

# UM ALGORITMO *BRANCH-AND-PRICE* PARA PROBLEMAS DE LOCALIZAÇÃO DE $P$ -MEDIANAS

**Edson L. F. Senne**

Universidade Estadual Paulista – UNESP  
Faculdade de Engenharia de Guaratinguetá – FEG  
12516-410 Guaratinguetá, SP  
elfsenne@feg.unesp.br

**Luiz A. N. Lorena**

Instituto Nacional de Pesquisas Espaciais – INPE/LAC  
12.201-970 São José dos Campos, SP  
lorena@lac.inpe.br

**Marcos A. Pereira**

Instituto Nacional de Pesquisas Espaciais – INPE/LAC  
12.201-970 São José dos Campos, SP  
marcos@lac.inpe.br

## RESUMO

Este trabalho descreve um algoritmo *branch-and-price* para problemas de localização de  $p$  medianas. O problema de  $p$ -medianas objetiva localizar  $p$  centros (medianas) em uma rede com  $n$  nós, de modo a minimizar a soma das distâncias de um ponto de demanda à mediana mais próxima. O processo tradicional de geração de colunas é comparado com uma versão estabilizada que combina geração de colunas e relaxação lagrangeana/*surrogate*. O multiplicador usado na relaxação lagrangeana/*surrogate* modifica o critério de custos reduzidos, fornecendo colunas mais produtivas a cada nó da árvore de busca. Testes computacionais demonstram a efetividade do algoritmo proposto, considerando instâncias do problema difíceis para o processo tradicional de geração de colunas e também algumas instâncias para redes com grande número de nós.

**Palavras-Chave:** Localização,  $P$ -medianas, Otimização, *Branch-and-Price*.

## ABSTRACT

This paper describes a *branch-and-price* algorithm for the  $p$ -median location problem. The objective is to locate  $p$  facilities (medians) such that the sum of the distances from each demand point to its nearest facility is minimized. The traditional column generation process is compared with a stabilized approach that combines the column generation and Lagrangean/*surrogate* relaxation. The Lagrangean/*surrogate* multiplier modifies the reduced cost criterion, providing the selection of new productive columns at the search tree. Computational experiments are conducted considering especially difficult instances to the traditional column generation and also with some large-scale instances.

**Keywords:** Location,  $P$ -median, Large-Scale Optimization, *Branch-and-Price*.

## 1. INTRODUÇÃO

Este trabalho descreve um algoritmo *branch-and price* para o problema de localização de  $p$  medianas. A busca por medianas em uma rede é um problema clássico de localização. O objetivo é localizar  $p$  nós (denominados medianas) de forma a minimizar a soma total das distâncias de cada ponto de

demanda até a mediana mais próxima. O problema é bem conhecido como sendo NP-difícil, o que justifica o grande número de heurísticas propostas para sua resolução.

O uso combinado da relaxação lagrangeana/*surrogate* com métodos de otimização por subgradientes, numa abordagem primal-dual, tem se mostrado eficiente para solução do problema [19]. A relaxação lagrangeana/*surrogate* generaliza a relaxação lagrangeana tradicional, usando informação local de restrições *surrogate* relaxadas na relaxação lagrangeana para acelerar os métodos de subgradientes. Uma busca local é realizada a cada iteração do algoritmo de modo a corrigir os tamanhos de passo a serem usados na abordagem por subgradientes. O ganho em tempo computacional pode ser substancial para grandes problemas [16, 19].

O processo de geração de colunas é uma ferramenta poderosa para a solução de grandes problemas de programação linear. Esta técnica pode ser empregada quando todas as colunas do problema não são conhecidas *a priori* e uma enumeração completa das colunas não for possível, ou quando o problema é reescrito usando a decomposição de Dantzig-Wolfe (em que as colunas correspondem aos pontos extremos de um certo conjunto de restrições) [4]. O método de geração de colunas tem sido explorado em várias aplicações, tais como problemas de corte, roteamento de veículos e escala de tripulações [5, 6, 7, 11, 12, 22, 23]. Em sua forma clássica, o método itera entre um problema mestre restrito e um subproblema gerador de colunas. A solução do problema mestre fornece os multiplicadores duais que serão usados no subproblema para determinar as novas colunas a serem incluídas no problema mestre.

