

UM ALGORITMO BRANCH-AND-CUT PARA UM PROBLEMA DE LOCALIZAÇÃO DISCRETA COMPETITIVA DE HUBS

Antonio Camargo Andrade de Araújo

Departamento de Engenharia de Produção – Universidade Federal Fluminense
Rua Passo da Pátria 156, 24210-240, Niterói, RJ, Brasil
antonio.camargo@id.uff.br

Artur Alves Pessoa

Departamento de Engenharia de Produção – Universidade Federal Fluminense
Rua Passo da Pátria 156, 24210-240, Niterói, RJ, Brasil
artur@producao.uff.br

Marcos Costa Roboredo

Instituto de Matemática e Estatística – Universidade do Estado do Rio de Janeiro
Rua São Francisco Xavier 524, 20550-900, Maracanã, Rio de Janeiro, RJ, Brasil
marcos.roboredo@ime.uerj.br

RESUMO

Neste trabalho tratamos do problema $(r|p)$ hub-centroide $((r|p)H-C)$, onde duas empresas, líder e seguidora, localizam hubs num ambiente competitivo, buscando maximizar a captura de fluxos entre clientes. Os clientes optam pela empresa que oferecer o menor custo, que varia em função da localização dos hubs. O problema consiste na melhor decisão da líder considerando as possíveis respostas da seguidora, que também busca maximizar sua própria captura de fluxo. Uma formulação minimax bilinear para o $(r|p)H-C$ é apresentada e a partir dela um algoritmo *branch-and-cut* é derivado para resolução do problema. Também é proposto um pré-processamento que elimina grande parte das variáveis do modelo. Diversos resultados são apresentados visando mostrar a eficiência e a robustez do método proposto.

PALAVRAS CHAVE. Otimização em dois níveis, Algoritmo *branch-and-cut*, Problema $(r|p)$ hub-centroide.

Área principal: OC - Otimização Combinatória

ABSTRACT

In this paper we treat the $(r|p)$ hub-centroid $((r|p) H-C)$ problem, where two companies, leader and follower, should locate hubs in a competitive environment seeking to maximize the flows captured between customers. Customers choose the company that offers the lowest cost, which varies depending on the location of hubs. The problem consists in the best decision of leader considering possible responses of follower, which also seeks to maximize its own captured flow. A minimax bilinear formulation for $(r|p) H-C$ is presented and a branch-and-cut algorithm is derived from it for solving the problem. It is also proposed a pre-processing that eliminates a lot of the model variables. Several results are presented in order to show the efficiency and the robustness of the method proposed.

KEYWORDS. Bilevel Programming, Branch-and-cut. $(r|p)$ Hub-centroid Problem.

Main area: Combinatorial Optimization

1. Introdução

O modelo de transporte através de hubs é muito utilizado atualmente e propicia vantagens dentre as quais se destaca os ganhos com economias de escala. Ao se utilizar hubs, se ganha a possibilidade de agregar diferentes cargas na mesma movimentação e também possibilita a redução de tempo e custo de transporte. Além disso, reduz os custos com gestão da distribuição e perdas durante o transporte por concentrar os itens passando pelo hub e facilitando assim o controle.

O problema de localização de hubs consiste na escolha da localização dos hubs dentre um conjunto de possíveis locais e alocação dos outros pontos do sistema a cada um desses hubs de modo a minimizar o custo global de operação desse sistema.

Na versão discreta deste problema, o conjunto de possíveis locais para localização de hubs é finito. Em geral, tais problemas levam em consideração que tanto os pontos de demanda, quanto os possíveis locais de instalação de facilidades (no caso os hubs) são nós de um grafo.

A primeira formulação do modelo de localização de hubs (problema da p -hub mediana de alocação única) foi proposta por O'Kelly (1987), através de programação quadrática. Em seguida, o autor propôs duas heurísticas para resolver o problema. Em tal artigo também foi proposta a utilização de uma base de dados CAB *data*, composta de um conjunto de 25 pontos, suas localizações e as matrizes de fluxos e custos entre esses pontos. Tal base de dados vem sendo bastante utilizada desde então para geração de instâncias para problemas de localização de hubs. Campbell (1994) apresentou a primeira formulação de programação linear inteira para o problema. Alumur e Kara (2008) apresentaram como evoluiu a literatura ao longo dos anos sobre os problemas de localização de hubs e as diferentes variações propostas do problema.

Entretanto ao se considerar um ambiente competitivo, onde mais de uma firma pode prestar o serviço, a decisão não pode ser tomada de forma isolada, mas deve levar em conta que já existem hubs de outras firmas e que podem vir a surgir novos hubs dessas firmas. No presente artigo, vamos considerar um duopólio, onde duas firmas tomam suas decisões de modo sequencial, conforme o clássico modelo proposto por Von Stackelberg (1934). Dessa forma, serão consideradas a partir de agora líder, a firma que toma decisão primeiro, e seguidora, a que toma decisão depois já tendo ciência da decisão da líder. Neste contexto, surge o problema deste artigo, que é conhecido como problema $(r|p)$ hub-centroide ($(r|p)H-C$). Tal problema consiste em decidir onde a firma líder deve localizar p hubs sabendo que uma firma seguidora reagirá localizando r hubs, ambas buscando capturar maior fluxo possível entre clientes. Cada um destes fluxos será atendido por um único hub ou um par de hubs da líder ou da seguidora, de acordo com o critério de menor custo de transporte.

Marianov et al. (1999) foi o primeiro a tratar um problema de localização competitiva de hubs, ao propor uma heurística para a solução do problema da seguidora, considerando que a líder já havia instalado seus hubs. Em tal estudo foi adotado como regra no caso de empates, a captura do fluxo pela seguidora. Wagner (2008) mostrou que tal regra de desempate, para $r \geq p$, possibilitaria à seguidora localizar hubs nos mesmos locais que a líder capturando toda demanda. O autor propôs então que os desempates fossem em favor da líder, além de resolver o problema por programação linear inteira. Sasaki (2005) propôs uma solução para uma variação do problema por enumeração e heurística gulosa que resolvia o problema da líder e da seguidora para a base CAB 25 e p e r assumindo valores entre 1 e 3 inclusive.

