \\ 20 \end{bmatrix},$$

$$(q_j) = (10 \quad 9 \quad 13 \quad 12 \quad 6).$$

Antes de aplicar o MAM, calcula-se o número mínimo de armazéns para atender totalmente a demanda dos clientes. Este número é obtido resolvendo o problema

$$(S) \quad \text{Min} \quad \sum_{i=1}^4 \delta_i$$

$$\text{s.a} \quad 20\delta_1 + 20\delta_2 + 15\delta_3 + 20\delta_4 \geq 50$$

$$\delta_i \in \{0,1\}, \quad i = 1, 2, 3, 4$$

Não é difícil ver que  $V(S) = 3 = L$ . Ou seja,  $m-L = 1$ .

Os passos iniciais do algoritmo produzem a seguinte in-  
formação:



$$(\lambda_j) = (24 \quad 23 \quad 19 \quad 15 \quad 20),$$

$$\mu = 30,$$

$$(p_{ij}^{-\lambda_j}) = \begin{bmatrix} 0 & -7 & 0 & 0 & 0 \\ -5 & 0 & -14 & 7 & 0 \\ 0 & 0 & -7 & -14 & -3 \\ 0 & -10 & 3 & -4 & -5 \end{bmatrix}, \quad (f_i^{-\mu}) = \begin{bmatrix} -20 \\ 0 \\ 10 \\ -10 \end{bmatrix},$$

$$(x_{ij}) = \begin{bmatrix} 1^* & 0 & 0 & 0 & 1^* \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}, \quad (z_i) = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \quad (ZD_i) = \begin{bmatrix} 0 \\ 7 \\ 10 \\ 3 \end{bmatrix}.$$

Com esta informação, o limite superior é 51.

Os números marcados com (\*) na matriz  $(x_{ij})$  representam as alocações feitas usando a heurística no passo 4. Neste ponto, não se satisfaz nenhuma condição de parada, portanto é efetuada uma iteração para tentar melhorar o atual limite.

O Único cliente tal que  $\sum_j x_{ij} = 0$  é  $j = 2$ . Então estabelece-se  $j^* = 2$  e calculam-se os  $\Delta_i$  (passos 5 e 6). Os valores obtidos são

$$(\Delta_i) = (7 \quad 7 \quad 10 \quad 13).$$

Como  $\min_2 \{\Delta_i\} = 7 > 0$  (passo 7) efetua-se

$$\lambda_2 := 23 - 7 = 16,$$

$$i^* := 1 = \arg \min \{\Delta_i\},$$

$$x_{1j} := \begin{cases} 1 & \text{se } j = 2 \\ 0 & \text{se } j \neq 2 \text{ (= solução de KP}(2, J - \{2\})) \end{cases},$$

$$ZD_i := p_{i^*j^*} - \lambda_{j^*} + V[\text{KP}(i^*, J - \{j^*\})],$$

agora o limite superior é

$$51 - \Delta_{i^*} = 51 - \Delta_1 = 44.$$

A nova solução satisfaz  $\sum_i x_{ij} \leq 1$  para todo  $j$  e  $\sum_j z_i \leq 1$ , portanto continua-se no passo 3 para o teste de otimalidade e, logo após, no passo 4 para aplicação da heurística. Obtêm-se assim

$$(x_{ij}) = \begin{bmatrix} 1^* & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1^* \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}, \quad (z_i) = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}.$$

Esta solução satisfaz a condição de otimalidade, portanto o algoritmo termina aqui. O valor desta solução é 44, o qual, obviamente, coincide com o valor atual do limite superior.

### II.3.2 - Exemplo 2

O exemplo a seguir foi construído para ilustrar uma outra condição de parada do algoritmo.

Considere-se o mesmo exemplo anterior mudando apenas as capacidades dos armazéns e seus custos fixos. Os novos valores são

$$(b_i) = \begin{bmatrix} 20 \\ 20 \\ 25 \\ 25 \end{bmatrix} \quad \text{e} \quad (f_i) = \begin{bmatrix} 30 \\ 20 \\ 0 \\ 10 \end{bmatrix},$$

Neste caso  $m-L = 2$ . A seguir se apresenta um resumo das informações produzidas pelo algoritmo durante as iterações.

### 1ª Iteração:

$(\lambda_j)$  e  $(p_{ij} - \lambda_j)$  tem os mesmos valores que na situação inicial do exemplo anterior.

Por outro lado

$$\mu = 20 \quad \text{e} \quad (f_i - \mu) = (10 \quad 0 \quad -20 \quad -10)^T.$$

O limite superior é 101 e está associado com a seguinte solução:

$$(x_{ij}) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1^* \\ 1^* & 1^* & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad \text{e} \quad (z_i) = \begin{bmatrix} 0^* \\ 0 \\ 1 \\ 0 \end{bmatrix}.$$

Esta solução é viável mas não Ótima porque  $\sum_{i=1}^4 z_i = 1 < 2$ .  
O valor desta solução é 83.

### 2ª Iteração:

Pela informação da iteração anterior, o algoritmo é reiniciado no passo 10, obtendo-se

$$(\Delta_i) = (\text{---}, 7, 20, 13),$$

$$\mu = 20 - 13 = 7, \quad i^* = 2,$$

$$x_{i^*j} = 0, \text{ para todo } j \in J,$$

$$z_{i^*} = 1, \quad ZD_{i^*} = 20 - 7 = 13,$$

$$\text{limite superior} = 101 - A_{i^*} = 101 - 7 = 94.$$

### 3ª Iteração:

$$j^* = 4, \quad J_V = \{5\},$$

$$(\Delta_i) = (23, 6, 14, 4), \quad i^* = 4,$$

$$\lambda_4 = 15 - 6 = 9, \quad z_4 = 0,$$

$$(x_{i^*j}) = (0 \quad 0 \quad 1 \quad 1 \quad 0),$$

$$ZD_{i^*} = (11 - 9) + 3 = 5,$$

$$\text{limite superior} = 94 - A_{i^*} = 94 - 4 = 90.$$

### 4ª Iteração:

$$j^* = 5, \quad J_V = \emptyset,$$

$$(\Delta_i) = (17, 0, 3, 7), \quad i^* = 2,$$

$$(x_{i^*j}) = (0 \quad 0 \quad 0 \quad 1 \quad 1).$$

Nesta altura, como  $\sum_{\mathbf{i}} x_{\mathbf{i}j^*} = 2 > 1$ , então o algoritmo deverá escolher um novo  $j^* \in J_{\mathbf{v}}$ , mas como  $J_{\mathbf{v}} = \emptyset$ , isto não é possível. Assim, se satisfaz uma das condições de parada.

O limite superior obtido é o Último calculado: neste caso o valor é 90 o qual foi calculado na 3ª iteração.

## CAPÍTULO III

## OBTENÇÃO DE LIMITES INFERIORES

## III.1 - INTRODUÇÃO

Vários métodos heurísticos poderão ser utilizados para encontrar soluções viáveis do problema (P), e portanto, limites inferiores para  $V(P)$ . Nas seções seguintes são apresentados três desses métodos.

Todas as heurísticas propostas podem terminar sem achar alguma solução viável. Por esta razão, no método exato que se descreverá no capítulo IV será usado um método que consiste na aplicação sucessiva dos primeiros dois algoritmos apresentados neste capítulo.

A seguinte notação será necessária: dado um conjunto  $F \subset I$ , define-se o seguinte problema de alocação generalizado:

$$\begin{aligned}
 (Q_F) \quad & \max \sum_{i \in F} \sum_{j \in J} p_{ij} x_{ij} \\
 & \text{s.a.} \quad \sum_{i \in F} x_{ij} = 1, \quad j \in J \\
 & \quad \sum_{j \in J} q_j x_{ij} \leq b_i, \quad i \in F \\
 & x_{ij} \in \{0, 1\}, \quad i \in F, \quad j \in J.
 \end{aligned}$$

Seja  $\bar{x}$  uma solução viável de  $(Q_F)$ . Então, de acordo com a notação introduzida no capítulo I

$$v(\bar{x}) = \sum_{i \in F} \sum_{j \in J} p_{ij} \bar{x}_{ij} - \sum_{i \in F} f_i$$

será o valor de solução aproximada de (P) associada com  $\bar{x}$ .

### III.2 - ALGORITMO HEURÍSTICO HRPLC1

Este algoritmo será aplicado sobre a versão (P') de (P). É conveniente lembrar que (P') pode ser visto como um problema de alocação generalizado (PAG) com  $n+1$  clientes e  $m$  armazéns e onde o  $(n+1)$ -ésimo cliente pode ser alocado a no máximo  $m-L$  armazéns. Assim sendo, uma boa alternativa para obter soluções viáveis é usar uma adaptação de alguma boa heurística para o PAG. Neste trabalho será considerado o algoritmo descrito por MARTELLO e TOTH (26), no qual são introduzidas as modificações necessárias que permitem sua aplicação ao nosso problema (P').

No passo geral, o algoritmo que será proposto nesta seção, considera o conjunto dos clientes ainda não alocados, ou seja o conjunto

$$J' = \{j | \sum_i x_{ij} = 0, j \in J \text{ ou } \sum_i x_{ij} < m-L, j = n+1\}.$$

(Aqui o  $(n+1)$ -ésimo cliente é considerado alocado quando  $\sum_i x_{ij} = m-L$  para  $j = n+1$ ). A seguir escolhe-se um  $j^* \in J'$  tal que

$$j^* = \arg \max \{ \max_i (p_{ij}) - \max_i (p_{ij}) \mid j = 1, \dots, n+1 \},$$

onde  $p_{i,n+1} = f_i$ . Logo efetua-se  $x_{i^*j^*} := 1$  (ou  $z_{i^*} := 1$  se  $j = n+1$ ), onde

$$i^* = \arg \max_i \{ p_{ij^*} \}.$$

O algoritmo termina com uma solução viável quando os  $n$  clientes  $j \in J$  foram alocados. Neste caso, nesta solução viável, não é necessário que o cliente  $n+1$  tenha sido alocado no sentido da definição de alocação dada acima. Como já foi dito é possível que o algoritmo termine sem encontrar uma solução viável.

Os detalhes do algoritmo, que será chamado **HRPLC1**, são estabelecidos na Figura 111.1.

No algoritmo **HRPLC1**,  $(\hat{x}_{ij}, \tilde{z}_i)$  denota uma solução viável do problema  $(P')$ . Aqui é conveniente lembrar que  $z_i=0$  indica no problema de localização associado, que o armazém  $i$  é usado para atender a demanda de ao menos um cliente. Caso contrário  $z_i = 1$ .

No pior caso este algoritmo fará  $m-L$  alocações do  $(n+1)$ -ésimo cliente. Por outro lado o valor mínimo que  $L$  pode assumir é 1. Ou seja, no pior caso, haverá  $m-1$  armazéns  $i$  tais que  $z_i = 0$  (ou  $y_i = 1$ ). A complexidade do algoritmo para alocar os outros  $n$  clientes aos  $m$  centros é  $O(mn \log m + n^2)$  (ver MARTELLO e TOTH (26)). Portanto a heurística **HRPLC1** poderá ser implementada em tempo proporcional a  $m(mn \log m + n^2)$ .