Em muitos casos, a aplicação direta do processo de geração de colunas resulta em baixa convergência [17]. Senne and Lorena [20] apresentaram recentemente um algoritmo estabilizado para problemas de  $p$ -medianas (ver também [14]). Mostra-se nestes trabalhos que a relaxação lagrangeana/*surrogate* consegue acelerar o processo de geração de colunas, produzindo conjuntos de colunas mais produtivas e reduzindo os conhecidos problemas de estabilidade, principalmente para grandes instâncias. Outras tentativas de estabilização já foram consideradas anteriormente, como os métodos Boxstep [15], os métodos Bundle e o método *Analytic center cutting plane* [8]. Neame [17] descreve estes e outros métodos alternativos recentes para estabilizar a solução dual (du Merle et al. [9]).

Os algoritmos do tipo *branch-and-price* têm sido implementados como busca em árvore, empregando geração de colunas a cada nó de busca (ver [1], para uma revisão). Neste trabalho, alguns aspectos do algoritmo, tais como a regra de ramificação e o regime de busca, são determinados considerando o problema de  $p$ -medianas como um problema de partição não-capacitado. O multiplicador usado na relaxação lagrangeana/*surrogate* modifica o critério de custo reduzido, fornecendo novas colunas a cada nó da árvore de busca. Alguns testes computacionais foram realizados comparando o desempenho das relaxações lagrangeana e lagrangeana/*surrogate* no método *branch-and-price*, considerando, especialmente, instâncias difíceis para o processo tradicional de geração de colunas e também instâncias para redes com grande número de nós.

O artigo está organizado da forma descrita a seguir. A Seção 2 resume a abordagem estabilizada apresentada em [20] do processo de geração de colunas para problemas de  $p$ -medianas. A Seção 3 destaca os aspectos relevantes desta implementação do algoritmo *branch-and-bound*. A Seção 4 mostra os resultados computacionais obtidos e na Seção 5 são apresentadas as conclusões do trabalho.

## 2. GERAÇÃO DE COLUNAS ESTABILIZADA PARA P-MEDIANAS

O problema de  $p$ -medianas considerado neste trabalho pode ser formulado como o seguinte problema de programação inteira binária:

$$(Pmed): \quad v(Pmed) = \text{Min} \sum_{i=1}^n \sum_{j=1}^n d_{ij} x_{ij}$$

$$\text{sujeito a:} \quad \sum_{i=1}^n x_{ij} = 1 \quad j \in N \quad (1)$$

$$\sum_{i=1}^n x_{ii} = p \quad (2)$$

$$x_{ij} \leq x_{ii} \quad i, j \in N \quad (3)$$

$$x_{ij} \in \{0, 1\} \quad i, j \in N \quad (4)$$

onde:

$[d_{ij}]_{n \times n}$  é uma matriz simétrica de custo (distância), com  $d_{ii} = 0, \forall i$ ;

$p$  é o número de medianas a serem localizadas;

$n$  é o número de nós da rede, com  $N = \{1, \dots, n\}$ ;

$[x_{ij}]_{n \times n}$  é uma matriz de alocação, com  $x_{ij} = 1$  se a mediana  $i$  está alocada ao nó  $j$ , e  $x_{ij} = 0$ , caso contrário;  $x_{ii} = 1$  se o nó  $i$  é uma mediana e  $x_{ii} = 0$ , caso contrário.

As restrições (1) e (3) garantem que cada nó  $j$  é alocado a apenas um nó  $i$ , que deve ser uma mediana. A restrição (2) determina o número exato,  $p$ , de medianas a serem localizadas, e a restrição (4) fornece as condições de integralidade.