Mahmutoğulları (2013) definiu formalmente o $(r|p)H-C$, provou que tal problema é NP-difícil e propôs uma solução para o problema baseada numa enumeração inteligente, onde se armazena a melhor solução para a líder até então em termos de captura de fluxo, não precisando percorrer todas possíveis repostas da seguidora em alguns casos. O autor aplicou seu algoritmo na base CAB 25, com p e r assumindo valores de no máximo 5.

O $(r|p)H-C$ é um problema de otimização em dois níveis inteiros (2-PONI), onde o problema de otimização de primeiro nível consiste em escolher p locais onde a líder localizará seus hubs, enquanto o problema do segundo nível consiste em se determinar onde a seguidora

localizará seus r hubs.

Os 2-PONI em geral são difíceis de se resolver de maneira exata. Moore e Bard (1990) propuseram um algoritmo *branch-and-bound* capaz de resolver instâncias relativamente pequenas. Em geral, os procedimentos exatos para 2-PONI são gerados a partir de características específicas dos problemas, como por exemplo métodos de decomposição baseados em desigualdades super válidas (ISRAELI, WOOD, 2002; O'HANLEY, CHURCH, 2011), algoritmos de plano de corte (TASKIN et al., 2009), dentre outros.

Posteriormente, Pessoa et al. (2013) observaram que quando o 2-PONI admite uma formulação minmax bilinear, uma reformulação de Programação Inteira Mista pode ser obtida com número polinomial de variáveis, substituindo o problema de otimização do 2º nível por um conjunto de desigualdades lineares. Isto possibilita o uso eficiente de resolvidores comerciais para resolução exata. Esta ideia foi aplicada para resolver problemas de localização competitiva (ROBOREDO, 2012; ROBOREDO, PESSOA, 2013) e problemas envolvendo fortificações e interdições (PESSOA et al., 2013; ROBOREDO, AIZEMBERG; 2014). Para todas as aplicações, o método foi comparado aos melhores métodos exatos da literatura, obtendo resultados computacionais satisfatórios. Além disso, várias instâncias grandes foram resolvidas com solução ótima pela primeira vez.

As contribuições deste trabalho são três. Primeiramente, nós provamos que o $(r|p)H-C$ é Σ_2^p -difícil, ratificando a complexidade do problema. Com intuito de resolvermos o problema de maneira exata, propomos uma formulação minimax bilinear para o $(r|p)H-C$ e, a partir dela, derivamos um algoritmo *branch-and-cut* para o mesmo. Foi proposto ainda um módulo de pré-processamento capaz de eliminar mais de 50% das variáveis nas instâncias testadas. Os resultados computacionais obtidos pelo nosso método foram comparados aos obtidos por Mahmutogullari (2013), onde nosso método se mostra mais adequado a instâncias grandes, resolvendo, pela primeira vez, 100 instâncias até a otimalidade.

A organização do artigo é a seguinte: a seção 2 apresenta o problema do $(r|p)H-C$, a seção 3 trata da complexidade do problema. A seção 4 mostra a formulação natural do problema com minimax bilinear e a variação ao substituir o problema de otimização do 2º nível por um conjunto de desigualdades lineares, além do pré-processamento que reduz em grande parte o número de variáveis. A seção 4 apresenta resultados computacionais obtidos e a seção 5 apresenta as conclusões e sugestões de trabalhos futuros.

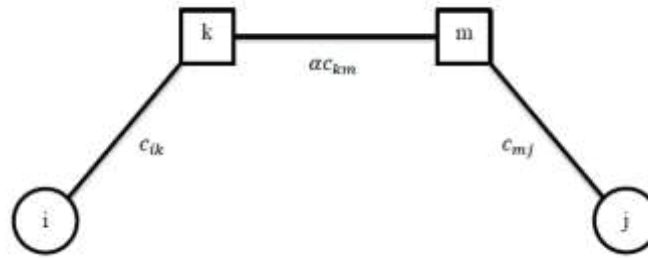
2. Definição do problema

O $(r|p)H-C$ é definido no seguinte contexto. Dado um grafo $G = (N, E)$, sendo N o número de nós e E o número de arestas, temos i e j sempre denotando clientes, localizados nos nós do grafo. Os fluxos entre nós é dado por w_{ij} e o custo por c_{ij} . A empresa líder deve localizar p hubs dentre os $H \subset N$ nós disponíveis para instalação de hub e, sequencialmente, a empresa seguidora deve localizar r hubs dentre os mesmos $H \subset N$. Consideramos que $p \geq 2$ e $r \geq 2$, de modo que as firmas possam se aproveitar das economias de escala na conexão entre hubs. Tal economia de escala é representada no modelo por um fator α que multiplica o custo entre tais hubs, onde $0 \leq \alpha \leq 1$. A Figura 1 mostra como se configura a estrutura de custos do transporte por hubs, utilizando no máximo 2 hubs para o transporte entre clientes. O fluxo entre os nós i e j passa por hubs nos nós k e m .

O custo total é dado pela soma dos custos entre arcos, considerando o fator α para o transporte entre hubs. Definimos então o custo para se transportar uma unidade a partir do nó i até o nó j , passando pelos hubs k e m (considerando que a ordem utilizada entre estes hubs será a de menor custo):

$$c_{ijkm} = \min\{c_{ik} + \alpha * c_{km} + c_{mj}, \quad c_{im} + \alpha * c_{mk} + c_{kj}\} \quad (1)$$

Figura 1: Custos do transporte através dos hubs (Fonte: Mahmutoğulları, 2013)



A empresa ainda tem a possibilidade de realizar o transporte entre i e j , utilizando apenas um hub. Nesse caso, consideramos $m=k$, e $c_{kk} = 0$, então temos:

$$c_{ijkk} = c_{ik} + c_{kj} \quad (2)$$

Os fluxos entre clientes são então atendidos por um único hub ou um par de hubs, da líder ou da seguidora, optando pela companhia que conseguir proporcionar menor custo com o conjunto de hubs alocados. Os eventuais empates são decididos em favor da líder já que o cliente já estaria sendo atendido pela líder no momento que a seguidora entra no mercado e, em teoria, não teria nenhuma vantagem em trocar de firma. Consideramos ainda que, para um dado par de clientes, todo seu fluxo será atendido, ou pela líder ou pela seguidora. Assim, não existe a possibilidade de uma demanda não ser atendida.