**Algoritmo HRPLC1****Começo**

1.  $H := J$  ;  $e := 0$  ;
2.  $J' := J \cup \{n+1\}$ ;
3. para todo  $i \in I$  fazer  $a_i := b_i$ ;
4. enquanto  $H \neq \emptyset$  fazer

**Começo**

5.  $g := -\infty$
6. para todo  $j \in J'$  fazer  $I_j := \{p_{ij} | q_j \leq a_i\}$
7. se  $I_j = \emptyset$ , parar. (Não foi possível achar uma solução viável)

**caso contrário****Começo**

8. Se  $|I_j| \leq 1$  e  $j = n + 1$  ir a 10  
caso contrário  $d := \max_{i \in I_j} \{p_{ij}\} - \max_i \{p_{ij}\}$ ;
9. se  $d > g$  então  
 $g := d$ ;  $j^* := j$ ;  $i^* := \arg \max_{i \in I_j} \{p_{ij}\}$   
fim
10.  $\hat{x}_{i^*j^*} := 1$  se  $j^* \in J$  ( $\hat{z}_{i^*} := 1$  se  $j^* = n+1$ );
11.  $H := J - \{j^*\}$  se  $j \in J$ ;
12.  $a_{i^*} := a_{i^*} - q_{j^*}$ ;
13. se  $j^* = n+1$  então  $e := e + 1$ ;
14. Se  $e = m - L$  então  $J' := J' - \{n+1\}$ ;
15. para todo  $i \in I - \{i^*\}$  fazer  $\hat{x}_{ij^*} := 0$   
fim
16. para todo  $i$  tal que  $\hat{z}_i = 0$  e  $\hat{x}_{ij} = 0, \forall j$  fazer  $\hat{z}_i := 1$ .  
fim

Fig. III.1 - Algoritmo HRPLC1

### III.3 - ALGORITMO HEURÍSTICO HRPLC2

No método descrito a seguir, a idéia é tentar achar uma solução viável do problema supondo que será usado um conjunto  $H$  de  $L$  armazéns. Se isto não for possível o algoritmo considerará iterativamente a inclusão em  $H$  de algum armazém ainda não considerado, até que uma solução viável é encontrada ou até que  $H = I$ .

Uma descrição formal do algoritmo é dada na Figura III.2.

#### Algoritmo HRPLC2

##### Começo

1. Escolher um conjunto  $H$  de  $L$  armazéns;
2.  $G := I - H$ ;
3. Aplicar uma heurística ao problema  $Q_H$ ;  
(se uma solução viável foi achada, parar)
4. Se  $G = \phi$  então parar (não foi possível achar uma solução viável)

##### Caso contrário

##### Começo

5. escolher  $k \in G$  ;
6.  $G := G - \{k\}$ ;
7.  $H := H \cup \{k\}$ ;
8. **ir a 3**

fim

fim

Fig. III.2 - Algoritmo HRPLC2

A complexidade desta heurística está dada pelo passo 3, o qual pode ser implementado em tempo  $O(mn \log m + n^2)$  (ver (26)). Este passo é executado (no pior caso)  $m-1$  vezes. Portanto a complexidade do algoritmo será  $O(m(mn \log m + n^2))$ .

Como o sucesso do passo 3 não está garantido, esta heurística poderá terminar sem achar uma solução viável. Para melhorar a probabilidade do sucesso podem-se definir critérios para a escolha inicial do conjunto  $H$  (passo 1) e para a escolha dos elementos que poderão ser incluídos em  $H$ . Nas implementações feitas neste trabalho, bons resultados foram obtidos usando o critério seguinte:

- (i) Colocar os armazéns  $i \in I$  na ordem não crescente dos valores de alguma função  $h_i$ , e
- (ii) incluir em  $H$  os primeiros  $L$  armazéns, e posteriormente, na medida que for necessário, ir incluindo em  $H$  os armazéns "melhor colocados".

A função  $h_i$  usada aqui foi  $h_i = b_i/f_i$ . Outras funções podem ser  $f_i$ ,  $b_i$ , etc. Em qualquer caso, a complexidade do algoritmo continuará sendo a mesma.

Uma maneira de garantir o sucesso na busca de uma solução viável seria utilizar no passo 3 um método exato em lugar de um método aproximado. Neste caso, devido a complexidade do problema a ser resolvido no passo 3, este algoritmo poderá chegar a ter um custo computacional muito alto.

### III.4 - ALGORITMO HEURÍSTICO HRPLC3

Este método se inicia com uma solução viável na qual todos os armazéns são utilizados (ou seja  $y_i = 1$ , para todo  $i \in I$ ). Se esta solução não estiver disponível, então o método não pode ser aplicado. No passo geral, dado um conjunto  $M = \{i \in I | y_i = 1\}$ , escolhe-se, para excluir de  $M$ , um armazém  $k$  tal que a solução viável do problema de alocação  $Q_{M-\{k\}}$  é melhor que a do problema  $Q_M$ . O algoritmo termina quando não existe nenhum  $k \in M$  tal que sua exclusão melhore a atual solução viável, ou então quando  $|M| = L$ .

Os passos deste algoritmo são descritos na Figura III.3.

Para calcular a complexidade deste método, pode-se considerar o número  $N(Q)$  de vezes que o passo 2 é executado. O pior caso consiste na exclusão de um elemento de  $M$  depois de testar todos os elementos de  $G$ , até que  $|M| = L$ . Isto implica que

$$N(Q) = m + (m-1) + (m-2) + \dots + (L+2) + (L+1)$$

$$= \frac{m(m-1)}{2} - \frac{L(L+1)}{2}$$

Portanto, é possível concluir que a implementação deste algoritmo pode ser feita em tempo proporcional  $\varepsilon$

$$m^2 (mn \log m + n^2)$$

**Algoritmo HRPLC3****Começo**

1.  $M := I; G := I;$
2. Achar uma solução viável  $\bar{x}$  de  $Q_M$ ; (se não for possível, parar)
3.  $\hat{x} := \bar{x};$
4. enquanto  $|M| > L$ , efetuar

**Começo**

5. Se  $G = \phi$ , parar: (a solução viável é  $\hat{x}$ ),  
caso contrário escolher  $k \in M$  e fazer  $G := G - \{k\};$
  6. achar uma solução viável  $\bar{x}$  de  $Q_{M - \{k\}}$ . Se não for possível, ir a 5.
  7. Se  $v(\bar{x}) < v(\hat{x})$  ir a 5,  
caso contrário,  $\hat{x} := \bar{x}; M := M - \{k\}; G := M;$
- fim

fim

Fig. III.3 - Algoritmo HRPLC3

## CAPÍTULO IV

## UM MÉTODO EXATO

## IV.1 - INTRODUÇÃO

Neste capítulo descreve-se um algoritmo enumerativo do tipo "branch and bound" para encontrar uma solução Ótima do problema (P) formulado no Capítulo I. A seguir se apresenta a terminologia básica para a descrição do método.

De um modo geral, um método de "branch and bound" consiste no particionamento do conjunto de soluções viáveis do problema em subconjuntos sucessivamente menores, os quais permitem obter cotas (limites) para o valor de função objetivo. Estes limites são calculados a partir de subproblemas (que são versões relaxadas do problema original) associados com cada um dos subconjuntos da partição. Alguns dos subconjuntos gerados poderão ser descartados em função dos limites calculados. O método termina quando cada subconjunto produz uma solução viável ou quando se pode mostrar que ele não pode fornecer uma solução melhor que a melhor solução viável disponível.

Seja (Q) um problema da forma

$$\begin{aligned} (Q) \max f(x) \\ \text{s.a } x \in S. \end{aligned}$$

Seja ( $Q_i$ ) o subproblema associado ao subconjunto  $i$  da partição. Os elementos básicos que compõem um esquema de "branch and bound" são os seguintes:

(i) uma regra de separação (ou particionamento, ou "branching") que permitirá construir uma lista de subproblemas (problemas candidatos). Esta regra é usada para particionar  $SV(Q_i)$  em subconjuntos  $S_{i1}, S_{i2}, \dots, S_{iq}$  tais que

$$\bigcup_{j=1}^q S_{ij} = SV(Q_i).$$

(ii) uma relaxação de  $Q$ , ou seja, um problema do tipo

$$\begin{aligned} \text{(QR)} \quad & \max g(x) \\ & \text{s.a. } x \in T \supseteq S, \end{aligned}$$

tal que para todo  $x, y \in S$  tem-se que

$$f(x) > f(y) \Rightarrow g(x) > g(y).$$

(iii) Um método de cálculo de  $V(\text{QR})$ . Este valor é um limite superior para  $V(Q)$ .

(iv) uma regra para selecionar da lista de problemas candidatos, o seguinte problema a ser processado.

(v) (opcional) uma heurística para encontrar soluções viáveis de  $(Q)$  e, em consequência, um limite inferior para  $V(Q)$ .

(vi) (opcional) métodos que permitam uma redução fixando os valores de algumas variáveis ou descartando alguns dos problemas da lista.

Obviamente, a qualidade dos elementos enumerados acima, terá uma influência muito grande na velocidade de convergência do método. Por exemplo, uma pequena diferença entre o limite inferior e superior fará que o número de soluções do problema que deverão ser enumerados explicitamente seja pequeno; etc.

## IV.2 - O ESQUEMA "BRANCH AND BOUND"

A seguir descrevem-se os passos do método proposto para resolver o problema (P). Nesta descrição o símbolo SV será usado em lugar da expressão "solução viável".

**Passo 1:** (Inicialização) Colocar (P) na lista de subproblemas. Opcionalmente, usar alguma heurística para obter um limite inferior LI. Senão fazer  $LI := -\infty$ .

**Passo 2:** (Seleção de um subproblema da lista). Se a lista está vazia, parar: a solução associada com LI é Ótima (ou, o problema (P) não tem solução se  $LI = -\infty$ ). Caso contrário escolher um subproblema ( $P_k$ ) e removê-lo da lista.

**Passo 3:** (Limite superior). Resolver a relaxação ( $LR_k$ ) de ( $P_k$ ) para achar um limite superior  $LS_k$ .

Se  $LS_k \leq LI$ , voltar ao passo 2.

Se  $LS_k > LI$ : se a solução de ( $LR_k$ ) não for viável, ir ao passo 4; caso contrário, se o valor da solução de ( $LR_k$ ) for maior que LI, então atualizar a SV e o limite LI. Ir ao passo 5.

**Passo 4:** (Limite inferior: opcional). Usar a solução de ( $LR_k$ ) para achar uma SV, utilizando alguma heurística. Se o valor da solução é maior que LI, então, atualizar a SV e LI.



Se  $LS_k > LI$ , ir ao passo 5. Senão, voltar ao passo 2.

Passo 5: Se alguma variável  $y_i$  estiver livre (não fixada em 0 ou 1) ir ao passo 6. Senão, ou seja se  $y_i = 0$  ou 1, para todo  $i \in I$ , utilizar um algoritmo exato para resolver o problema de alocação generalizado definido pelo conjunto de armazéns  $\{i | y_i = 1\}$ ; atualizar a SV e LI se a solução encontrada for melhor que a atual; ir ao passo 2.

Passo 6: (Redução: opcional) Fixar os valores de algumas variáveis  $y_i$  em 0 ou 1, utilizando algum método de redução.

Passo 7: (Separação ou "branching") Escolher uma variável  $y_i$  e gerar dois novos subproblemas:  $(P_{k0})$  fazendo  $y_i = 0$  em  $(P_k)$ , e  $(P_{k1})$  fazendo  $y_i = 1$  em  $(P_k)$ . Colocá-los na lista nessa ordem e ir ao passo 2.

O procedimento descrito acima pode ser representado por uma árvore binária (chamada árvore de busca ou de "branch and bound") com raiz, o vértice correspondente ao problema original, e onde os sucessores do vértice  $k$  estão associados com os subproblemas  $(P_{k0})$  e  $(P_{k1})$  gerados a partir de  $(P_k)$  pela regra de "branching" do passo 7.

No passo 1 podem ser utilizadas as heurísticas descritas no Capítulo III, para obter soluções viáveis e limites inferiores. Na implementação do método foi aplicado um algoritmo em duas fases: a primeira consiste da aplicação da heurística HRPLC1 e, no caso de não conseguir uma SV, utiliza-se, na segunda fase, a heurística HRPLC2.

