LII Simpósio Brasileiro de Pesquisa Operacional
João Pessoa - PB, 3 a 5 de novembro de 2020

UM MODELO DE PROGRAMAÇÃO LINEAR INTEIRA MISTA PARA UM PROBLEMA DE LOCALIZAÇÃO DE HUBS COM COBERTURA MÁXIMA

Patrick Doglio
Pós-Graduação em Engenharia de Produção - UFF
R. Passo da Pátria, nº 156, Bloco D, sala 309, São Domingos, Niterói-RJ
patrickdoglio@id.uff.br

Marcos Roboredo, Artur Pessoa
Departamento de Engenharia de Produção - UFF
R. Passo da Pátria, nº 156, Bloco D, sala 306, São Domingos, Niterói-RJ
mcroboredo@id.uff.br, artur@producao.uff.br

RESUMO

Hubs são facilidades que consolidam fluxos em sistemas de transporte, gerando redução de custos através da economia de escala. O problema de localização de hubs com cobertura máxima e alocação simples visa decidir onde localizar p hubs e como alocar cada ponto não hub a um único hub localizado, maximizando o total de fluxos cobertos. Cada fluxo é dito estar coberto se o custo de atendimento desse fluxo não excede um fluxo máximo preestabelecido. Este trabalho propõe um conjunto de restrições que são incorporadas à melhor formulação existente para este problema. O número de tais restrições cresce exponencialmente com o tamanho da instância. Portanto, este trabalho propõe a inclusão das restrições sob demanda durante o clássico algoritmo de *branch-and-bound*. Experimentos computacionais são reportados de modo a comprovar o ganho da inclusão das restrições propostas, onde o método proposto apresenta uma melhor performance computacional para 45 das 54 instâncias comparadas.

PALAVRAS CHAVE. Programação Inteira, Localização de Hubs, Cobertura Máxima.

Tópicos: Otimização Combinatória, Logística e Transportes

ABSTRACT

Hubs are facilities that consolidate flows in transportation systems, generating cost savings through economies of scale. The single allocation p-hub maximal covering problem aims to decide where to locate p hubs and how to allocate each non-hub point to a single located hub, maximizing the total flows covered. Each flow is said to be covered if the cost of serving this flow does not exceed a pre-established maximum flow. This work proposes a set of restrictions that incorporate the best existing formulation for this problem. The number of such restrictions grows exponentially with the size of the instance. Therefore, this work proposes an inclusion of restrictions on demand during the classic branch-and-bound algorithm. Computational experiments are reported in order to prove the gain of the inclusion of the proposed restrictions, where the proposed method presents a better computational performance for 45 of the 54 compared instances.

KEYWORDS. Integer Programming, Hubs Location, Maximal Covering.

Topics: Combinatorial Optimization, Logistics and Transport

Galoá { Este trabalho foi publicado utilizando o **Galoá** proceedings

1. Introdução

Hubs são facilidades especiais designadas para atuarem como pontos de troca em sistemas de transporte, onde é necessário transferir fluxos entre diversos pares origem e destino (O/D). Em sistemas de distribuição *many-to-many*, ao invés de pares O/D serem ligados diretamente, os fluxos são consolidados através da utilização de hubs, o que implica na redução do total de rotas necessárias para o transporte e do custo de transporte devido à exploração de economias de escala. Esse tipo de modelo de sistema de distribuição é amplamente aplicado na aviação [Button, 2002], sistemas logísticos [Ishfaq e Sox, 2011] e redes de telecomunicações [Kim e O'Kelly, 2009].

O problema de localização de hubs (PLH) foi proposto primeiramente por O'Kelly [1986]. Em um PLH, duas decisões são tomadas com a finalidade de reduzir o custo total de operação do sistema: a localização de um número finito de hubs e a alocação de pontos não hub a hubs localizados. De acordo com o tipo de alocação considerada, cada PLH pode ser classificado em alocação simples (AS-PLH), alocação múltipla (AM-PLH) ou alocação r (r-PLH). Nos AS-PLH, cada ponto não hub deve ser atribuído a um único hub enquanto nos AM-PLH, cada ponto não hub pode ser alocado a quantos hubs forem necessários. Já no caso da alocação r, cada ponto não hub deve ser atribuído a no máximo r de dentre os hubs localizados, representando uma generalização de alocação simples ($r = 1$) e alocação múltipla ($r = +\infty$). Cada PLH pode ser classificado ainda como discreto ou contínuo. Nos PLH discretos, o conjunto de potenciais hubs é discreto enquanto nos PLH contínuos, os hubs podem ser alocados em qualquer ponto do espaço de decisão.