O (r|p)H-C consiste em decidir onde a líder deve localizar seus p hubs maximizando seu fluxo atendido, ciente de que a seguidora responderá em seguida racionalmente, ou seja, localizará r hubs de modo a maximizar sua própria demanda capturada para aquela escolha da líder.

3. Complexidade do (r|p)H-C

Nesta seção vamos provar que o problema é \sum_2^P -Difícil. Assim, este problema é mais difícil do que qualquer problema de otimização cuja versão de decisão pertence a NP. A dificuldade deste problema vem do fato que para avaliar uma estratégia da líder é necessário resolver um problema NP-difícil para otimizar a estratégia da seguidora.

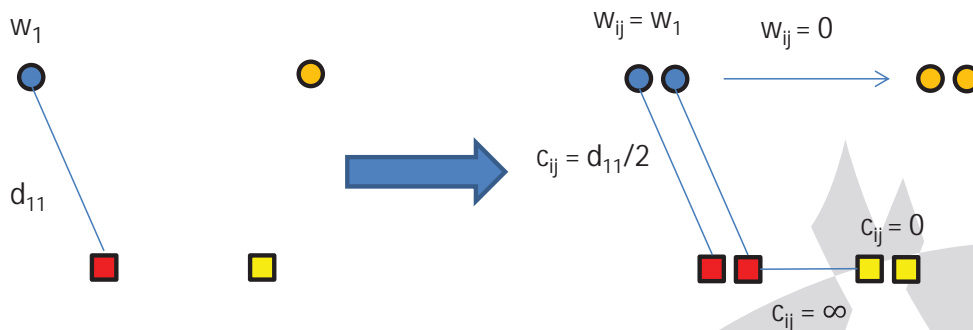
Para realizarmos a prova, vamos fazer uma redução a partir do problema (r|p)-centroide (HAKIMI, 1983). Tal problema foi provado ser \sum_2^P -Difícil por Noltemeier et al (2007) e é definido a seguir.

O (r|p)-centroide considera duas firmas não cooperativas (líder e seguidora). A líder tem que localizar p facilidades em um espaço de decisão sabendo que a seguidora reagirá localizando r facilidades. O espaço de decisão é um grafo $G = (V,E)$ onde cada vértice $v \in V$ é um consumidor ou um candidato a facilidade para a líder ou a seguidora. Como resultado, V pode ser particionado em dois subconjuntos disjuntos I e J , onde I é o conjunto de candidatos a facilidades e J o conjunto de clientes. O conjunto de arestas E de G possui uma aresta $e = \{i, j\}$ para cada $i \in I$ e $j \in J$, com uma distância associada d_{ij} . Cada consumidor j possui uma demanda w_j que é totalmente servida pela firma que localizar a facilidade mais próxima deste consumidor. Empates são quebrados em favor das facilidades da líder (orientação conservadora) e empates entre facilidades da mesma firma são quebrados arbitrariamente. Cada firma visa atender a maior demanda possível. O (r|p)-centroide consiste em decidir onde a líder deve localizar suas p facilidades sabendo que a firma seguidora reagirá localizando r facilidades.

Proposição 1: O (r|p)H-C é \sum_2^P -Difícil.

Prova: Para a prova, vamos fazer uma redução a partir do (r|p)-centroide. Seja $\langle G_1 = (I_1 \cup J_1, E_1), r_1, p_1, w_1, d_1 \rangle$ uma instância do (r|p)-centroide. A partir desta, vamos construir uma instância $\langle G = (I \cup J, E), r, p, w, c, \alpha \rangle$ para o (r|p)H-C da seguinte maneira. Para cada cliente $j_1 \in J_1$, existem dois clientes j e j' em J localizados no mesmo lugar que j_1 com fluxo $w_{jj'} = w_{j_1}$. Além disso, os demais fluxos envolvendo j e/ou j' são nulos. De maneira similar, para cada candidato a facilidade $i_1 \in I_1$, existem dois candidatos a hubs i e i' em I localizados no mesmo lugar que i_1 com custo $c_{ii'} = c_{i'i} = 0$, enquanto os demais custos envolvendo i ou i' , com outros hubs são infinitos. Além disso, os custos envolvendo um dado cliente $j \in J$ e um dado hub $i \in I$ é dado pela distancia $d_{ij}/2$. Os parâmetros r e p serão respectivamente iguais a r_1 e p_1 , enquanto $\alpha = 1$. Dessa forma ao resolvermos o (r|p)H-C, obtemos a solução do (r|p)-centroide. A Figura 2 ilustra a conversão do (r|p)-centroide ao (r|p)H-C, onde os quadrados representam possíveis locais de facilidades ou hubs e círculos representam os clientes.

Figura 2: Redução do (r|p)-centroide ao (r|p)H-C



4. Formulação Matemática

Nesta seção é apresentada a formulação minimax bilinear para o (r|p)H-C. Na formulação abaixo a líder busca minimizar a demanda atendida pela seguidora (dessa forma maximizando a demanda captura pelo próprio). Sejam $i, j \in N$, os clientes com fluxos a serem transportados entre esses dado por w_{ij} . Seja H , o conjunto de locais disponíveis para instalação de hub, onde a empresa líder deve localizar p hubs e a seguidora r hubs.