O problema (*Pmed*) é uma formulação clássica explorada em muitos trabalhos. Garfinkel et al. [10] e Swain [21] foram os primeiros a aplicar a decomposição de Dantzig-Wolfe ao problema (*Pmed*), obtendo o seguinte problema de partição de conjuntos com uma restrição de cardinalidade:

$$\text{(SP-Pmed):} \quad v(\text{SP-Pmed}) = \text{Min} \sum_{k=1}^m c_k x_k$$

$$\text{sujeito a:} \quad \sum_{k=1}^m A_k x_k = 1 \quad (5)$$

$$\sum_{k=1}^m x_k = p \quad (6)$$

$$x_k \in \{0,1\}$$

onde:

$S = \{S_1, \dots, S_m\}$ , é o conjunto de todos os subconjuntos de  $N$ ;

$A = [a_{ik}]_{n \times m}$  é uma matriz com  $a_{ik} = 1$  se  $i \in S_k$ , e  $a_{ik} = 0$ , caso contrário;

$$c_k = \text{Min}_{j \in S_k} \left( \sum_{i \in S_k} d_{ij} \right).$$

Para cada subconjunto  $S_k$ , a mediana aberta é decidida quando o custo  $c_k$  é calculado, e assim as colunas de (*SP-Pmed*) consideram implicitamente o conjunto de restrições (3) de (*Pmed*). As restrições (1) e (2) são conservadas e atualizadas para (5) e (6), de acordo como o processo de decomposição de Dantzig-Wolfe.

Neste artigo, considera-se uma versão relaxada do problema (*SP-Pmed*), para a qual  $x_k \in [0,1]$ , e será denotado como (*LP-Pmed*). Este problema será considerado como o problema mestre restrito no contexto de geração de colunas [1].

Senne e Lorena [19] apresentaram a relaxação lagrangeana/*surrogate* para o problema de  $p$ -medianas. Uma descrição geral da relaxação lagrangeana/*surrogate* aparece no trabalho de Narciso e Lorena [16]. Para um dado  $t \geq 0$  e  $\mathbf{p} \in R_+^n$ , a relaxação lagrangeana/*surrogate* do problema ( $Pmed$ ) é dada por:

$$(LS_t Pmed^P): \quad v(LS_t Pmed^P) = \text{Min} \sum_{i=1}^n \sum_{j=1}^n (d_{ij} - t\mathbf{p}_j) x_{ij} + t \sum_{j=1}^n \mathbf{p}_j$$

sujeito a: (2), (3) e (4).

$(LS_t Pmed^P)$  é resolvido considerando implicitamente a restrição (2), decompondo-se o problema no índice  $i$ , obtendo-se os seguintes  $n$  subproblemas:

$$\text{Min} \sum_{j=1}^n (d_{ij} - t\mathbf{p}_j) x_{ij}$$

sujeito a (3) and (4).

Cada subproblema é facilmente resolvido fazendo-se  $\mathbf{b}_i = \sum_{j=1}^n [\text{Min}(0, d_{ij} - t\mathbf{p}_j)]$ , e definindo-se  $I$  como o conjunto de índices correspondente aos  $p$  menores  $\mathbf{b}_i$  (aqui a restrição (2) é considerada implicitamente). Assim, uma solução  $x_{ij}^p$  para o problema  $(LS_t Pmed^P)$  é dada por:

$$x_{ii}^p = \begin{cases} 1, & \text{se } i \in I \\ 0, & \text{caso contrário} \end{cases}$$

e para todo  $i \neq j$ ,

$$x_{ij}^p = \begin{cases} 1, & \text{se } i \in I \text{ e } d_{ij} - t\mathbf{p}_j < 0 \\ 0, & \text{caso contrário} \end{cases}$$

A solução lagrangeana/*surrogate* é dada por:

$$v(LS_t Pmed^P) = \sum_{i=1}^n \mathbf{b}_i x_{ii} + t \sum_{j=1}^n \mathbf{p}_j$$

Note que  $x_{ij}^p$  não é uma solução necessariamente viável, mas o conjunto  $I$  identifica as medianas que devem ser usadas para produzir soluções viáveis para  $(Pmed)$ . Estas soluções viáveis podem ser melhoradas por heurísticas de troca, como discutido em [19].