O critério de escolha do próximo problema da lista (passo 2) é "último em entrar, primeiro a sair" (UEPS ou "LIFO"). Portanto a lista de subproblemas é, na realidade uma "pilha" de problemas e, toda vez que deve ser feita uma seleção, o próximo problema a ser escolhido é aquele que está no topo da pilha. Este critério define um processo de busca em profundidade na árvore de "branch and bound". Neste caso a idéia é encontrar o mais rapidamente possível uma SV do problema.

No cálculo de um limite superior no passo 3 foi adotado o seguinte procedimento: se o problema a ser resolvido for  $(LR_1)$  (ou seja, no vértice inicial) então é utilizado o MAM descrito no Capítulo II. Nos vértices intermediários, os limites superiores foram obtidos calculando simplesmente os efeitos de fazer  $y_i = 0$  ou  $y_i = 1$  em  $(P_k)$ . Não foram realizadas iterações adicionais para melhorar os limites.

No passo 4 (que é opcional) o método heurístico é aplicado sobre o PAG definido pelo conjunto  $\{i | y_i = 1 \text{ no problema lagrangeano do passo 3}\}$ . O método escolhido é o descrito por MARTELLO e TOTH (26).

Os possíveis métodos de redução a serem utilizados no passo 6, serão descritos na seção seguinte.

Para resolver o PAG do passo 5, foi usado o método que MARTELLO e TOTH apresentam em (26) e (27). Outros métodos podem ser encontrados em ROSS e SOLAND (35) e FISHER et alii (14).

Finalmente, a escolha da próxima variável para separação foi feita da seguinte forma: seja  $\{\bar{\lambda}_j | j \in J\}$  o conjunto de multiplicadores correspondentes ao limite superior derivado de

(LR<sub>1</sub>). Considere-se para todo  $i \in I$  o seguinte problema de mochila em variáveis 0-1:

$$\begin{aligned}
 (K^i) \quad & \max \sum_j (p_{ij} - \hat{\lambda}_j) \\
 \text{s.a.} \quad & \sum_j q_j x_{ij} \leq b_i \\
 & x_{ij} \in \{0, 1\}, \quad j \in J,
 \end{aligned}$$

e definamos  $\alpha_i = V(K^i) - f_i$ .

A próxima variável a ser separada é  $y_{i_0}$ , onde  $i_0$  é tal que

$$\alpha_{i_0} = \max\{\alpha_i \mid i \in I\}$$

Cada vértice da árvore de "branch and bound" que representa o algoritmo descrito, está associado a um subproblema. Este subproblema fica definido por um vetor que contém a informação sobre o estado das variáveis  $y_i$  (que pode ser 0, 1 ou livres) e os valores de L e U até esse momento.

Um vértice terminal da árvore de busca é aquele vértice k onde não existem variáveis  $y_i$  livres, ou seja onde  $y_i = 0$  ou 1 para todo  $i \in I$ . Se num vértice terminal k,  $LS_k > LI$  então deve-se resolver um PAG e, se for o caso, atualizar a SV e o limite inferior LI.

Se em algum vértice k acontecer que  $LS_k \leq LI$  então o próximo problema da lista deve ser escolhido. Esta operação implica uma "volta atrás" (ou "backtracking") na árvore de "branch and bound", até o vértice associado ao próximo problema na lista.

### IV.3 - MÉTODOS DE REDUÇÃO

Algumas reduções no tamanho do problema podem ser conseguidas utilizando o limite inferior LI e algumas penalidades que podem ser calculadas a partir do problema lagrangeano. Para um dado vértice se o limite superior obtido usando essas penalidades, é menor que LI, então a condição penalizada não pode acontecer e, portanto, pode ser descartada de futuras considerações em qualquer vértice da subárvore com raiz o vértice atual.

Vários métodos de cálculo de penalidades podem ser encontrados na literatura e que poderiam ser aplicados aqui. Ver por exemplo (5) e (8). Nesta seção se descrevem três desses métodos.

Sejam  $k$  o vértice atual na árvore de "branch and bound" e  $(P_k)$  o correspondente problema candidato. Dada uma solução  $(x^*, y^*)$  do problema dual lagrangeano de  $(P_k)$  define-se

$$M = \{i | y_i^* = 1\}.$$

Definem-se também os seguintes conjuntos:

$$K_0 = \{i | y_i = 0\},$$

$$K_1 = \{i | y_i = 1\}$$

$$K_2 = \{i | y_i \text{ é livre}\}$$

Notar que  $K_1 \subseteq M$  e que os elementos de  $K_0$  não podem fazer parte da solução do dual lagrangeano.

Os métodos de redução que serão considerados neste trabalho são os seguintes:

(i) Método para fixar  $y_i = 0$  onde  $i \notin M$  (ou  $i \in I - M - K_0$ )

Este método está baseado na penalidade de incluir  $i$  em  $M$ , ou seja, de incluir o armazém  $i$  na solução do problema dual lagrangeano. Esta penalidade está dada por

$$p_1 = \begin{cases} -\alpha_i & \text{se } |M| \neq L \text{ e } |M| \neq U \\ \min_{j \in M - K_1} \{\alpha_j\} - \alpha_i; & \text{se } |M| = U, \text{ e} \\ \min(0, \min_{j \in M - K_1} \{\alpha_j\} - \alpha_i), & \text{se } |M| \neq U \text{ e } |M| = L. \end{cases}$$

Neste caso se  $LS_k - p_1 < LI$ , então  $y_i$  deve ser fixada em 0.

(ii) Método para fixar  $y_i = 1$  onde  $i \in M - K_1$

Aqui é considerada a penalidade de excluir de  $M$  o armazém  $i$  tal que  $y_i$  permanece livre até o vértice  $k$ . O valor desta penalidade é

$$p_0 = \begin{cases} \alpha_i - \max(0, \max_{j \in I - M - K_0} \{\alpha_j\}) & \text{se } |M| \neq L, \text{ e} \\ \alpha_i - \max_{j \in I - M - K_0} \{\alpha_j\} & \text{se } |M| = L. \end{cases}$$

Se  $LS_k - p_0 < LI$ , então a variável  $y_i$ , deve ser fixada em 1.

(iii) Método para diminuir a diferença  $U - L$

Uma pequena diferença entre o número máximo e mínimo de armazéns, poderá melhorar o limite superior e o poder de "poda" das penalidades calculadas nos dois métodos anteriores. Se no vértice  $k$  se estabelece a condição de que na solução Ótima haverá exatamente  $q$  armazéns  $i$  tais que  $y_i = 1$ , então o valor do limite superior será

$$LS'_k = \sum_j \lambda_j + \sum_{i \in K_1} \alpha_i + \sum_{j=1}^{q-|K_1|} \max_j \{\alpha_i \mid i \in K_2\}$$

onde, de acordo com a notação introduzida no capítulo II,  $\max_j \{\alpha_i \mid i \in K_2\}$  é o  $j$ -ésimo maior  $\alpha_i$  tal que  $y_i$  ainda é uma variável livre.

Se  $LS'_k < LI$  então pode-se fazer  $L := L + 1$  se  $q = L$ , ou  $U := U - 1$  se  $q = U$ .

É importante ressaltar que todos estes métodos preservam a condição de que na solução Ótima o número de armazéns  $i$  tais que  $y_i = 1$  estará no intervalo  $[L, U]$ .

#### IV.4 - UM EXEMPLO

Com o objetivo de ilustrar o algoritmo de "branch and bound" descrito neste capítulo, será considerado um exemplo com os seguintes dados:

$$I = \{1, 2, 3, 4, 5\},$$

$$J = \{1, 2, 3, 4, 5, 6\},$$

$$(p_{ij}) = \begin{bmatrix} 780 & 630 & 666 & 392 & 218 & 570 \\ 600 & 798 & 432 & 468 & 314 & 270 \\ 675 & 570 & 738 & 582 & 410 & 410 \\ 270 & 450 & 396 & 677 & 778 & 10 \\ 600 & 522 & 468 & 525 & 250 & 750 \end{bmatrix}, \quad (f_i)^T = \begin{bmatrix} 300 \\ 500 \\ 600 \\ 800 \\ 1100 \end{bmatrix},$$

$$(b_i)^T = (20 \quad 30 \quad 38 \quad 42 \quad 55),$$

$$(q_j) = (15 \quad 12 \quad 18 \quad 19 \quad 16 \quad 20).$$

No vértice inicial da árvore (vértice 1) o limite inferior (LI) e o limite superior (LS) são 1961 e 2393, respectivamente. Para o cálculo do LI foram utilizados os algoritmos HRPLC1 e HRPLC2 de maneira sugerida na seção 111.4. Por outro lado, o LS foi calculado utilizando o método de ajuste de multiplicadores (MAM) proposto no Capítulo II. Neste caso, o MAM forneceu como subproduto uma nova solução viável de valor 2036. Este valor permite a atualização da solução viável e do LI.

Os multiplicadores associados com LS foram

$$(\lambda_j) = (600 \quad 630 \quad 486 \quad 582 \quad 410 \quad 410)$$

Com estes valores é possível calcular  $a_i$  para todo  $i \in I$ , resolvendo os problemas  $(K_i)$  da seção IV.2, resultando

$$(a_i) = (-120 \quad -332 \quad -273 \quad -339 \quad -760)$$

Como não se satisfaz o critério de otimalidade deveria ser escolhida uma variável para separar. Antes de efetuar a separação, são aplicados os métodos de redução (i) e (ii) da seção anterior. Estes métodos permitiram fixar em 0 a variável  $y_5$ .

Para determinar a variável a ser separada obtém-se  $i_0$  tal que

$$a_{i_0} = \max_{i \in K_2} \{\alpha_i\},$$

e é escolhida para separar a variável  $y_{i_0}$ . Neste caso  $i_0 = 1$ .

Dois novos vértices são criados: um associado a  $y_1 = 1$  e o outro a  $y_1 = 0$ . O seguinte vértice a ser considerado é o associado com  $y_1 = 1$ . Neste vértice  $(LI, LS) = (2036, 2393)$ , portanto uma nova variável é escolhida para separar, criando mais dois vértices.

Este processo continua até que no vértice 5 os limites são  $(LI, LS) = (2036, 2052)$ . Neste ponto é resolvido um PAG cuja solução tem valor 2036. Aqui deve ser feita uma "volta atrás" ou "backtracking" até o vértice 6, onde o problema candidato associado é não viável.

O procedimento termina depois que todos os vértices criados são considerados. Isto acontece no vértice 9. A árvore completa é mostrada na Figura IV.1.