Sejam as seguintes variáveis da líder:

x_k – variável binária que indica se haverá hub da líder instalado no local k ;

y_{ijkm} - variável binária que indica se os hubs k e m são localizados pela líder para atendimento do fluxo entre os clientes i e j . (quando $k = m$, apenas 1 hub é utilizado);

Sejam as seguintes variáveis da seguidora:

s_k – variável binária que indica se haverá hub da seguidora instalado no local k ;

t_{ijkm} - variável binária que indica se os hubs k e m são localizados pela seguidora para atendimento do fluxo entre os clientes i e j . (quando $k = m$, apenas 1 hub é utilizado);

Formulação natural:

$$\min_{x,y} \sum_{i \in N} \sum_{j \in N} w_{ij} - \sum_{i \in N} \sum_{j \in N} w_{ij} * \left[\sum_{k \in H} \sum_{m \in H} t_{ijkm} * \left(\sum_{k' \in H} \sum_{m' \in H | c_{ijk'm'} \leq c_{ijkm}} y_{ijkm} \right) \right] \quad (3)$$

$$s. a. \quad \sum_{k \in H} x_k = p \quad (4)$$

$$y_{ijkm} \leq x_k \quad \forall i \in N, \forall j \in N, \forall k \in H, \forall m \in H \quad (5)$$

$$y_{ijkm} \leq x_m \quad \forall i \in N, \forall j \in N, \forall k \in H, \forall m \in H \quad (6)$$

$$\sum_{k \in H} \sum_{m \in H} y_{ijkm} = 1 \quad \forall i \in N, \forall j \in N \quad (7)$$

$$y_{ijkm} \in \{0,1\} \quad \forall i \in N, \forall j \in N, \forall k \in H, \forall m \in H \quad (8)$$

$$x_k \in \{0,1\} \quad \forall k \in H \quad (9)$$

$$\max_{s,t} \sum_{i \in N} \sum_{j \in N} w_{ij} - \sum_{i \in N} \sum_{j \in N} w_{ij} * \left[\sum_{k \in H} \sum_{m \in H} t_{ijkm} * \left(\sum_{k' \in H} \sum_{m' \in H | c_{ijk'm'} \leq c_{ijkm}} y_{ijkm} \right) \right] \quad (10)$$

$$s. a. \quad \sum_{k \in H} s_k = r \quad (11)$$

$$t_{ijkm} \leq s_k \quad \forall i \in N, \forall j \in N, \forall k \in H, \forall m \in H \quad (12)$$

$$t_{ijkm} \leq s_m \quad \forall i \in N, \forall j \in N, \forall k \in H, \forall m \in H \quad (13)$$

$$\sum_{k \in H} \sum_{m \in H} t_{ijkm} = 1 \quad \forall i \in N, \forall j \in N \quad (14)$$

$$t_{ijkm} \in \{0,1\} \quad \forall i \in N, \forall j \in N, \forall k \in H, \forall m \in H \quad (15)$$

$$s_k \in \{0,1\} \quad \forall k \in H \quad (16)$$

As funções objetivo de 1º e 2º níveis (3) e (10) são idênticas, porém uma é de maximização e a outra de minimização. Tais funções representam que a demanda total atendida pela seguidora, que é dada pela diferença entre a demanda total existente e a demanda atendida pela líder. A restrição de 1º nível (4) indica que p hubs serão instalados pela líder. As restrições de 1º nível (5) e (6) garantem a consistência entre as variáveis x e y. As restrições de 1º nível (7) garantem que para cada par de clientes apenas um par de hubs k, m da líder é o que melhor atende a esse par de clientes. A restrição de 2º nível (11) indica que r hubs serão instalados pela seguidora. As restrições de 2º nível (12) e (13) garantem a consistência entre as variáveis s e t. As restrições de 2º nível (14) garantem que para cada par de clientes apenas um par de hubs k, m da seguidora é o que melhor atende a esse par de clientes.

Utilizando a proposta de Pessoa et al. (2013) substituímos o problema de otimização do 2º nível (10) - (16) por um conjunto de desigualdades lineares, representadas por (18). Para tal nova formulação foi introduzida uma variável z que representa o fluxo capturado pela seguidora, ou seja, a função objetivo a ser minimizada. Obtemos assim a seguinte formulação de Programação Linear Inteira (PLI):

$$\min z \quad (17)$$

$$s. a. (4) - (9)$$

$$z \geq \sum_{i \in N} \sum_{j \in N} w_{ij} - \sum_{i \in N} \sum_{j \in N} w_{ij} * \left[\sum_{k \in H} \sum_{m \in H} t_{ijkm} * \left(\sum_{k' \in H} \sum_{m' \in H | c_{ijk'm'} \leq c_{ijkm}} y_{ijkm} \right) \right] \quad (18)$$

$$\forall s, t \text{ que satisfazem (11) - (16)}$$

O número de restrições (18) é dado pelo número de estratégias viáveis da seguidora. Como este número tende a ser exponencial, tais restrições devem ser geradas durante um algoritmo de *branch-and-cut*. O problema da separação consiste de, dada uma solução relaxada (\bar{x}, \bar{y}) , encontrar a estratégia da seguidora (s, t) que minimiza a captura de fluxo da líder. Tal estratégia pode ser encontrada pela resolução do PLI (10) - (16) considerando os valores constantes \bar{y} na função objetivo (10).

Além da formulação acima ainda foi proposto um pré-processamento, onde boa parte das variáveis y_{ijkm} e t_{ijkm} são eliminadas para casos onde $c_{ijkk} < c_{ijkm}$ ou $c_{ijmm} < c_{ijkm}$, tal condição reflete a opção das empresas de utilizar apenas um hub para o transporte, quando tal solução possuir custo inferior à utilização dos 2 hubs. A Tabela 1 mostra, para $\alpha = 0,6$ e $\alpha = 0,8$ respectivamente, o número de variáveis y eliminadas para cada instância CAB n , gerada pelas n primeiras cidades da CAB 25 *data*. As matrizes de custo e fluxo são geradas por submatrizes $n \times n$, obtidas do canto superior esquerdo da CAB 25.

Pode-se notar uma maior eliminação de variáveis para $\alpha = 0,8$, explicado pelo fato do desconto entre hubs ser menor, não havendo tanto incentivo para as empresas utilizarem os dois hubs ao invés de apenas um.