A relaxação lagrangeana usual resulta de  $(LS_t Pmed^P)$  para  $t = 1$ . Para um dado multiplicador  $\mathbf{p}$ , o melhor valor para o fator *surrogate*  $t$  pode ser encontrado resolvendo-se um problema dual lagrangeano local  $\text{Max}_{t \geq 0} v(LS_t Pmed^P)$ . Em [19] apresenta-se um método de busca dicotômica usado para determinar um valor aproximado para o fator  $t$ .

Na integração da relaxação lagrangeana/*surrogate* ao processo de geração de colunas, os multiplicadores  $\mathbf{p}_j$  ( $j = 1, \dots, n$ ) do problema (*LP-Pmed*) são utilizados no problema

$\text{Max}_{t \geq 0} v(\text{LS}_t \text{Pmed}^P)$ . A mediana que contribui com o menor valor para  $v(\text{Max}_{t \geq 0} v(\text{LS}_t \text{Pmed}^P))$  (e seus nós de demanda alocados) é selecionada para produzir uma nova coluna como solução do subproblema:

$$(\text{Sub}_t \text{Pmed}) : \quad v(\text{Sub}_t \text{Pmed}) = \text{Min}_{j \in N} \left[ \text{Min}_{y_{ij} \in \{0,1\}} \sum_{i=1}^n (d_{ij} - t\mathbf{p}_i) y_{ij} \right]$$

Seja  $\mathbf{a}$  a variável dual correspondente à restrição (6) e  $j^*$  o índice correspondente ao valor mínimo de  $v(\text{Sub}_t \text{Pmed})$ . Os novos conjuntos  $S_j$  são:  $\{i \mid y_{ij} = 1 \text{ em } (\text{Sub}_t \text{Pmed})\}$  e a coluna  $\left[ \frac{y_{j^*}}{1} \right]$  é acrescentada

ao problema (*LP-Pmed*) se  $\left[ \sum_{i=1}^n (d_{ij^*} - \mathbf{p}_i) y_{ij^*} \right] < |\mathbf{a}|$ . Na verdade, todas as colunas correspondentes a  $\left[ \frac{y_j}{1} \right]$  que satisfazem a expressão:

$$\left[ \text{Min}_{y_{ij} \in \{0,1\}} \sum_{i=1}^n (d_{ij} - \mathbf{p}_i) y_{ij} \right] < |\mathbf{a}|, \quad (8)$$

podem ser acrescentadas ao conjunto de colunas do problema mestre. Para  $t = 1$  (caso lagrangeano usual) este processo é conhecido como *multi-pricing*, no contexto de geração de colunas [1].

### 3. O ALGORITMO BRANCH-AND-PRICE

Discute-se, nesta seção, o algoritmo *branch-and-price* proposto, examinando separadamente o nó raiz da árvore de busca, a regra de ramificação e as condições de poda nos ramos da árvore.

#### 3.1 – O nó raiz da árvore de busca

O seguinte algoritmo é usado na raiz da árvore de busca:

*Algoritmo CG (t)*

- a) Estabelecer um conjunto inicial de colunas para (*LP-Pmed*);
- b) Resolver (*LP-Pmed*) usando o pacote CPLEX [13], obtendo os preços duais  $\mathbf{p}_j$ ,  $j = 1, \dots, n$  e  $\mathbf{a}$ ;
- c) Resolver de forma aproximada (usando uma busca dicotômica) o problema dual lagrangeano/*surrogate* local  $\text{Max}_{t \geq 0} v(\text{LS}_t \text{Pmed}^P)$ , retornando as correspondentes colunas de (*Sub<sub>t</sub>Pmed*);
- d) Acrescentar ao problema (*LP-Pmed*) as colunas  $\left[ \frac{y_j}{1} \right]$  que satisfazem a expressão (8);

- e) Se nenhuma coluna for encontrada no passo (d) ou se  $|v(LP-Pmed) - v(LS_tPmed^p)| < 1$ , então terminar o algoritmo;
- f) Executar os testes de remoção de colunas e retornar ao passo (b).