A solução Ótima do problema corresponde à Última solução viável encontrada. Os valores Ótimos são

$$V(\text{problema}) = 2101,$$

$$(y_i^*) = (0 \ 1 \ 1 \ 1 \ 0),$$

$$(x_j^*) = (2 \ 2 \ 3 \ 4 \ 4 \ 3),$$

onde,  $x_j = i_0$  implica que, para o cliente  $j$ ,



$$x_{i_0 j}^* = \begin{cases} 1 & \text{se } i = i_0 \\ 0 & \text{para todo } i \neq i_0 . \end{cases}$$

Neste exemplo, os métodos de redução (i) e (ii) foram aplicados em cada um dos vértices da árvore. Uma comparação com a árvore resultante de aplicar o algoritmo sem incluir os méto-

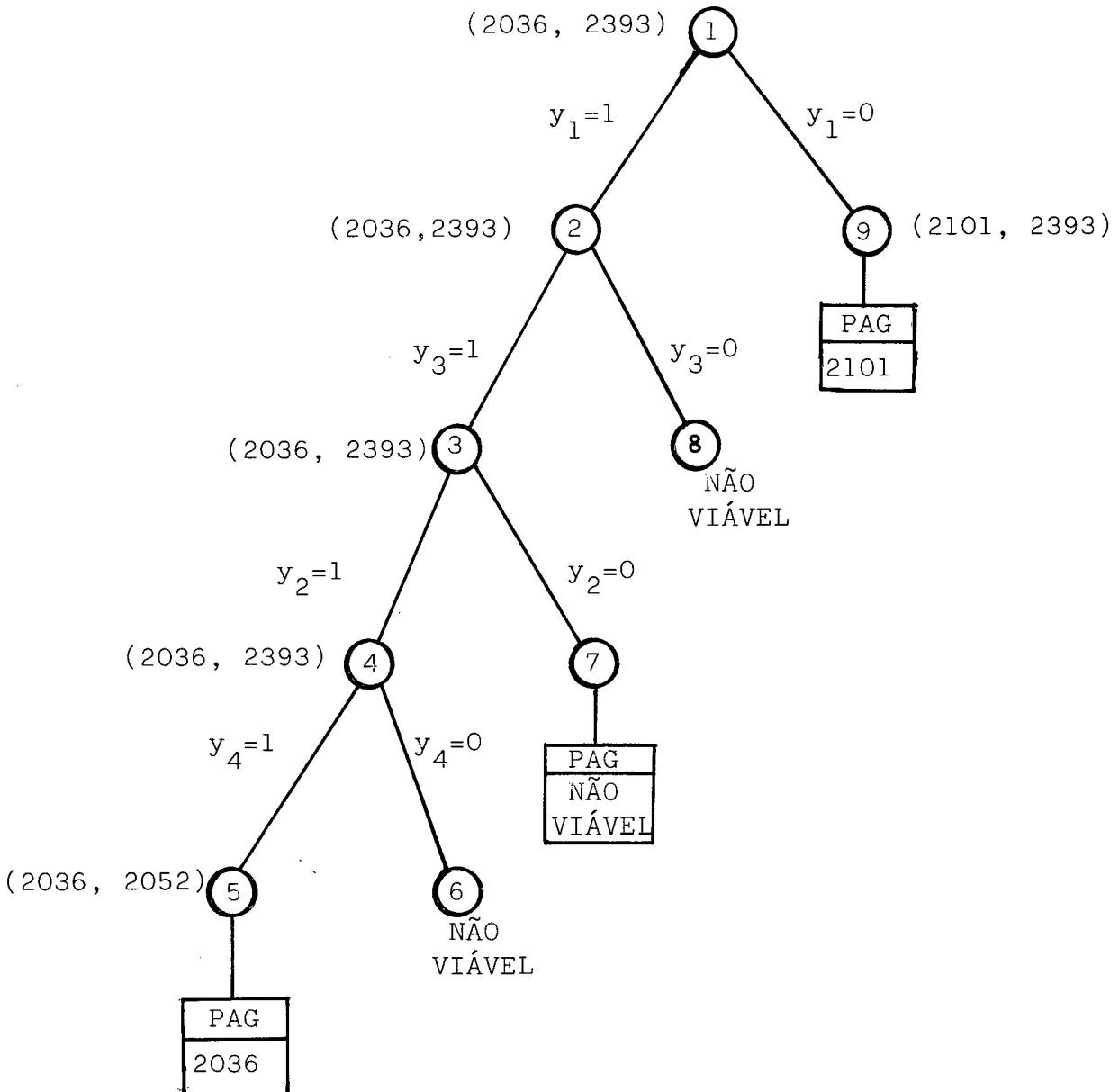


Fig. IV.1 - Árvore de "branch and bound" para o exemplo

dos de redução permite ter uma idéia das vantagens de sua utilização. Os dados relevantes são mostrados na Tabela IV.1.

	SEM REDUÇÃO	COM REDUÇÃO
Vértices gerados	31	9
Vértices terminais	7	3
Heurísticas aplicadas	13	2
SAP's resolvidos	3	3

TABELA IV.1

Na árvore mostrada na Figura IV.1 os números nos vértices indicam a ordem em que eles foram processados; os pares (—, —) representam (LI, LS) no vértice.

#### IV.5 - RESULTADOS COMPUTACIONAIS

O algoritmo de "branch and bound" proposto neste capítulo, foi codificado em FORTRAN IV e rodado num computador UNISYS A10 do Núcleo de computação Eletrônica da Universidade Federal do Rio de Janeiro. Foram testados 60 problemas gerados aleatoriamente. Os lucros  $c_{ij}$  e demandas  $q_j$  são números inteiros selecionados de uma distribuição uniforme entre 1 e 40, e entre 5 e 25, respectivamente.

Os 60 problemas da amostra foram agrupados de acordo com certos valores de  $b_i$ , da seguinte maneira:

$$\text{Grupo 1: } b_i = 1,8 \sum_j q_j / m,$$

$$\text{Grupo 2: } b_i = 1,5 \sum_j q_j/m,$$

$$\text{Grupo 3: } b_i = 1,2 \sum_j q_j/m.$$

Os custos fixos considerados foram  $f_i = 10 \text{ n/m}$ . Os resultados são mostrados na Tabela IV.2. Dentro de cada grupo testaram-se cinco problemas para cada par de valores de  $m$  e  $n$ . Para cada um desses conjuntos de cinco problemas se mostra o tempo médio e máximo de execução, em segundos, excluindo os tempos de entrada e saída.

A coluna (1) da tabela contém dois dados: o primeiro é o número de problemas que necessitaram ramificação e, o segundo (entre parênteses) a diferença percentual média do limite superior respeito do valor ótimo do problema.

Os dados na coluna (2) correspondem ao número de problemas que se reduziram a um problema de alocação generalizado (PAG). É claro que este número aumenta na medida que a diferença entre  $\sum_i b_i$  e  $\sum_j q_j$  diminui.

A informação da Tabela IV.2 mostra que, apesar do pequeno porte dos problemas, o algoritmo proposto é viável.

Alguns testes adicionais foram feitos sobre problemas de maior tamanho. Os tempos de execução nestes problemas mostraram uma forte dependência dos tempos de execução na solução dos PAG. Em todos os problemas testados, a solução dos PAG consumiu entre 94 e 99% do tempo total de execução.

GRUPO	m	n	TEMPO DE EXECUÇÃO (SEG.)		(1)	(2)
			MÉDIO	MÁXIMO		
1	3	10	0,077	0,117	1 (4,33)	0
	3	20	0,060	0,067	-	0
	5	10	0,553	1,083	5 (2,99)	0
	5	20	0,853	1,283	4 (5,37)	0
2	3	10	0,010	0,017	-	5
	3	20	3,520	8,533	3 (2,72)	2
	5	10	0,250	0,433	2 (3,05)	0
	5	20	0,130	0,283	1 (0,42)	0
3	3	10	0,027	0,067	-	5
	3	20	0,030	0,100	-	5
	5	10	1,633	7,167	-	5
	5	20	2,037	4,833	-	5

TABELA IV.2

Existem pelo menos duas maneiras de melhorar os resultados obtidos: a primeira é o uso de programação dinâmica para resolver os problemas da mochila 0-1 no algoritmo de cálculo do limite superior ou, a utilização de um código mais eficiente, como por exemplo os de FAYARD e PLATTEAU (11) ou MARTELLO e TOTH (28); a segunda possibilidade de melhora, é a utilização do algoritmo de FISHER et alii (14) para resolver os PAG's. Este algoritmo precisa aproximadamente 1/10 do tempo do algoritmo de MARTELLO e TOTH (26), naqueles PAG's considerados mais difíceis.

## CAPÍTULO V

MÉTODOS DE DECOMPOSIÇÃO LAGRANGEANA PARA  
OBTER LIMITES SUPERIORES MAIS FORTES

## V.1 - INTRODUÇÃO

Neste capítulo serão discutidas relaxações lagrangeanas não convencionais, visando a obtenção de limites superiores que podem ser consideravelmente mais fortes que nos métodos clássicos.

Estes métodos, que serão chamados de decomposição lagrangeana, consistem na formulação de um problema (P1) equivalente ao problema original (P), criando uma (ou várias) cópia(s) dos vetores das variáveis de decisão, utilizando cada uma delas em cada conjunto de restrições de (P). Assim sendo, é possível definir uma relaxação lagrangeana de (P1) que decompõe aditivamente em vários subproblemas, onde cada um dos quais conserva um dos subconjuntos de restrições de (P).

Este tipo de método vem sendo pesquisado e utilizado por vários autores desde 1975. No entanto, foi nos trabalhos de GUIGNARD e KIM (18) e (19), onde as implicações e propriedades deste novo uso da relaxação lagrangeana são estudados mais sistematicamente.

Nas seções V.2 e V.3 deste capítulo, se faz uma descrição geral dos métodos de decomposição de lagrangeana. Logo, na seção V.4 se apresenta um exemplo para ilustrar a potencial melhoria nos limites. Finalmente, na seção V.5 se discute a aplicação destes novos métodos ao problema de localização capacitado.

## V.2 - MÉTODO 1 (GUIGNARD e KIM (18) e (19))

Considere-se o seguinte problema de programação linear inteira:

$$(P) \quad \max f x$$

$$\text{s.a} \quad Ax \leq b \quad (V.1)$$

$$Cx \leq d \quad (V.2)$$

$$x \in X \quad (V.3)$$

onde  $f^T \in \mathbb{R}^n$ ,  $x \in \mathbb{R}^n$ ,  $b \in \mathbb{R}^p$  e  $d \in \mathbb{R}^{m-p}$ ; A e C são matrizes de dimensões apropriadas. Supõe-se que as restrições (V.1), (V.2) e (V.3) definem um conjunto não vazio, ou seja, supõe-se que o problema (P) é viável.

Depois de criar uma cópia de vetor x das variáveis de decisão, pode-se formular o seguinte problema equivalente a (P):

$$(P1) \quad \max f x$$

$$\text{s.a} \quad Ay \leq b \quad (V.4)$$

$$y = x \quad (V.5)$$

$$Cx \leq d \quad (V.6)$$

$$x \in X, y \in Y \supseteq X \quad (V.7)$$

Duas relaxações lagrangeanas de (P) podem ser definidas: a primeira consiste na dualização das restrições (V.1) em (P) utilizando o vetor de multiplicadores duais  $v \geq 0$ ; na segunda relaxação é dualizado o conjunto de restrições (V.5) em (P1) associando-lhe o vetor de multiplicadores  $w^T \in \mathbb{R}^n$ . Os problemas lagrangeanos resultantes são, respectivamente,

$$(PR_v) \quad \max (f - vA)x + vb,$$

$$\text{s.a.} \quad Cx \leq d$$

$$x \in X,$$

e

$$(PR_w) \quad \max fx + w(y - x)$$

$$\text{s.a.} \quad (V.4), (V.6) \text{ e } (V.7).$$

ou, equivalentemente,

$$(PR_w) \quad \max (f-w) x + \max wy$$

$$\text{s.a.} \quad Cx \leq d \quad \text{s.a.} \quad Ay \leq b$$

$$x \in X \quad y \in Y.$$

Associado com  $(PR_v)$  e  $(PR_w)$ , respectivamente, definem-se os seguintes problemas duais lagrangeanos:

$$(R) \quad \min_{v \geq 0} V(PR_v) \quad , \quad e$$

$$(D1) \quad \min_w V(PR_w).$$

O seguinte teorema garante que o limite fornecido por (D1) é, no pior caso, igual ao fornecido por (R).