Tabela1: Variáveis y eliminadas com pré-processamento

Instância	# total de variáveis y	$\alpha = 0,6$		$\alpha = 0,8$	
		# variáveis y eliminadas	% de variáveis eliminadas	# variáveis y eliminadas	% de variáveis eliminadas
CAB_5	150	77	51,3%	81	54,0%
CAB_6	315	184	58,4%	192	61,0%
CAB_7	588	364	61,9%	381	64,8%
CAB_8	1008	657	65,2%	697	69,1%
CAB_9	1620	1107	68,3%	1166	72,0%
CAB_10	2475	1719	69,5%	1837	74,2%
CAB_11	3630	2585	71,2%	2754	75,9%
CAB_12	5148	3703	71,9%	3950	76,7%
CAB_13	7098	5259	74,1%	5572	78,5%
CAB_14	9555	7324	76,7%	7700	80,6%
CAB_15	12600	9878	78,4%	10340	82,1%
CAB_16	16320	13013	79,7%	13602	83,3%
CAB_17	20808	16601	79,8%	17423	83,7%
CAB_18	26163	20814	79,6%	21956	83,9%
CAB_19	32490	25797	79,4%	27327	84,1%
CAB_20	39900	31769	79,6%	33686	84,4%
CAB_21	48510	38940	80,3%	41193	84,9%
CAB_22	58443	46752	80,0%	49671	85,0%
CAB_23	69828	56271	80,6%	59893	85,8%
CAB_24	82800	67558	81,6%	71705	86,6%
CAB_25	97500	79578	81,6%	84578	86,7%

5. Resultados Computacionais

Foram realizados experimentos computacionais utilizando-se CPLEX 12.5.1 num computador CORE i7 de 3.6 GHz, com 3Gb de RAM, utilizando-se sempre um processador apenas.

Os dados utilizados foram os da instância CAB 25, baseada no tráfego aéreo de

passageiros entre 25 cidades dos Estados Unidos da América (EUA), conforme dados do Civil Aeronautics Board. Foi considerado $H = N$, ou seja, existe a possibilidade de localização de hubs em todos 25 nós da instância. O modelo de PLI para o problema da separação foi resolvido apenas para soluções relaxadas inteiras, apesar de também ser apto a separar soluções relaxadas fracionárias. A eficiência da abordagem proposta é avaliada comparando-se os tempos de execução do algoritmo implementado com os tempos relatados em Mahmutogullari (2013), que se utilizou de uma técnica de enumeração para resolver o problema. Em tal trabalho, foram utilizados valores de p e r variando entre 2 e 5 e α assumindo os valores 0,6 e 0,8. O computador utilizado foi um 4x AMD Opteron Interlagos 16C 6282SE 2.6G 16M 6400MT. Os resultados foram os expressos nas tabelas 2 e 3, onde E representa o tempo de CPU em segundos para o método de enumeração proposto por Mahmutogullari (2013) e PLI representa o tempo CPU em segundos obtido com o algoritmo proposto no presente trabalho. Para cada instância, nós marcamos em negrito o menor tempo computacional.

Tabela 2: Comparação de resultados para $\alpha = 0,6$

$p \setminus r$	2		3		4		5	
2	E	PLI	E	PLI	E	PLI	E	PLI
	1,52	119,23	12,71	82,54	70,32	474,47	320,78	5275,32
3	E	PLI	E	PLI	E	PLI	E	PLI
	5,81	165,23	19,94	159,77	88,02	112,44	557,16	159,41
4	E	PLI	E	PLI	E	PLI	E	PLI
	19,27	165,09	36,62	113,46	141,60	293,07	631,10	110,23
5	E	PLI	E	PLI	E	PLI	E	PLI
	70,35	445,09	117,40	720,33	371,14	348,54	1498,9	447,48

Tabela 3: Comparação de resultados para $\alpha = 0,8$

$p \setminus r$	2		3		4		5	
2	E	PLI	E	PLI	E	PLI	E	PLI
	1,35	31,36	11,78	70,43	100,37	98,79	535,15	83,83
3	E	PLI	E	PLI	E	PLI	E	PLI
	4,31	53,25	23,13	95,95	142,68	104,39	791,55	81,81
4	E	PLI	E	PLI	E	PLI	E	PLI
	17,96	195,97	30,70	89,74	212,61	76,00	1015,49	73,25
5	E	PLI	E	PLI	E	PLI	E	PLI
	74,25	278,51	139,05	385,77	382,09	203,00	1335,03	219,43

Observando as tabelas 2 e 3, percebemos que conforme crescem os valores de p e r , no algoritmo de enumeração os tempos crescem de forma bem mais acentuada, sendo mais vantajoso o modelo de PLI proposto. Como com $\alpha = 0,8$, mais variáveis foram eliminadas com o pré-processamento, isto trouxe resultados melhores que $\alpha = 0,6$ para nosso modelo.

A seguir, foram testadas instâncias com p e r entre 6 e 10, e entre 11 e 15, apresentados seus resultados nas tabelas 4 e 5 respectivamente. Tais resultados foram apresentados ao final deste artigo. As 100 instâncias apresentadas foram resolvidas otimamente pela primeira vez, sendo 61 resolvidas em menos de 10 minutos (600 segundos), 38 resolvidas entre 10 minutos e 1 hora (600 – 3600 segundos) e apenas 1 com duração superior a 1 hora. Apesar dos bons resultados, verificamos gaps ainda elevados na raiz, que podem ser reduzidos caso encontrássemos cortes fortalecidos para o problema. A obtenção de tais cortes pode ser tratada em trabalhos futuros.

O Gráfico 1 mostra como varia o fluxo capturado pela líder para as instâncias onde $p = r$, ou seja, ambas as empresas localizam o mesmo número de hubs. Fica evidenciado que

conforme aumenta o número de hubs instalados maior é o fluxo capturado pela líder. Essa tendência pode ser explicada pelo critério de desempate em favor da líder. O Gráfico 2 mostra como variam os tempos de CPU para resolução para as mesmas instâncias com $p = r$. Percebe-se um aumento bem suave de tempo, com o crescimento de p , diferentemente do que ocorreria no método enumerativo, que tenderia a crescer exponencialmente. No gráfico ainda se evidencia que a partir de determinados valores de p , passa a haver uma tendência de redução do tempo de resolução.