Fazendo  $t = 1$ , the algoritmo  $CG(I)$  reproduz o processo tradicional de geração de colunas. Neste caso, a busca dicotômica do passo (c) não é necessária e o limitante lagrangeano usual ( $LS_1Pmed^p$ ) resolve implicitamente o problema ( $Sub_1Pmed$ ). Tanto no caso usual como para a relaxação lagrangeana/surrogate, os limites  $v(LP-Pmed)$  e  $v(LS_tPmed^p)$  são calculados a cada iteração do algoritmo.

O seguinte algoritmo é usado para estabelecer o conjunto inicial de colunas para ( $LP-Pmed$ ):

### Algoritmo IC

Seja  $Max\_Cols$  o número máximo de colunas para o conjunto inicial de colunas.

$ncols = 0$ ;

Enquanto ( $ncols < Max\_Cols$ ) fazer:

Seja  $M = \{n_1, \dots, n_p\} \subset N$  um conjunto de nós distintos gerados aleatoriamente.

Para cada  $i = 1, \dots, p$  fazer:

$$S_i = \{n_i\} \cup \{k \in N - M \mid d_{kn_i} = \underset{j \in M}{\text{Min}}(d_{kj})\}$$

$$c_i = \underset{j \in S_i}{\text{Min}} \left( \sum_{k \in S_i} d_{jk} \right)$$

Para cada  $j = 1, \dots, n$  fazer:

$$y_j = 1 \text{ se } j \in S_i$$

$$y_j = 0, \text{ caso contrário}$$

Incluir a coluna  $\begin{bmatrix} y_j \\ 1 \end{bmatrix}$  no conjunto inicial de colunas.

$ncols = ncols + p$ ;

O algoritmo a seguir é usado no passo (f) de  $CG(t)$ :

### Algoritmo RC

Seja  $rc\_mean$  o valor médio dos custos reduzidos para o conjunto inicial de colunas de ( $LP-Pmed$ ) e  $m$  o número total de colunas existentes atualmente em ( $LP-Pmed$ ). Seja  $rc_i$  ( $i = 1, \dots, m$ ) o custo reduzido das colunas atuais de ( $LP-Pmed$ ).

Para cada  $i = 1, \dots, m$  fazer:

Remover a coluna  $i$  do problema ( $LP-Pmed$ ) se  $rc_i > rc\_mean$ .

## 3.2 – A regra de ramificação

A regra de ramificação considera o particionamento com subconjuntos idênticos descrito em Wolsey [22].

Sejam  $q$  e  $r$  os índices de linhas apresentando pares de colunas fracionárias tais como  $\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$  na solução final do problema mestre restrito. Na ramificação da árvore, considera-se que os ramos à esquerda correspondem aos problemas nos quais as colunas contêm as linhas  $q$  e  $r$  (ou ambas aparecem na coluna ou ambas estão fora da coluna) e que os ramos à direita correspondem aos problemas nos quais as colunas não contêm ao mesmo tempo as linhas  $q$  e  $r$  (ou somente a linha  $q$  está presente na coluna, ou somente a linha  $r$  está presente na coluna, ou ambas as linhas  $q$  e  $r$  estão ausentes da coluna).

O par  $(q, r)$  é identificado da forma descrita a seguir.

### Identificação de $q$

Considere que  $\#A$  denota a cardinalidade do conjunto  $A$ . Seja  $X = \{x_1, \dots, x_m\}$  o conjunto de variáveis de decisão não-nulas correspondentes ao conjunto  $S = \{S_1, \dots, S_m\}$  de colunas de  $(LP-Pmed)$ . Seja  $QS(i) = \{S_j \mid i \in S_j, j = 1, \dots, m\}$  para cada índice de linha  $i$  ( $i = 1, \dots, n$ ). Então,  $q$  é escolhido como o índice de linha tal que:  $\#QS(q) > \#QS(i), \forall i$  ( $i = 1, \dots, n, i \neq q$ ). Note que, se  $\#QS(q) = 1$ , então  $X$  é uma solução viável de  $(LP-Pmed)$ .