**Teorema V.1** (GUIGNARD e KIM (18) e (19))

Considere-se  $v^0 \in SO(R)$ ,  $\bar{w} = v^0 A$ , e seja  $(x^0, y^0) \in SO(PR_w^-)$ . Então

$$(i) \quad V(PR_w^-) - V(PR_{v^0}) = v^0(Ay^0 - b), \text{ e}$$

$$(ii) \quad V(D_1) \leq V(R)$$

**Demonstração:** Ver (18) e (19).

**Teorema V.2** (GUIGNARD e KIM (18) e (19))

Sejam (R1) e (R2) os problemas duais associados com as duas relaxações lagrangeanas clássicas obtidas relaxando (V.1) e (V.2), respectivamente. Se algum dos subproblemas de  $(PR_w^-)$  satisfaz a propriedade da integralidade então

$$V(D1) = \min\{V(R1), V(R2)\}$$

**Demonstração:** Ver (18) e (19)

Uma observação importante é que a diferença entre  $V(D1)$  e  $V(R)$  pode vir de duas fontes:

(i)  $V(PR_w^-) \leq V(PR_v)$  para  $w = vA$ . Se não se satisfaz a complementariedade das folgas haverá uma diferença entre  $V(PR_w^-)$  e



$$V(PR_v).$$

(ii) Pode haver outros valores de  $w$  que não sejam da forma  $w = vA$ ,  $v \geq 0$ , e que fornecem melhores limites. Isto porque

$$V(D1) = \min_w V(PR_w) \leq V(PR_w), \quad \forall w.$$

### V.3 - MÉTODO 2

Nesta seção se discute um novo esquema de decomposição lagrangeana aplicado a problemas de programação linear inteira. Uma das características deste esquema é que ele pode fornecer limites superiores consideravelmente melhores que os fornecidos pela relaxação lagrangeana convencional, com um custo computacional que pode ser menor que no método de decomposição descrito na seção anterior. Isto porque, como se verá mais adiante, o número de restrições que devem ser relaxadas para conseguir a separação do problema lagrangeano, poderá ser muito menor.

#### V.3.1 - Alguns Resultados Importantes

Consideremos o problema (P) definido na seção V.2. As restrições (V.1) podem ser reescritas da seguinte maneira:

$$A_1 x^1 + A_2 x^2 \leq b,$$

onde  $x^1 \in \mathbb{R}^k$ ,  $x^2 \in \mathbb{R}^{n-k}$  e onde  $A_1$  e  $A_2$  são submatrizes de  $A$  com dimensões apropriadas.

Seja  $y = \begin{bmatrix} y^1 \\ y^2 \end{bmatrix}$  uma cópia do vetor das variáveis de deci

são. Aqui as dimensões de  $y^1$  e  $y^2$  são as mesmas de  $x^1$  e  $x^2$ , respectivamente. Isto permite formular o seguinte problema equivalente a (P):

$$(P2) \quad \max fx$$

$$\text{s.a} \quad Ay = A_1 y^1 + A_2 y^2 \leq b \quad (V.8)$$

$$y^1 = x^1 \quad (V.9)$$

$$A_2 y^2 = A_2 x^2 \quad (V.10)$$

$$Cx \leq d \quad (V.11)$$

$$y \in Y, x \in X \subseteq Y \quad (V.12)$$

Pode-se definir uma relaxação lagrangeana de (P2), dualizando as restrições (V.9) e (V.10). Sejam  $(u^1)^T \in R^k$  e  $(u^2)^T \in R^P$  os vetores duais. O problema lagrangeano resultante

$$(PR_{u^1, u^2}) \quad \max fx + u^1(y^1 - x^1) + u^2(A_2 y^2 - A_2 x^2)$$

$$\text{s.a} \quad (V.8), (V.11) \text{ e } (V.12),$$

ou, equivalentemente,

$$(PR_{u^1, u^2}) \quad \max fx - u^1 x^1 - u^2 A_2 x^2 + \max u^1 y^1 + u^2 A_2 y^2$$

$$\text{s.a} \quad Cx \leq d \quad \text{s.a} \quad A_1 y^1 + A_2 y^2 \leq b$$

$$x \in X, \quad y \in Y.$$

É importante notar que no caso que  $k = n$ , ou seja, quando a matriz  $A$  e os vetores das variáveis de decisão não são particionados, as restrições (V.9) e (V.10) ficam reduzidas a restrição (V.5) do problema (P1).

O problema dual lagrangeano de (P2) é

$$(D2) \quad \text{Min}_{u^1, u^2} V(\text{PR}_{u^1, u^2}) .$$

A seguir se estabelecem alguns resultados que mostram a relação existente entre os limites fornecidos por (D2) e aqueles fornecidos por (D1) e (R).

#### Lema V.1

Seja  $v^0 \in SO(R)$  e considere-se  $\bar{u}^1 = v^0 A$ , e  $\bar{u}^2 = v^0$ . Seja  $(x^0, y^0) \in SO(\text{PR}_{\bar{u}^1, \bar{u}^2})$ . Então,

$$(i) \quad V(\text{PR}_{\bar{u}^1, \bar{u}^2}) - V(\text{PR}_{v^0}) = v^0(Ay^0 - b), \text{ e}$$

$$(ii) \quad V(D2) \leq V(R)$$

#### Demonstração:

$$V(\text{PR}_{\bar{u}^1, \bar{u}^2}) = \max fx - \bar{u}^1 x^1 - \bar{u}^2 A_2 x^2 + \max \bar{u}^1 y^1 + \bar{u}^2 A_2 y^2$$

$$\text{s.a} \quad Cx \leq d \qquad \text{s.a} \quad A_1 y^1 + A_2 y^2 \leq b$$

$$x \in X ,$$

$$y \in Y.$$

Seja  $(x^0, y^0) = \begin{bmatrix} x_0^1 & y_0^1 \\ x_0^2 & y_0^2 \end{bmatrix}$ . Como  $(x_0, y_0)$  é solução

ótima de  $(PR_{\bar{u}^1, \bar{u}^2})$  têm-se que

$$\begin{aligned} V(PR_{\bar{u}^1, \bar{u}^2}) &= fx^0 - \bar{u}^1 x_0^1 - \bar{u}^2 A_2 x_0^2 + \bar{u}^1 y_0^1 + \bar{u}^2 A_2 y_0^2 \\ &= fx^0 - v^0 A_1 x_0^1 - v^0 A_2 x_0^2 + v^0 A_1 y_0^1 + v^0 A_2 y_0^2 \\ &= (f - v^0 A) x^0 + v^0 A y^0 \end{aligned}$$

Somando e restando  $v^0 b$ , e como  $x^0 \in SO(PR_{v^0})$  obtém-se que

$$V(PR_{\bar{u}^1, \bar{u}^2}) = V(PR_{v^0}) + v^0 (Ay^0 - b)$$

Fica, portanto, demonstrada a parte (i).

Para provar (ii) é suficiente observar que  $v^0 \geq 0$ ,  $Ay^0 - b \leq 0$  e que  $V(PR_{v^0}) = V(R)$ , implicando que

$$V(PR_{\bar{u}^1, \bar{u}^2}) - V(R) = v^0 (Ay^0 - b) \leq 0,$$

logo,  $V(D2) < V(R)$ .

Tal como acontece no método de decomposição lagrangeana descrita na seção anterior, aqui também a diferença entre  $V(PR_{\bar{u}^1, \bar{u}^2})$  e  $V(PR_{v^0})$  pode ser causada por dois fatores:

- (i) a não satisfação de complementariedade das folgas, e
- (ii) o fato que

$$V(D2) = \min_{u^1, u^2} V(PR_{u^1, u^2}) \leq V(PR_{u^1, u^2}), \text{ para todo } u^1 \text{ e } u^2.$$

A seguir prova-se um resultado que estabelece uma relação entre os limites fornecidos pelos dois esquemas de decomposição lagrangeana apresentados.

**Lema V.2:**  $V(D1) \leq V(D2)$ .

**Demonstração:**

Sejam  $(u_0^1, u_0^2) \in SO(D2)$  e considere-se  $\bar{w} = (u_0^1 \quad u_0^2 \quad A_2)$ . Seja também  $(x^0, y^0) \in SO(PR_{\bar{w}})$ . Então

$$\begin{aligned} V(PR_{\bar{w}}) &= \max(f - \bar{w}) x && + \text{máx } \bar{w} y \\ \text{s.a } Cx &\leq d && \text{s.a } Ay \leq b \\ x &\in X, && y \in Y. \end{aligned}$$

$$= (f - \bar{w}) x^0 + \bar{w} y^0$$

Substituindo  $\bar{w}$  têm-se que

$$V(PR_{\bar{w}}) = fx^0 - u_0^1 x_0^1 - u_0^2 A_2 x_0^2 + u_0^1 y_0^1 + u_0^2 A_2 y_0^2.$$

Como  $(x^0, y^0)$  é também solução Ótima de  $(PR_{u_0^1, u_0^2})$ , então

$$V(PR_{\bar{w}}) = V(PR_{u_0^1, u_0^2}) = V(D2).$$

Mas,  $V(D1) \leq V(PR_{\bar{w}})$ , logo  $V(D1) \leq V(D2)$ .

**Teorema V.3**  $V(D1) \leq V(D2) \leq V(R)$ .

**Demonstração:** Imediata dos lemas (V.1) e (V.2).

Sejam (RA) e (RC) os problemas duais lagrangeanos resultantes da dualização das restrições  $Ax \leq b$  e  $Cx \leq d$ , respectivamente, e, seja  $(\bar{P})$  a relaxação de programação linear do problema (P). No teorema a seguir mostra-se um resultado que permite derivar uma condição necessária para que os limites fornecidos por (D2) sejam melhores que os fornecidos por (RA).

**Teorema V.4** Se o problema em variáveis  $y$  satisfaz a propriedade da integralidade então

$$V(D2) = V(RA) \leq V(RC) = V(\bar{P}),$$

**Demonstração:** Se o problema em variáveis  $y$  satisfaz a propriedade da integralidade, então

$$V(RA) \leq V(RC) = V(\bar{P}).$$

Por outro lado

$$V(D1) = V(RA), \text{ (por Teorema V.2) e,}$$

$$V(D1) \leq V(D2) \leq V(RA), \text{ (por Teorema V.3).}$$

$$\text{Logo, } V(D2) = V(RA).$$

Ou seja, uma condição necessária para que  $V(D2)$  seja estritamente menor que  $V(RA)$  é que o problema em variáveis  $y$  não tenha a propriedade da integralidade.

### V.3.2 - Observações e Comentários

Uma observação interessante é que a qualidade dos limites fornecidos pelo esquema de decomposição apresentado nesta seção, depende da formulação do problema equivalente a (P), a partir do qual vão ser derivados os limites. Por exemplo, considere-se o problema resultante de acrescentar em (P2) e restrição

$$C_2 y^2 = C_2 x^2 \quad (V.13)$$

onde  $C_2$  é uma submatriz ou  $C$  de dimensões apropriadas. Seja (P3) o novo problema.

Seja (D3) o problema dual lagrangeano fornecido pela dualização das restrições (V.9), (V.10) e (V.13). Sejam  $(\lambda^1)^T \in \mathbb{R}^k$ ,  $(\lambda^2)^T \in \mathbb{R}^p$  e  $(\lambda^3)^T \in \mathbb{R}^{m-p}$ , os respectivos vetores duais. Seja também  $(PR_{\lambda^1, \lambda^2, \lambda^3})$  o problema lagrangeano resultante da relaxação. Então pode-se provar o seguinte resultado:

**Teorema V.5**  $V(D3) \leq V(D2)$ .

**Demonstração:** Sejam  $(u_o^1, u_o^2) \in SO(D2)$  e considere-se  $\bar{\lambda}^1 = u_o^1$ ,  $\bar{\lambda}^2 = u_o^2$  e  $\bar{\lambda}^3$  o vetor nulo. Seja também  $(x^o, y^o) \in SO(PR_{\bar{\lambda}^1, \bar{\lambda}^2, \bar{\lambda}^3})$ . É fácil mostrar que:

$$V(PR_{\bar{\lambda}^1, \bar{\lambda}^2, \bar{\lambda}^3}) = V(PR_{u_o^1, u_o^2}) = V(D2)$$

e, como  $V(D3) \leq V(PR_{\bar{\lambda}^1, \bar{\lambda}^2, \bar{\lambda}^3})$  o teorema fica demonstrado.