Gráfico 1: Captura de fluxo do líder por número de hubs instalados para $p = r$

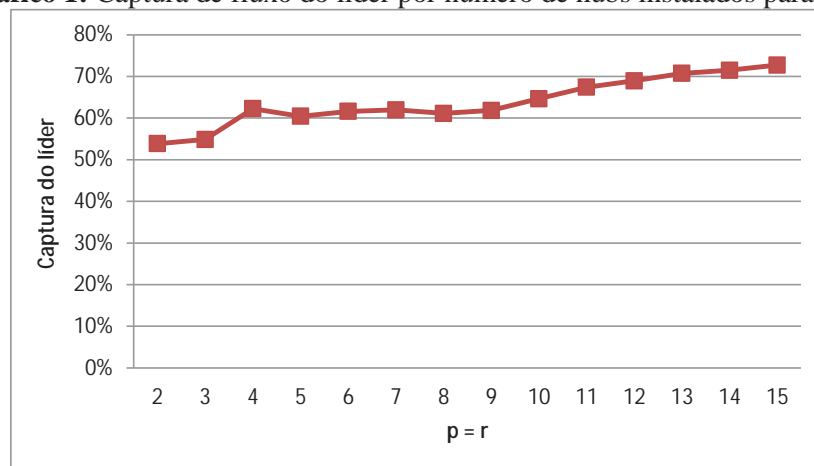
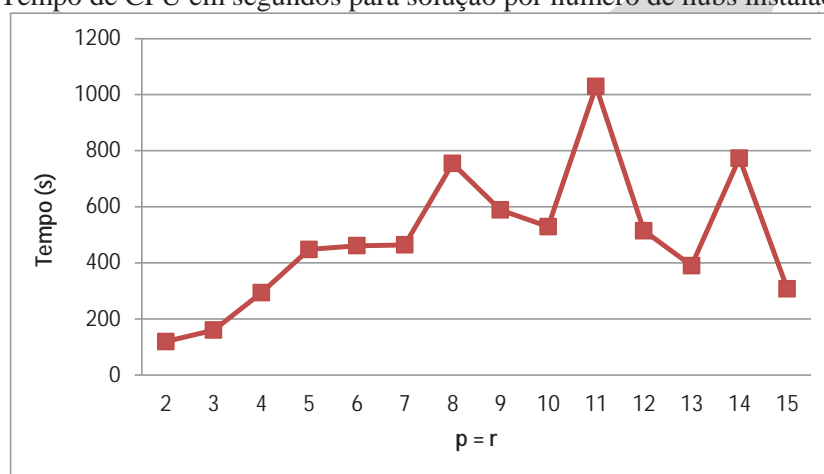


Gráfico 2: Tempo de CPU em segundos para solução por número de hubs instalados para $p = r$



6. Conclusões

Neste trabalho apresentamos o problema $(r|p)H-C$. Um algoritmo *branch-and-cut* exato foi proposto para o problema obtendo-se resultados extremamente satisfatórios para instâncias razoavelmente grandes. Para trabalhos futuros sugere-se a aplicação do algoritmo a instâncias maiores, com maior número de clientes e mais locais possíveis de instalar hub. Outra sugestão é testar o modelo também utilizando separações fracionárias, ou outros métodos buscando reduzir os gaps na raiz, o que diminuiria a árvore de *branch-and-bound* a ser percorrida.

Referências

- Alumur, S. e Kara, B. Y.** (2008), Network hub location problems: The state of the art. *European Journal of Operational Research*, v. 190, n. 1, p. 1-21.
- Campbell, J. F.** (1994). Integer programming formulations of discrete hub location problems. *European Journal of Operational Research*, v. 72, n. 2, p. 387-405.
- Hakimi, S.** (1983), On locating new facilities in a competitive environment. *European Journal of Operational Research*, v. 12, n. 1, p. 29–35.
- Israeli, E. e Wood, R. K.** (2002), Shortest-path network interdiction. *Networks*, v. 40, n. 2, p. 97–111.
- Mahmutoğulları, A. İ.** (2013), Hub location under competition. Dissertação de mestrado. Departamento de engenharia industrial. Universidade de Bilkent.
- Marianov, V., Serra, D., ReVelle, C.** (1999), Location of hubs in a competitive environment. *European Journal of Operational Research*, v. 114, n. 2, p. 363-371.
- Moore, J. T. e Bard, J. F.** (1990), The mixed integer linear bilevel programming problem. *Operations research*, v. 38, n. 5, p. 911–921
- Noltmeier, H., Spoerhase, J. and Wirth, H.** (2007), Multiple voting location and single voting location on trees. *European Journal of Operational Research*, v. 181, n. 2, p. 654–667
- O’Hanley, J. R. e Church, R. L.** (2011), Designing robust coverage networks to hedge against worst-case facility losses. *European Journal of Operational Research*, v. 209, n. 1, p. 23–36.
- O’Kelly, M. E.** (1987), A quadratic integer program for the location of interacting hub facilities. *European Journal of Operational Research*, v. 32, n. 3, p. 393-404.
- Pessoa, A. A., Poss, M., Roboredo, M. C., Aizemberg, L.** (2013), Solving bilevel combinatorial optimization as bilinear min-max optimization via a branch-and-cut algorithm. *Anais do XLV Simpósio Brasileiro de Pesquisa Operacional (SBPO)*.
- Roboredo, M. C.** (2012), A branch-and-cut algorithm for a budget constrained centroid problem. *Anais do XLIV Simpósio Brasileiro de Pesquisa Operacional (SBPO)*.
- Roboredo, M. C. e Pessoa, A. A.** (2013), A branch-and-cut algorithm for the discrete (r|p)-centroid problem. *European Journal of Operational Research*, v. 224, n. 1, p. 101-109.
- Roboredo, M. C. e Aizemberg, L.** (2014), O Problema da Máxima Cobertura com Fortificações e Interdições. *Anais do XLVI Simpósio Brasileiro de Pesquisa Operacional (SBPO)*.
- Sasaki, M.** (2005). Hub network design model in a competitive environment with flow threshold. *Journal of the Operations Research Society of Japan-Keiei Kagaku*, v. 48, n. 2, p. 158-171.
- Taskin, Z. C., Smith, J. C., Ahmed, S. e Schaefer, A. J.** (2009), Cutting plane algorithms for solving a stochastic edge-partition problem. *Discrete Optimization*, v. 6, n. 4, p. 420–435
- Von Stackelberg, H.**, *Marktform und gleichgewicht*. J. Springer, 1934.
- Wagner, B.** (2008), A note on “location of hubs in a competitive environment”. *European Journal of Operational Research*, v. 184, n. 1, p. 57-62.