### Identificação de $r$

Seja  $RS(i) = \{S_j \in QS(k) \mid i \in S_j, j = 1, \dots, m\}$  para cada índice de linha  $i$  ( $i = 1, \dots, n$ ). Seja  $R$  o conjunto de índices de linha  $i$ , para as quais o conjunto  $RS(i)$  é não-vazio, ou seja,  $R = \{i \mid RS(i) \neq \emptyset, i = 1, \dots, n\}$ . Então,  $r$  é escolhido como o índice de linha tal que:  $\#RS(r) < \#RS(i), \forall i \in R, i \neq r$ .

Uma vez determinado o par  $(q, r)$  de índices de linha, o seguinte problema inteiro binário deve ser resolvido nos ramos esquerdos da árvore de busca:

$$v(Sub_rPmed) = \underset{j \in N}{\text{Min}} \left[ \underset{y_{ij} \in \{0,1\}}{\text{Min}} \sum_{i=1}^n (d_{ij} - t\mathbf{p}_i) y_{ij} \right] \quad (9)$$

$$\text{sujeito a:} \quad y_{qj} = y_{rj}, j = 1, \dots, n.$$

Nos ramos direitos da árvore, deve ser resolvido o seguinte problema inteiro binário:

$$v(Sub_rPmed) = \underset{j \in N}{\text{Min}} \left[ \underset{y_{ij} \in \{0,1\}}{\text{Min}} \sum_{i=1}^n (d_{ij} - t\mathbf{p}_i) y_{ij} \right] \quad (10)$$

$$\text{sujeito a:} \quad y_{qj} + y_{rj} \leq 1, j = 1, \dots, n.$$

Observe que em um nó qualquer da árvore de busca o problema inteiro binário a ser resolvido será uma combinação de vários problemas como os dois acima, dependendo do caminho necessário da raiz da árvore até o nó em questão.

### 3.3 – As condições de poda dos ramos da árvore

No algoritmo *branch-and-price* proposto, a árvore binária é construída num processo de busca em profundidade. A cada nó da árvore uma correspondente combinação de problemas (9) e (10) é resolvida usando o pacote CPLEX [13]. Uma poda ocorre em um nó se o correspondente  $v(LP-Pmed)$  ou  $v(LS_Pmed^P)$  são não menores do que a melhor solução viável disponível.

#### 4. TESTES COMPUTACIONAIS

O algoritmo *branch-and-price* descrito na seção anterior foi codificado na linguagem C e executado em um microcomputador Pentium III 1.1GHz com 512MB RAM. Os testes computacionais foram realizados considerando  $Max\_Cols = 3000$ . Os resultados obtidos são mostrados nas tabelas a seguir. Nestas tabelas:

$n$  é o número de nós da rede;  
 $p$  é o número de medianas;  
 Solução é a solução ótima do problema;  
 Num\_Cols é o tamanho máximo do problema, em termos do número de colunas;  
 Tam\_Arv é o número de nós gerados na árvore de busca;  
 Tempo é o tempo computacional total (em segundos).

A Tabela I mostra os resultados obtidos para alguns problemas disponíveis na OR-Library [2].