O Teorema V.5 mostra que é possível melhorar o limite derivado de (D2) simplesmente acrescentando algumas restrições adicionais (sem perder a equivalência com o problema original), ou escolhendo adequadamente a(s) matriz(es) para particionar. Di

to de outra maneira, na medida que o conjunto de restrições de acoplamento entre  $x$  e  $y$  (as quais garantem a equivalência) se aproximam à restrição " $y = x$ ", o limite ficará mais próximo de  $V(D1)$ .

Obviamente as restrições adicionais, a serem acrescentadas e os particionamentos dos vetores das variáveis de decisão a serem realizados, dependem da estrutura do problema original (P).

Uma outra observação, derivada dos comentários acima, é que uma boa formulação do problema equivalente a (P), que será usado para obter limites, implicará num número de variáveis duais que poderá ser muito menor que no esquema de decomposição descrito na seção (V.2). Isto possibilitará um menor esforço computacional no cálculo dos limites.

Em relação ao método que poderia ser utilizado para resolver o problema dual lagrangeano, dado que este esquema está "orientado às restrições" (e não "às variáveis", como no enfoque de GUIGNARD e KIM (18) e (19)), parece ser que o método mais indicado seria o clássico método do subgradiente.

#### V.4 - UM EXEMPLO

O exemplo discutido a seguir foi proposto e analisado por GUIGNARD e KIM em (18). Considere-se o seguinte problema da mochila generalizado:



$$(P) \quad \max 2x_1 + 3x_2 + 4x_3$$

$$\text{s.a} \quad 12x_1 + 19x_2 + 30x_3 \leq 46 \quad (V.14)$$

$$49x_1 + 40x_2 + 31x_3 \leq 76 \quad (V.15)$$

$$x_j \in \{0,1\}, \quad j = 1, 2, 3 \quad (V.16)$$

Sejam (R1) e (R2) os problemas duais lagrangeanos obtidos relaxando em (P) as restrições (V.14) e (V.15), respectivamente. Sejam  $v^1$  e  $v^2$  os multiplicadores duais respectivos. Em (18) prova-se que  $V(R1) = 6,6$  e  $V(R2) = 5,84$ . Estes mínimos são alcançados em  $v^1 = 4/30$  no caso de (R1) e  $v^2 = 2/49$  no caso de (R2). Na mesma referência mostra-se também que  $V(P) = 4$ , que  $V(\bar{P}) = 6,688$  (onde  $\bar{P}$  é a relaxação de programação linear de (P)), e que o mínimo de  $V(PR_w)$  é alcançado em  $w_1 = 2$ ,  $w_2 = 0,5$  e  $w_3 = 1,5$ , sendo  $V(D1) = 4,5$ .

Para aplicar o novo esquema de decomposição lagrangeana proposto na seção anterior, pode-se considerar o seguinte problema equivalente a (P):

$$(P1) \quad \max 2x_1 + 3x_2 + 4x_3$$

$$\text{s.a} \quad 49y_1 + 60y_2 + 31y_3 \leq 76 \quad (V.17)$$

$$y_2 = x_2 \quad (V.18)$$

$$49y_1 + 31y_3 = 49x_1 + 31x_3 \quad (V.19)$$

$$12x_1 + 19x_2 + 30x_3 \leq 46 \quad (V.20)$$

$$x_j \in \{0,1\}, \quad j = 1, 2, 3 \quad (V.21)$$

$$y_j \in \{0,1\}, \quad j = 1, 2, 3 \quad (V.22)$$

Dualizando (V.18) e (V.19) associando-lhes os multiplicadores  $u^1$  e  $u^2$ , respectivamente, obtém-se o seguinte problema lagrangeano:

$$V(\text{PR}_{u^1, u^2}) = \max(2 - 49u^2) x_1 + (3 - u^1) x_2 + (4 - 31u^2) x_3$$

$$\text{s.a (V.20) e (V.21)}$$

$$+ \max 49u^2 y_1 + u^1 y_2 + 31u^2 y_3$$

$$\text{s.a (V.17) e (V.22).}$$

O problema dual lagrangeano de (P1) é

$$(\text{DP1}) \min_{u^1, u^2} V(\text{PR}_{u^1, u^2}) .$$

Neste exemplo, pode-se mostrar que se  $u^2 = 2/49$ , o mínimo é alcançado para qualquer  $u^1$  no intervalo  $[13/49, 36/49]$  e

$$V(\text{PR}_{u^1, 2/49}) = 4,735.$$

A Figura (V.1) representa os valores de  $V(\text{PR}_{u^1, 2/49})$  em termos de  $u^1$ .

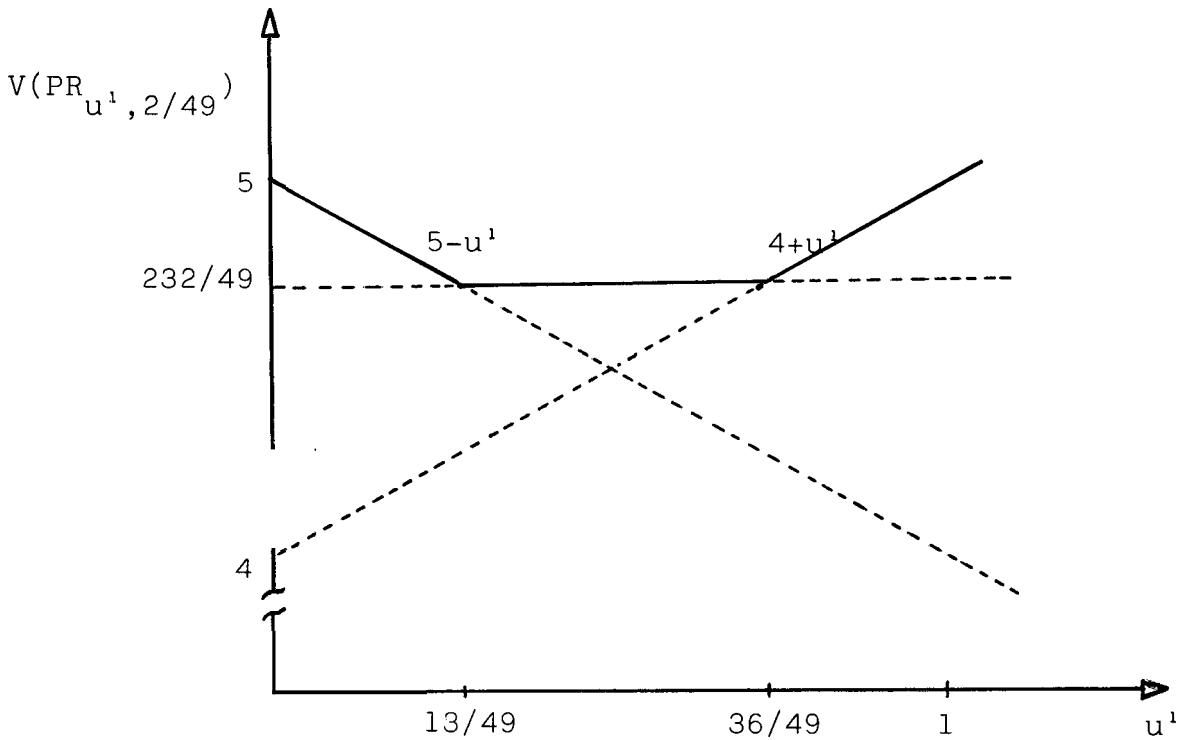


Fig. V.1

Uma maneira de calcular  $V(DP1)$  é a seguinte: como o conjunto de soluções viáveis de  $(PR_{u^1, u^2})$  é finito, ele pode ser expresso assim:

$$\begin{aligned} \{(x^k, y^k) | k = 1, 2, \dots, K\} = \\ = \{[(0,0,0), (1,0,0)], [(0,0,1), (0,1,0)], \dots\} \end{aligned}$$

Neste caso  $K = 29$ . Portanto o dual lagrangeano pode ser formulado como o seguinte problema de programação linear:

(DP1)  $\min t$

$$\begin{aligned} \text{s.a } t \geq & 2x_1^k + 3x_2^k + 4x_3^k + u^1(y_2^k - x_2^k) + \\ & + u^2(49y_1^k + 31y_3^k - 49x_1^k - 31x_3^k), \quad k = 1, \dots, 29. \end{aligned}$$

Uma solução Ótima deste problema (obtida usando o pacote de programação matemática LINDO) é

$$V(DP1) = 4,734694 \approx 4,735$$

$$u^1 = 0,734694 \quad [= 36/49]$$

$$u^2 = 0,040816 \quad [= 2/49]$$

Esta solução coincide com a da figura (V.1).

O exemplo ilustra claramente que, usando menos multiplicadores que variáveis, é possível obter um limite superior bastante próximo ao melhor limite conhecido para o problema (P), e substancialmente menor que o limite superior fornecido pela melhor relaxação lagrangeana convencional. É importante notar que neste exemplo  $V(P)$  é inteiro, logo tanto 4,5 como 4,735 implicam separadamente que  $V(P) \leq 4$ .

Para mostrar como o limite superior pode mudar de acordo com a formulação do problema, considere-se o seguinte problema equivalente a (P):

$$(P2) \quad \max 2x_1 + 3x_2 + 4x_3$$

$$\text{s.a} \quad 12y_1 + 19y_2 + 30y_2 \leq 46$$

$$y_2 = x_2$$

$$12y_1 + 30y_1 = 12x_1 + 30x_3$$

$$49x_1 + 40x_2 + 31x_3 \leq 76$$

$$x_j, y_i \in \{0,1\}, \quad j = 1, 2, 3$$

Dualizando as restrições de acoplamento entre  $x$  e  $y$ , e resolvendo o problema dual lagrangeano de maneira descrita no caso anterior, obtém-se um limite superior igual a 5,2. Este valor, embora seja melhor que  $V(R1)$ ,  $V(R2)$  e  $V(\bar{P})$ , é pior que o limite obtido usando a formulação (P1).

#### V.5 - APLICAÇÃO AO PROBLEMA DE LOCALIZAÇÃO CAPACITADO

Nesta seção será usada a seguinte formulação do problema de localização capacitado:

$$(PLC) \quad \max \sum_i \sum_j p_{ij} x_{ij} - \sum_i f_i y_i$$

$$\text{s.a} \quad \sum_i x_{ij} = 1, \quad j \in J \quad (V.23)$$

$$x_{ij} \leq y_i, \quad i \in I, \quad j \in J \quad (V.24)$$

$$\sum_i b_i y_i \geq \sum_j q_j \quad (V.25)$$

$$\sum_j q_j x_{ij} \leq b_i y_i, \quad i \in I \quad (V.26)$$

$$0 \leq x_{ij} \leq 1, \quad i \in I, \quad j \in I \quad (V.27)$$

$$y_i \in \{0,1\}, \quad i \in I \quad (V.28)$$

A notação utilizada nesta formulação é a mesma do capítulo I. Pode-se notar aqui que as restrições (V.24) e (V.25) são redundantes.

GUIGNARD e KIM demonstram em (19) que o esquema de decomposição lagrangeana que eles descrevem produz o seguinte resultado:

$$V(\text{PLC}) \leq V(\text{R1}) \leq V(\text{D}) \leq V(\overline{\text{PLC}}), \text{ e}$$

$$V(\text{PLC}) \leq V(\text{R2}) \leq V(\text{D}) \leq V(\overline{\text{PLC}}),$$

onde (D) é o problema dual lagrangeano associado com a decomposição. (R1) e (R2) denotam os duais lagrangeanos correspondentes às relaxações sugeridas por NAUSS (31) e VAN ROY (39), respectivamente;  $(\overline{\text{PLC}})$  é a relaxação de programação linear de (PLC).