Tabela 4: Resultados para p e r entre 6 e 10

p	r	α	F. O.		Raiz			Árvore de <i>branch-and-bound</i>				
			Captura seg. ($z/\sum w_{ij}$)	z	LB (raiz)	Gap (%) raiz	Tempo Raiz (s)	# Nós	# Seps	# Cortes	Tempo Sep (s)	Tempo Total (s)
6	6	0,6	0,384088	1640057	1284814	21,66%	51,11	2392	32	23	110,19	461,08
6	7	0,6	0,442178	1888103	1488583	21,16%	43,64	982	39	29	118,93	273,81
6	8	0,6	0,506203	2161488	1820484	15,78%	47,41	1087	43	31	117,72	322,55
6	9	0,6	0,556556	2376496	2013455	15,28%	32,56	354	38	29	63,36	127,32
6	10	0,6	0,600029	2562127	2166906	15,43%	30,19	697	31	22	48,51	165,44
7	6	0,6	0,330406	1410833	987518	30,00%	50,50	9619	33	25	124,62	1317,39
7	7	0,6	0,380244	1623641	1252821	22,84%	48,39	2108	42	28	135,65	463,77
7	8	0,6	0,438433	1872112	1495812	20,10%	35,58	2121	37	25	90,47	385,82
7	9	0,6	0,488832	2087312	1655341	20,70%	27,64	2780	34	27	61,79	378,66
7	10	0,6	0,533432	2277758	1871383	17,84%	18,56	2188	39	32	69,94	336,28
8	6	0,6	0,291828	1246107	802223	35,62%	67,91	4001	51	42	181,23	751,51
8	7	0,6	0,341304	1457370	974033	33,17%	34,34	5198	55	43	172,93	831,80
8	8	0,6	0,388813	1660233	1197915	27,85%	29,52	3103	74	61	207,64	754,44
8	9	0,6	0,438409	1872008	1395665	25,45%	23,77	2506	66	58	156,98	491,03
8	10	0,6	0,472468	2017439	1556368	22,85%	27,19	1262	48	39	90,59	258,79
9	6	0,6	0,264395	1128969	585772	48,11%	32,59	6642	99	84	357,23	1286,59
9	7	0,6	0,307019	1310974	836607	36,18%	44,08	3843	92	79	275,10	843,32
9	8	0,6	0,347684	1484613	933904	37,09%	27,11	27577	97	84	273,48	3239,78
9	9	0,6	0,381767	1630146	1112854	31,73%	22,77	2668	75	66	166,50	588,35
9	10	0,6	0,410427	1752526	1274532	27,27%	19,80	7617	48	40	87,18	841,79
10	6	0,6	0,229644	980579	480819	50,97%	34,44	33950	106	91	358,99	4024,23
10	7	0,6	0,263733	1126139	682527	39,39%	74,11	3909	97	84	280,04	867,71
10	8	0,6	0,297179	1268957	798472	37,08%	39,47	11610	78	71	214,70	1701,98
10	9	0,6	0,324074	1383798	950718	31,30%	31,89	11447	54	46	121,84	1439,06
10	10	0,6	0,353544	1509632	1107756	26,62%	38,70	2997	64	55	148,80	529,17
6	6	0,8	0,390406	1667033	1179048	29,27%	34,53	6358	35	24	94,92	552,10
6	7	0,8	0,439714	1877581	1537907	18,09%	31,73	2340	42	28	106,19	343,84
6	8	0,8	0,487329	2080895	1780672	14,43%	33,92	4268	33	22	79,55	458,95
6	9	0,8	0,531934	2271359	1963088	13,57%	28,91	647	35	21	63,83	133,35
6	10	0,8	0,574220	2451920	2189993	10,68%	30,39	2183	45	32	77,95	281,73
7	6	0,8	0,329224	1405788	1047408	25,49%	61,92	3210	25	20	75,30	416,34
7	7	0,8	0,383011	1635460	1242721	24,01%	49,81	10935	38	30	108,98	1104,86
7	8	0,8	0,432643	1847387	1466518	20,62%	28,31	2994	45	39	124,58	506,58
7	9	0,8	0,475462	2030225	1679181	17,29%	30,89	4409	41	34	91,46	511,52
7	10	0,8	0,511288	2183202	1878910	13,94%	36,25	2640	31	25	57,31	288,13
8	6	0,8	0,290945	1242336	825886	33,52%	61,11	8846	47	38	142,45	942,84
8	7	0,8	0,342781	1463678	989032	32,43%	52,12	3264	58	51	159,87	506,25
8	8	0,8	0,383760	1638656	1215258	25,84%	54,05	6366	60	48	168,97	766,24
8	9	0,8	0,421121	1798188	1327568	26,17%	22,00	3059	59	49	131,35	476,35
8	10	0,8	0,449020	1917315	1571166	18,05%	45,06	2310	28	23	50,87	275,76
9	6	0,8	0,258589	1104175	658056	40,40%	51,44	29157	77	66	252,31	3188,11
9	7	0,8	0,296665	1266760	830414	34,45%	57,67	12981	65	60	203,28	1441,28
9	8	0,8	0,332053	1417869	976433	31,13%	47,55	11668	64	52	149,65	1110,28
9	9	0,8	0,364881	1558043	1194527	23,33%	55,14	22590	49	37	96,34	1692,21
9	10	0,8	0,391766	1672843	1305045	21,99%	48,28	3395	56	44	122,08	429,95
10	6	0,8	0,222037	948099	552011	41,78%	71,94	3201	83	74	274,00	680,58
10	7	0,8	0,261204	1115341	646005	42,08%	43,89	16070	90	75	258,14	1695,04
10	8	0,8	0,288762	1233014	814209	33,97%	51,27	5947	68	55	170,17	912,81
10	9	0,8	0,316649	1352093	939097	30,54%	42,34	3226	56	50	119,47	529,43
10	10	0,8	0,342030	1460468	1089113	25,43%	42,31	3759	50	39	102,27	473,81