Tabela I – Resultados obtidos para instâncias da OR-Library

$n$	$p$	Solução	Lagrangeano			Lagrangeano/surrogate		
			Max_Cols	Tam_Arv	Tempo	Max_Cols	Tam_Arv	Tempo
100	20	3034	2172	1	0,28	2124	1	0,31
100	33	1355	1614	1	0,11	1603	1	0,12
200	40	2734	3344	9	39,23	3442	7	67,61
200	67	1255	2046	3	2,81	2013	1	0,49
300	60	2968	4051	15	113,95	3231	5	42,31
300	100	1729	2352	1	1,39	2384	1	1,38
400	80	2845	3669	1	24,29	3167	1	9,83
400	133	1789	2798	1	1,30	2708	1	1,57
500	100	2961	23191	9	3394,57	14417	7	1288,72
500	167	1828	3237	3	59,20	3620	9	223,94
600	120	3033	28357	25	9064,46	6453	5	304,56
600	200	1989	3499	7	204,45	3532	5	241,11
700	140	3013	48579	3	22688,06	7860	1	170,54
700	233	1847	3950	5	91,52	3966	5	159,10
800	267	2026	4966	21	787,65	3756	1	18,75
900	300	2106	3900	5	233,55	4643	15	1390,87
<b>Médias</b>			<b>8857,8</b>	<b>6,9</b>	<b>2294,2</b>	<b>4307,4</b>	<b>4,1</b>	<b>245,1</b>

Pode-se notar pelos resultados que o método *branch-and-price* que utiliza a relaxação lagrangeana/surrogate para determinar as novas colunas do problema mestre restrito consegue resolver os problemas explorando árvores menores e gerando menos colunas.

É bem conhecida a observação de Christofides and Beasley [3] de que as instâncias mais difíceis do problema de  $p$ -medianas para abordagens lagrangeanas com otimização por subgradientes ocorre quando  $p = n/3$ . Para a abordagem de geração de colunas, para um dado valor de  $n$ , quanto maior for a relação  $n/p$  mais difícil se torna o problema. A Tabela II, portanto, apresenta os valores obtidos pelos algoritmos CG(1) e CG(t) para instâncias consideradas difíceis para a abordagem de geração de colunas, onde a relação  $n/p \geq 10$ .



Tabela II – Resultados obtidos para instâncias difíceis da OR-Library

<i>n</i>	<i>p</i>	Solução	Lagrangeano			Lagrangeano/ <i>surrogate</i>		
			Max_Cols	Tam_Arv	Tempo	Max_Cols	Tam_Arv	Tempo
100	5	5819	3230	1	12,98	3444	1	13,61
100	10	4093	5510	9	52,57	3830	7	42,80
100	10	4250	3626	31	78,65	3298	23	84,30
200	20	4445	7657	1	123,56	7409	1	94,75
300	30	4374	6686	1	458,56	8430	3	601,73
400	40	4809	54830	39	21694,66	14297	19	1742,25
500	50	4619	40243	3	19550,80	13911	3	1921,23
<b>Médias</b>			<b>17397,4</b>	<b>12,1</b>	<b>5996,0</b>	<b>7802,7</b>	<b>8,1</b>	<b>643,0</b>

Os resultados confirmam a superioridade da relaxação lagrangeana/*surrogate* para o processo de geração de colunas.

## 5. CONCLUSÃO

Este trabalho apresentou um método *branch-and-price* para a solução de problemas de localização de *p*-medianas. Este método usa um processo de geração de colunas que difere do processo tradicional porque emprega a relaxação lagrangeana/*surrogate*. O multiplicador lagrangeana/*surrogate* modifica o critério de custo reduzido e fornece colunas mais produtivas a cada nó da árvore de busca do que as obtidas pelo processo tradicional. O algoritmo proposto é capaz de encontrar a solução ótima de problemas de *p*-medianas explorando árvores de busca menores, ou seja, em menor tempo computacional.