Para aplicar o esquema de decomposição proposto nesta tese substitui-se a restrição (V.24) por

$$\sum_j x_{ij} \leq n y_i, \quad i \in I, \quad (\text{V.24}')$$

e cria-se uma cópia de cada uma das variáveis de decisões. Sejam  $z_{ij}$  e  $t_i$ ,  $i \in I$ ,  $j \in J$ , as cópias de  $x_{ij}$  e  $y_i$ , respectivamente. Pode-se assim, formular o seguinte problema equivalente a (PLC):

$$\begin{aligned} (\text{PLC}') \quad \max \quad & \sum_i \sum_j p_{ij} x_{ij} - \sum_i f_i y_i \\ \text{s.a} \quad & \sum_i z_{ij} = 1, \quad j \in J \end{aligned} \quad (\text{V.31})$$

$$\sum_j z_{ij} \leq n t_i, \quad i \in I \quad (\text{V.32})$$

$$\sum_i t_i b_i \geq \sum_j q_j \quad (\text{V.33})$$

$$z_{ij}, t_i \in \{0,1\}, \quad i \in I, \quad j \in J \quad (\text{V.34})$$

$$t_i = y_i, \quad i \in I \quad (\text{V.35})$$

$$\sum_i z_{ij} = \sum_i x_{ij}, \quad j \in J \quad (\text{V.36})$$

$$\sum_j z_{ij} = \sum_j x_{ij}, \quad i \in I \quad (\text{V.37})$$

$$\sum_j q_j x_{ij} \leq b_i y_i, \quad i \in I \quad (\text{V.38})$$

$$\sum_i b_i y_i \geq \sum_i q_j \quad (\text{V.39})$$

$$0 \leq x_{ij} \leq 1, \quad i \in I, j \in J \quad (\text{V.40})$$

$$y_i \in \{0,1\}, \quad i \in I \quad (\text{V.41})$$

Considere-se a relaxação lagrangeana das restrições (V.35), (V.36) e (V.37), às quais associam-se, respectivamente, as variáveis duais irrestritas  $u_i$ ,  $v_j$  e  $w_i$ ,  $i \in I$ ,  $j \in J$ . O problema lagrangeano resultante é

$$\begin{aligned} (\text{RLC}_{u,v,w}) \max \{ & \sum_i \sum_j p_{ij} x_{ij} - \sum_i f_i y_i + \sum_i u_i (t_i - y_i) \\ & + \sum_j v_j (\sum_i z_{ij} - \sum_i x_{ij}) + \sum_i w_i (\sum_j z_{ij} - \sum_j x_{ij}) \}, \end{aligned}$$

s.a (V.31)-(V.34) e (V.38)-(V.41).

Depois de algumas operações pode-se escrever

$$\begin{aligned} (\text{RLC}_{u,v,w}) \max \sum_i \sum_j \bar{p}_{ij} x_{ij} - \sum_i \bar{f}_i y_i + \max \sum_i \sum_j (w_i + v_j) z_{ij} + \\ + \sum_i u_i t_i \end{aligned}$$

s.a (V.38)-(V.42) s.a (V.31)-(V.34), onde

$$\bar{p}_{ij} = p_{ij} - w_i - v_j \text{ e } \bar{f}_i = f_i + u_i.$$

Portanto, dados  $u$ ,  $v$  e  $w$ , o problema lagrangeano se decompõe aditivamente em dois subproblemas. O primeiro é o seguinte problema de mochila 0-1 de fácil solução:

$$\begin{aligned} \max \quad & \sum_i (r_i - \bar{f}_i) y_i \\ \text{s.a.} \quad & \sum_i b_i y_i \geq \sum_j q_j \\ & y_i \in \{0,1\}, \quad i \in I, \end{aligned}$$

onde, para todo  $i \in I$ ,  $r_i$  é o valor Ótimo do problema

$$\begin{aligned} (K^i) \quad \max \quad & \sum_j \bar{p}_{ij} x_{ij} \\ \text{s.a.} \quad & \sum_j q_j x_{ij} \leq b_i, \quad j \in J \\ & 0 \leq x_{ij} \leq 1, \quad j \in J. \end{aligned}$$

Aqui  $r_i$  representa o efeito de incluir na solução o armazém  $i$ .

O segundo subproblema corresponde a um problema de localização não capacitado, nas variáveis  $z_{ij}$  e  $t_i$ , com a restrição adicional (V.33). Vários métodos eficientes podem ser usados para resolver este problema: ver por exemplo (9), (10) e (15).

O problema dual lagrangeano associado com esta decomposição é

$$(DLC) \quad \min_{u,v,w} V(RLC_{u,v,w})$$



Na relaxação lagrangeana clássica, podem-se dualizar as restrições (V.23) e (V.24') associando-lhes as variáveis duais  $x_j \in \mathbb{R}$ ,  $j \in J$  e  $\delta_i \geq 0$ ,  $i \in I$ , respectivamente. O problema dual lagrangeano é

$$(R) \quad \min_{\lambda, \delta} V(LR_{\lambda, \delta}),$$

onde

$$V(LR_{\lambda, \delta}) = \max \left\{ \sum_i \sum_j (p_{ij} - \delta_i - \lambda_j) x_{ij} - \sum_i (f_i - n\delta_i) y_i + \sum_j \lambda_j \right. \\ \left. \text{s.a. (V.25)-(V.28)}. \right.$$

Não é difícil mostrar que se são considerados os multiplicadores Ótimos  $\lambda^0$  e  $\delta^0$ , e se faz  $\bar{v}_j = \lambda_{j'}^0$ ,  $\bar{w}_i = \delta_i^0$  e  $\bar{u}_i = n\delta_i^0$ ;  $i \in I$ ,  $j \in J$ , então

$$V(LR_{\lambda^0, \delta^0}) - V(RLC_{\bar{u}, \bar{v}, \bar{w}}) = \sum_i \delta_i^0 (nt_i^0 - \sum_j z_{ij}^0)$$

onde  $z_{ij}^0$  e  $t_i^0$  são os valores Ótimos das variáveis  $z_{ij}$  e  $t_i$  em  $(RLC_{\bar{u}, \bar{v}, \bar{w}})$ . Como  $\delta_i^0 \geq 0$ , para todo  $i \in I$ , a equação anterior garante que  $V(RLC) \leq V(R)$ .

No problema de localização tratado nesta tese, a restrição (V.27) deve ser substituída por  $x_{ij} \in \{0, 1\}$ ,  $i \in I$ ,  $j \in J$ . Assim os problemas  $(K^i)$ ,  $i \in I$ , descritos acima são agora problemas da mochila 0-1, fato que aumenta a complexidade do problema lagrangeano.

## CAPÍTULO VI

## LIMITES SUPERIORES: OUTRAS RELAXAÇÕES

## VI.1 - RELAXAÇÃO LAGRANGEANA BASEADA NUM PROBLEMA DE LOCALIZAÇÃO NÃO CAPACITADO

Nesta seção será utilizada a formulação (PLC) do capítulo V, substituindo as restrições (V.27) por

$$x_{ij} \in \{0,1\}, \quad i \in I, \quad j \in J \quad (\text{V.27}')$$

Seja (LC) o problema obtido com esta substituição.

Pode-se definir uma relaxação lagrangeana das restrições de capacidade (V.26), associando-lhes multiplicadores  $\beta_i \geq 0$ ,  $i \in I$ . Obtém-se assim o seguinte problema lagrangeano:

$$(\text{LR}_\beta) \max \left\{ \sum_i \sum_j p_{ij} x_{ij} - \sum_i f_i y_i + \sum_i \beta_i (b_i y_i - \sum_j q_j x_{ij}) \right\}$$

s.a (V.23)-(V.25), (V.27') e (V.28), ou, equivalentemente

$$(\text{LR}_\beta) \max \left\{ \sum_i \sum_j (p_{ij} - \beta_i q_j) x_{ij} - \sum_i (f_i - \beta_i b_i) y_i \right\}$$

s.a (V.23)-(V.25), (V.27') e (V.28).

Dado o vetor  $\beta$  de multiplicadores, o problema  $(\text{LR}_\beta)$  consiste num problema de localização não capacitado com a restrição adicional (V.25).

A heurística proposta por KLINCEWICZ e LUSS (24) é baseada nesta relaxação. Estes autores relatam resultados computacionais que mostram que os limites superiores que podem ser obtidos, ficam bastante próximos ao valor da solução Ótima do problema. Como a formulação utilizada em (24) não inclui as restrições (V.25), é provável que sua inclusão permita a obtenção de limites um pouco melhores.

## VI.2 - RELAXAÇÃO LAGRANGEANA BASEADA NUM PROBLEMA DE LOCALIZAÇÃO CAPACITADO

Uma outra alternativa que parece interessante explorar é dada a seguir. Substitui-se em (LC) as restrições (V.24) por

$$\sum_j x_{ij} \leq d_i y_i, \quad i \in I \quad (\text{VI.1})$$

onde  $d_i$  representa o número máximo de clientes que poderiam ser atendidos no armazém  $i$ . Estes números podem ser calculados resolvendo, para todo  $i \in I$ , o seguinte problema da mochila 0-1 de fácil solução:

$$\begin{aligned} (D^i) \quad & \max \sum_j \delta_j \\ \text{s.a} \quad & \sum_j q_j \delta_j \leq b_i \\ & \delta_j \in \{0,1\} \end{aligned}$$

Então  $d_i = V(D^i)$  para todo  $i \in I$ .

Agora pode-se definir uma relaxação lagrangeana dualizando as restrições de capacidade (V.26) usando as variáveis duais  $\lambda_i \geq 0$ , para todo  $i \in I$ . O problema lagrangeano resultante é

$$(LR_\gamma) \quad \max \sum_i \sum_j (p_{ij} - \gamma_i q_j) x_{ij} - \sum_i (f_i - \gamma_i b_i) y_i$$

s.a. (V.23), (VI.1), (V.25), (V.27) e (V.28).

Dados os multiplicadores  $\gamma_i$ , o problema  $(LR_\gamma)$  tem a estrutura de um problema de localização capacitado que, apesar de ser um problema de difícil solução, possui algumas características que poderiam permitir tirar vantagens de sua utilização. A mais importante dessas características é que, dados os valores de  $y_i$ , a restrição  $x_{ij} \in \{0,1\}$  poderá ser relaxada substituindo-se por  $0 \leq x_{ij} \leq 1$ , sem alterar a integralidade das  $x_{ij}$ . Ou seja, o problema fica reduzido ao clássico problema de localização capacitado para o qual existem vários algoritmos muito eficientes, como por exemplo, os propostos por CHRISTOFIDES e BEASLEY (8) e BEASLEY (5).

Mesmo assim, o custo computacional da solução do dual lagrangeano poderá ser muito alto. Uma saída, é utilizar como limite superior o valor da função objetivo de  $(LR_\lambda)$  quando  $\gamma_i = 0$  para todo  $i \in I$ , ou seja,  $V(LR_0)$ .

Para testar a alternativa descrita acima, neste trabalho foram calculados os limites superiores para cada um dos problemas dos grupos 1 e 2 da Tabela IV.2. Os resultados obtidos são mostrados na Tabela VI.1.

As colunas (A) mostram os limites superiores usando o método proposto nesta seção e, nas colunas (B) os limites superiores (ou valor da solução Ótima no caso daqueles problemas indicados com \*) obtidos com o método de ajuste de multiplicadores do Capítulo 2.

GRUPO 1					GRUPO 2				
m	n	PROBLEMA. nº	A	B	m	n	PROBLEMA. nº	A	B
3	10	1	207	207	3	10	1*	196	196
		2	228	227			2*	197	197
		3	197	195			3*	185	185
		4	217	208			4*	198	198
		5	209	208			5*	187	187
3	20	1	452	452	3	20	1*	431	431
		2	461	461			2	461	456
		3	406	406			3	406	406
		4	388	388			4	388	379
		5	435	435			5*	430	430
5	10	1	271	276	5	10	1	263	262
		2	266	270			2	255	247
		3	237	238			3	221	221
		4	257	270			4	257	255
		5	238	231			5	221	217
5	20	1	486	513	5	20	1	474	473
		2	528	537			2	510	510
		3	509	515			3	477	473
		4	538	560			4	524	515
		5	518	518			5	493	493

TABELA VI. 1

Os problemas de localização capacitados resultantes de usar esta relaxação foram resolvidos utilizando o algoritmo do capítulo IV. Aqui, o objetivo é apenas mostrar a relação entre os possíveis limites superiores. Obviamente, numa eventual implementação poderão ser usados métodos mais eficientes.