Tabela 5: Resultados para p e r entre 11 e 15

p	r	α	F. O.		Raiz			Árvore de <i>branch-and-bound</i>				
			Captura seg. ($z/\sum w_{ij}$)	z	LB (raiz)	Gap (%) raiz	Tempo Raiz (s)	# Nós	# Seps	# Cortes	Tempo Sep (s)	Tempo Total (s)
11	11	0,6	0,325943	1391779	1057672	24,01%	42,09	7275	60	47	123,55	1029,10
11	12	0,6	0,351605	1501356	1166805	22,28%	26,44	3194	58	46	99,92	469,38
11	13	0,6	0,375102	1601687	1290388	19,44%	38,08	2820	36	29	51,52	321,63
11	14	0,6	0,395527	1688903	1398580	17,19%	34,56	3760	53	43	80,52	498,46
11	15	0,6	0,412512	1761429	1486887	15,59%	34,81	960	49	38	57,15	191,59
12	11	0,6	0,285544	1219273	849044	30,36%	24,48	2053	52	40	99,53	354,32
12	12	0,6	0,310771	1326994	968247	27,03%	18,75	3883	60	49	97,16	514,62
12	13	0,6	0,333902	1425764	1066759	25,18%	17,05	7117	56	46	86,44	745,50
12	14	0,6	0,350392	1496174	1160738	22,42%	20,45	2586	47	38	60,45	335,75
12	15	0,6	0,367303	1568387	1270007	19,02%	22,19	2997	42	36	51,85	329,22
13	11	0,6	0,247231	1055679	771402	26,93%	47,73	3539	43	33	74,50	533,66
13	12	0,6	0,271785	1160524	855964	26,24%	38,91	15195	31	27	43,90	1273,02
13	13	0,6	0,292786	1250195	972236	22,23%	41,34	2824	42	36	58,00	389,95
13	14	0,6	0,311558	1330353	1061542	20,21%	27,33	1596	59	50	71,86	292,38
13	15	0,6	0,327665	1399129	1140965	18,45%	34,25	1225	31	23	35,52	203,08
14	11	0,6	0,229204	978700	680388	30,48%	39,33	6087	59	53	103,23	756,99
14	12	0,6	0,250204	1068371	743176	30,44%	40,33	9778	57	49	90,83	1012,66
14	13	0,6	0,268976	1148529	832742	27,49%	36,28	5890	61	53	79,66	653,98
14	14	0,6	0,285083	1217305	912718	25,02%	31,73	7585	55	47	59,84	773,48
14	15	0,6	0,300059	1281252	1016657	20,65%	27,31	2472	51	45	52,76	310,43
15	11	0,6	0,208393	889838	588112	33,91%	52,39	5468	88	82	145,65	765,59
15	12	0,6	0,229516	980033	648651	33,81%	32,91	12606	111	105	154,73	1638,52
15	13	0,6	0,245623	1048809	747203	28,76%	30,09	7630	86	80	102,53	866,76
15	14	0,6	0,259400	1107639	816114	26,32%	32,02	22373	72	65	78,40	1884,74
15	15	0,6	0,272787	1164800	895401	23,13%	30,56	2345	67	61	66,80	307,41
11	11	0,8	0,326841	1395610	1031154	26,11%	41,39	1625	62	55	113,48	300,95
11	12	0,8	0,347684	1484611	1154226	22,25%	35,22	1259	55	44	86,21	230,06
11	13	0,8	0,366009	1562860	1274734	18,44%	41,20	6845	51	41	81,70	620,45
11	14	0,8	0,383666	1638257	1400867	14,49%	35,00	2272	38	33	56,83	244,31
11	15	0,8	0,399050	1703943	1495961	12,21%	31,25	1977	37	30	41,45	187,24
12	11	0,8	0,285706	1219967	844526	30,77%	28,69	5398	63	56	148,67	701,30
12	12	0,8	0,302997	1293797	1000864	22,64%	39,38	3578	54	41	93,67	440,73
12	13	0,8	0,321765	1373937	1057598	23,02%	18,53	1217	48	41	73,12	213,00
12	14	0,8	0,337872	1442713	1169044	18,97%	24,36	795	37	31	47,10	135,92
12	15	0,8	0,355062	1516115	1309085	13,66%	36,41	2985	38	29	46,88	254,33
13	11	0,8	0,247469	1056694	722078	31,67%	31,31	6180	57	43	115,56	636,68
13	12	0,8	0,264895	1131103	851896	24,68%	30,66	4039	46	39	85,31	454,21
13	13	0,8	0,283170	1209138	936973	22,51%	33,83	2638	37	30	51,08	305,01
13	14	0,8	0,300747	1284189	1022610	20,37%	31,11	973	29	24	35,79	157,91
13	15	0,8	0,320298	1367673	1125687	17,69%	26,13	3043	42	35	45,94	299,33
14	11	0,8	0,219657	937934	661919	29,43%	41,38	4083	49	43	84,55	501,94
14	12	0,8	0,238627	1018936	720307	29,31%	37,63	9014	42	35	70,50	735,40
14	13	0,8	0,256784	1096469	816765	25,51%	26,83	1782	58	49	71,30	271,45
14	14	0,8	0,275556	1176627	896770	23,78%	24,42	2511	41	37	46,45	268,24
14	15	0,8	0,291663	1245403	1006428	19,19%	30,53	1808	53	45	49,23	235,17
15	11	0,8	0,200203	854867	539484	36,89%	39,84	12944	70	59	108,62	1085,44
15	12	0,8	0,218497	932984	621193	33,42%	19,52	7049	63	54	79,11	579,68
15	13	0,8	0,235082	1003801	720601	28,21%	32,64	12946	63	56	78,23	1025,47
15	14	0,8	0,250166	1068208	787164	26,31%	32,00	3887	75	66	73,17	457,74
15	15	0,8	0,262996	1122995	867525	22,75%	24,50	5572	51	44	40,63	520,36