**Agradecimentos:** Os autores agradecem à FAPESP – Fundação de Amparo à Pesquisa do Estado de São Paulo e ao CNPq – Conselho Nacional de Desenvolvimento Científico e Tecnológico, pelos apoios financeiros recebidos.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Barnhart, C.; Johnson, E.L.; Nemhauser, G.L.; Savelsbergh, M.W.P. and Vance, P.H. *Branch-and-Price: Column Generation for Solving Huge Integer Programs*, Operations Research 46 (1998) 316-329.
- [2] Beasley, J.E. *Or-library: Distributing test problems by electronic mail*. Journal Operational Research Society, 41:1069-1072, 1990.
- [3] Christofides, N.; Beasley, J.E. A tree search algorithm for the *p*-median problems, *European Journal of Operational Research*, 10: 196-204, 1982.
- [4] Dantzig, G.B. and Wolfe, P. *Decomposition principle for linear programs*. Operations Research, 8: 101-111, 1960.
- [5] Day, P.R. and Ryan, D.M. *Flight Attendant Rostering for Short-Haul Airline Operations*, Operations Research 45 (1997) 649-661.
- [6] Desrochers, M. and Soumis, F. *A Column Generation Approach to the Urban Transit Crew Scheduling Problem*, Transportation Science 23 (1989) 1-13.
- [7] Desrochers, M.; Desrosiers, J. and Solomon, M. *A New Optimization Algorithm for the Vehicle Routing Problem with Time Windows*, Operations Research 40 (1992) 342-354.

- [8] du Merle, O.; Goffin, J.L. and Vial, J.P. *On Improvements to the Analytic Centre Cutting Plane Method* Computational Optimization and Applications 11 (1998) 37-52.
- [9] du Merle, O.; Villeneuve, D.; Desrosiers, J. and Hansen, P. *Stabilized column generation*. Discrete Mathematics, 194: 229-237, 1999.
- [10] Garfinkel, R.S.; Neebe, W. and Rao, M.R. *An Algorithm for the M-median Location Problem*. Transportation Science 8: 217-236, 1974.
- [11] Gilmore, P.C. and Gomory, R.E. *A linear programming approach to the cutting stock problem*. Operations Research, 9: 849-859, 1961.
- [12] Gilmore, P.C. and Gomory, R.E. *A linear programming approach to the cutting stock problem - part ii*. Operations Research, 11: 863-888, 1963.
- [13] ILOG Inc., Cplex Division. CPLEX 6.5, 1999.
- [14] Lorena, L.A.N. ; Senne, E.L.F. *A column generation approach to capacitated p-median problems*. Computers and Operations Research, to appear, 2003.
- [15] Marsten, R.M.; Hogan, W. and Blankenship, J. *The Boxstep method for large-scale optimization*. Operations Research, 23: 389-405, 1975.
- [16] Narciso, M.G.; Lorena, L.A.N. *Lagrangean/surrogate Relaxation for Generalized Assignment Problems*. European Journal of Operational Research , 114(1), 165-177, 1999.
- [17] Neame, P.J. *Nonsmooth Dual Methods in Integer Programming* Phd Thesis - Department of Mathematics and Statistics, The University of Melbourne March 1999.
- [18] Reinelt, G. *The traveling salesman problem: computational solutions for TSP applications*. Lecture Notes in Computer Science 840, Springer Verlag, Berlin, 1994.
- [19] Senne, E.L.F.; Lorena, L.A.N. *Lagrangean/Surrogate Heuristics for p-Median Problems*. In Computing Tools for Modeling, Optimization and Simulation: Interfaces in Computer Science and Operations Research, M. Laguna and J.L. Gonzalez-Velarde (eds.) Kluwer Academic Publishers, pp. 115-130, 2000.
- [20] Senne, E.L.F.; Lorena, L.A.N. *Stabilizing column generation using Lagrangean/surrogate relaxation*. European Journal of Operational Research, submitted, 2001.
- [21] Swain, R.W. *A Parametric Decomposition Approach for the Solution of Uncapacitated Location Problems*. Management Science, 21: 955-961, 1974.
- [22] Wolsey, L. A. *Integer programming* John Wiley & Sons, New York, 1998.
- [23] Valério de Carvalho, J.M. *Exact Solution of Bin-Packing Problems Using Column Generation and Branch-and-Bound*, Universidade do Minho, Departamento Produção e Sistemas, Working Paper, 1996.
- [24] Vance, P.H.; Barnhart, C.; Johnson, E.L. and Nemhauser, G.L. *Solving Binary Cutting Stock Problems by Column Generation and Branch-and-Bound*, Computational Optimization and Applications 3 (1994) 111-130.