Os dados da Tabela VI.1 mostram que os limites obtidos com o método alternativo (colunas A) são de boa qualidade. Nesta amostra de 40 problemas, este limite foi melhor ou igual que os limites das colunas B, em 65% dos casos. Mais adiante, para os casos onde os limites são diferentes, tem-se a seguinte informação:

NÚMERO DE CASOS EM QUE:	DIFERENÇA MÉDIA (%)	NÚMERO CASOS
LSA < LSB	1,56	14
LSA < LSB	2,67	8

onde LSA e LSB representam os limites superiores das colunas A e B, respectivamente.

## CAPÍTULO VII

## CONCLUSÕES

Nesta tese é discutido um problema de localização capacitado onde existe a restrição adicional de que cada cliente deve ser atendido por um Único armazém. Mostra-se que este problema é NP-árduo e que pode ser considerado como um modelo geral que inclui vários problemas tais como o problema de localização não capacitado, o problema de alocação generalizado, etc.

Na primeira parte desta tese, propõe-se um algoritmo exato do tipo "branch and bound" para resolver o problema. Para isto, são sugeridos, por um lado, um método de ajuste de multiplicadores para calcular limites superiores (usando relaxação lagrangeana) e, por outro lado, vários métodos heurísticos para calcular limites inferiores.

Os testes computacionais realizados sobre uma amostra de 60 problemas gerados aleatoriamente, mostraram que o algoritmo exato proposto pode ser bastante eficiente para problemas de pequeno porte. No entanto, no caso de problemas de tamanho maior que os resolvidos nesses testes, encontrou-se que a eficiência do algoritmo depende fortemente da eficiência do algoritmo utilizado para resolver os problemas de alocação generalizados (PAG), durante a busca na árvore de "branch and bound".

Como uma maneira de diminuir a quantidade de vértices terminais na árvore de busca (e, portanto, o número de GAP's que é resolvido durante a busca), na segunda parte deste trabalho, são sugeridos alguns métodos de obtenção de limites superiores, que poderiam fornecer limites mais fortes (ou melhores) que os fornecidos pelo método usado no capítulo II. Entre estes méto

dos, é discutido um esquema de decomposição lagrangeana estudado por GUIGNARD e KIM (18) e (19), e é sugerido um novo enfoque de decomposição lagrangeana. Este novo enfoque poderá acarretar algumas vantagens computacionais no cálculo dos limites. Embora não se apresentem resultados computacionais usando estes enfoques, provam-se resultados que estabelecem que os limites obtidos por qualquer um dos métodos de decomposição discutidos, poderão ser consideravelmente melhores que os limites obtidos por métodos lagrangeanos clássicos.

Finalmente, são discutidos outros dois métodos para melhorar os limites. O primeiro é sugerido por KLINCEWICZ e LUSS (24). O segundo (sugerido aqui), consiste de uma reformulação do problema e uma relaxação lagrangeana definida sobre o problema reformulado. Os testes computacionais realizados sobre um grupo de 40 problemas gerados aleatoriamente, mostraram que este método é bastante promissor.

Futuras pesquisas sobre o problema, devem incluir fundamentalmente, entre outros aspectos, os seguintes:

- (i) utilização de métodos mais eficientes para resolver o problema de alocação generalizado, por exemplo, o método devido a FISHER et alii (14).
- (ii) Incorporação no algoritmo de "branch and bound" dos novos esquemas de cálculo de limites superiores sugeridos neste trabalho. Isto implicará, eventualmente a introdução de novos critérios de separação ("branching") na árvore de busca.
- (iii) Implementação de novos métodos de redução.



## REFERÊNCIAS BIBLIOGRÁFICAS

- ( 1) BAKER, B. M. - "Linear Relaxations of the Capacitated Warehouse Location Problem", **J. of the Operational Research Society**, Vol. 33, pp. 475-474, (1982).
- ( 2) BAKER, B. M. - "A Partial Dual Algorithm for the Capacitated Warehouse Location Problem", **European Journal of Operational Research**, Vol. 23, pp. 48-56, (1986).
- ( 3) BARTEZZAGHI, E., COLORNI, A. e PALERMO, P. C. - "A Search Tree Algorithm for Plant Location Problems", **European Journal of Operational Research**, Vol. 7, pp. 371-379, (1981).
- ( 4) BARCELÓ, J. e CASANOVAS, J. - "A Heuristic Lagrangean Algorithm for the Capacitated Plant Location Problem", **European Journal of Operational Research**, Vol. 15, PP. 212-226, (1984).
- ( 5) BEASLEY, J. E. - "An Algorithm for Solving Large Capacitated Warehouse Location Problems", **Technical Report**, Department of Management Science, Imperial College, London, England, (1986).
- ( 6) BITRAN, G. R., CHANDRU, V., SEMPOLINSKI, D. E. e SHAPIRO, J. F. - "Inverse Optimization: An Application to the Capacitated Plant Location Problem", **Management Science**, Vol. 27, nº 10, pp. 1120-1141, (1981)
- ( 7) CRHISTOFIDES, N. - **Graph Theory: An Algorithm Approach**, Academic Press, (1975).

- ( 8) CHRISTOFIDES, N. e BEASLEY, J. E. - "Extensions to a Lagrangean Relaxation Approach for the Capacitated Warehouse Location Problem", **European Journal of Operational Research**, Vol. 12, pp. 19-28, (1983).
- ( 9) CORNUEJOLS, G., FISHER, M. L. e NEMHAUSER G. L. - "Location of Bank Accounts to Optimize Float: An Analytic Study of Exact and Approximate Algorithms", **Management Science**, Vol. 23, n° 8, pp. 789-810, (1977).
- (10) ERLKOTTER, D. - "A Dual-Based Procedure for Uncapacitated Facility Location", **Operations Research**, Vol. 26, PP. 992-1009, (1978).
- (11) FAYARD, D. and PLATEAU, G. - "An Algorithm for the Solution of the 0-1 Knapsack Problem", **Computing** Vol. 28,(1982).
- (12) FISHER, M. L. - "The Lagrangean Relaxation Method for Solving Integer Programming Problems", **Management Science**, Vol. 27, n° 1, pp. 1-18, (1981).
- (13) FISHER, M. L. - "An Applications Oriented Guide to Lagrangean Relaxation", **Interfaces**, Vol. 15, n° 2, pp. 10-21, (1985).
- (14) FISHER, M. L., JAIKUMAR, R. e VAN WASSEHOUSE, L. N. - "A Multiplier Adjustment Method for the Generalized Assignment Problem", **Management Science**, Vol. 32, n° 9, pp. 1095-1103, (1986).

- (15) GALVÃO, R. D. e RAGGI, L. A. - "A Method for Solving to Optimality Large Uncapacitated Location Problems", aceito em **Annals of Operations Research**, (1988).
- (16) GEOFFRION, A. M. - "Lagrangean Relaxation for Integer Programming", **Mathematical Programming Study** 2, pp. 82-114, (1974).
- (17) GEOFFRION, A. e McBRIDE, R. - "Lagrangean Relaxation Applied to Capacitated Facility Location Problems", **AIIE Transactions**, Vol. 10, nº 1, pp. 40-47, (1978).
- (18) GUIGNARD, M. e KIM, S. - "Lagrangean Decomposition: A Model Yielding Stronger Lagrangean Bounds", **Mathematical Programming**, Vol. 39, pp. 215-228, (1987).
- (19) GUIGNARD, M. e KIM, S. - "Lagrangean Decomposition: Theory and Applications", **Technical Report** # 93, Department of Statistics, University of Pennsylvania, (1987).
- (20) HELD, M., WOLFE, P. e CROWDER, H. P. - "Validation of Subgradient Optimization", **Mathematical Programming Study** 6, pp. 82-88, (1984).
- (21) JACOBSEN, S. K. - "Heuristics for the Capacitated Plant Location Model", **European Journal of Operational Research**, Vol. 12, pp. 253-261, (1983).
- (22) JORNSTEN, K. e NASBERG, M. - "A New Lagrangian Relaxation Approach to the Generalized Assignment Problem", **European Journal of Operational Research**, Vol. 27, pp. 313-323, (1986).

- (23) KELLY, D. L. e KHUMAWALA, B. M. - "Capacitated Warehouse Location with Concave Costs", **J. of the Operational Research Society**, Vol. 33, pp. 817-826, (1982).
- (24) KLINCEWICZ, J. G. e LUSS, H. - "A Lagrangian Relaxation Heuristic for capacitated Facility Location with Single-Source Constraints", **J. of the Operational Research Society**, Vol. 37, n° 5, pp. 495-500, (1986).
- (25) MARTELLO, S. e TOTH, P. - "Algorithm for the Solution of the 0-1 Single Knapsack Problem", **Computing**, Vol. 21, pp. 81-86, (1978).
- (26) MARTELLO, S. e TOTH, P. - "An Algorithm for the Generalized Assignment Problem", **J. P. Brans, Ed., Operational Research '81**, North Holland, Amsterdam, (1981).
- (27) MARTELLO, S. e TOTH, P. - "Linear Assignment Problems", **Journals of Discrete Mathematic**, Vol. 31, pp. 259-282, (1987).
- (28) MARTELLO, S. e TOTH, P. - "A New Algorithm for the 0-1 Knapsack Problem", **Management Science**, Vol. 34, n° 5, pp. 633-644, (1988).
- (29) MINOUX, M. - **Mathematical Programming: Theory and Algorithms**, John Wiley and Sons Ltd., (1986).
- (30) MACULAN, N., REINOSO, H., GAZMURI, P. e ROBLES, J. M. - "Capacitated Facility Location: Distribution Problem with Complex Side Constraints", 11th Triennial Conference on Operational Research (IFORS'87), Buenos Aires, Argentina, agosto (1987).

- (31) NAUSS, R. M. - "An Improved Algorithm for the Capacitated Facility Location Problem", **J. of the Operational Research Society**, Vol. 29, pp. 1195-1201, (1978).
- (32) NEEBE, A. W. e RAO, M. R. - "An Algorithm for the Fixed-Charge Assigning Users to Sources Problem", **J. of the Operational Research Society**, Vol. 34, nº 11, pp. 1107-1113, (1983).
- (33) REINOSO, H. e MACULAN, N. - "Um Novo Esquema de Decomposição Lagrangeana em Programação Linear Inteira", Relatório Técnico ES-141/88, Programa de Engenharia de Sistemas e Computação, Universidade Federal do Rio de Janeiro, fevereiro (1988).
- (34) ROSS, G. T. e SOLAND, R. M. - "A Branch and Bound Algorithm for the Generalized Assignment Problem", **Mathematical Programming**, Vol. 8, pp. 91-103, (1975).
- (35) ROSS, G. T. e SOLAND, R. M. - "Modelling Facility Location Problems as Generalized Assignment Problems", **Management Science**, Vol. 24, nº 3, (1977).
- (36) SRLKIN, H. M. - **Integer Programming**, Addison-Wesley Publishing Co., (1975).
- (37) SHAPIRO, J. F. - "A Survey of Lagrangean Techniques for Discrete Optimization", **Annals of Discrete Mathematics**, Vol. 5, pp. 113-138, (1979).
- (38) SYSLO, M. M., DEO, N. e KOWALIK, J. S. - **Discrete Optimization Algorithms with PASCAL Programs**, Prentice Hall, Inc. (1983).

- (39) VAN ROY, T. J. - "A Cross Decomposition Algorithm for Capacitated Facility Location", **Operations Research**, Vol. 34, n° 1, (1986).