Campbell [1994] classificou os PLHs em quatro categorias, de acordo com seus objetivos, são elas: p-hub mediana, localização de hubs não-capacitados, p-hub centro e problemas de cobertura de hubs. Este trabalho foca nesta última categoria. Para mais detalhes sobre as outras, ver Campbell [1994]. De acordo com Farahani et al. [2013], os problemas de cobertura de hubs podem ser divididos em dois grupos: o problema da cobertura da localização de um conjunto de hubs (PCLCH) e o problema da cobertura máxima de p-hubs (PCMpH). No PCLCH o objetivo consiste em minimizar o número de hubs a serem localizados satisfazendo restrições de serviço para todos os pares origem-destino. Já no PCMpH (foco deste trabalho), o objetivo é maximizar os fluxos cobertos de pares origem-destino localizando um predeterminado número de hubs.

Neste artigo é estudado a versão discreta do problema de cobertura máxima de p-hubs com alocação simples (PCMpHAS). O PCMpHAS foi proposto por Campbell [1994] e visa decidir onde localizar p hubs e a maneira de alocar cada ponto não hub a um único hub localizado. Após estas decisões, é possível se identificar, para cada par O/D, qual par de hubs localizados ou qual hub está associado este par O/D, gerando um custo de atendimento. Se este custo não exceder um custo máximo previamente definido, o fluxo deste par O/D é dito estar coberto. O objetivo do problema é maximizar a soma total dos fluxos que são cobertos.

Para o PCMpHAS, Hwang e Lee [2012] propuseram um novo modelo com $O(n^4)$ variáveis e restrições. Peker e Kara [2015] apresentaram novas formulações tanto para para o PCMpHAS quanto para variante de alocação múltipla. A formulação proposta pelos autores para PCMpHAS possui $O(n^2)$ variáveis. Os autores mostraram ainda que a formulação proposta para PCMpHAS possui uma performance computacional mais satisfatória do que as formulações existentes até aquele momento. Janković et al. [2017] apresentaram heurísticas e uma nova formulação de programação inteira mista para os casos de alocação simples e alocação múltipla. Apesar de mais recente, a formulação proposta pelos autores para o PCMpHAS não apresentou tempos computacionais menores quando comparada a formulação proposta por Peker e Kara [2015].

Além de métodos exatos, esse problema também apresenta heurísticas e outros variantes. Silva e Cunha [2017] propuseram uma heurística de busca tabu para o problema de cobertura

máxima de p-hubs não capacitados com alocação simples. Janković et al. [2017] solucionaram as formulações propostas através de dois métodos heurísticos baseados em pesquisa de vizinhança. Janković e Stanimirović [2017] analisaram através de alocação r a versão do problema de maximização da cobertura de p-hubs não capacitado com cobertura binária, e utilizaram uma heurística como método de solução para o problema.

Este trabalho tem como objetivo melhorar a melhor formulação exata existente para o PCMpHAS, proposta por Peker e Kara [2015], através da proposição de restrições fortalecidas baseadas no clássico Teorema do Fluxo máximo e Corte Mínimo. A quantidade de tais restrições cresce exponencialmente com o tamanho da instância e portanto este trabalho também propõe uma estratégia de inclusão das restrições sob demanda até um determinado nível da árvore de *branch-and-bound*. Para comprovar o ganho da inclusão das restrições propostas, diversos experimentos computacionais são apresentados utilizando 54 instâncias do tipo AP [Ernst e Krishnamoorthy, 1996] com até 1275 pares O/D, 50 potenciais hub e localizando 10 hubs. Os resultados mostram que o uso das instâncias propostas proporciona um menor tempo computacional em 45 das 54 instâncias comparadas, especialmente nas maiores.

O presente trabalho é dividido da seguinte maneira. Na Seção 2 o problema é formalmente descrito e é apresentada a melhor formulação da literatura. Na Seção 3 são apresentadas as contribuições deste trabalho. Na Seção 4 são apresentados e discutidos diversos experimentos computacionais. Por fim, a Seção 6 sumariza as conclusões deste trabalho, bem como apresenta propostas para possíveis trabalhos futuros.

2. Definição do problema e principal formulação existente

Esta seção apresenta uma definição formal e a principal formulação existente para o PCMpHAS. De agora em diante, este problema é referido apenas como PCMpH.

2.1. Definição formal do $PCMpH$

Seja $G = (N, A)$ um grafo completo, onde N e A denotam respectivamente o conjunto de nós e arcos. Seja ainda um conjunto H ($H \subseteq N$) representando os potenciais hubs. Cada arco $(i, j) \in A$ representa um par O/D com um fluxo associado denotado por w_{ij}. O problema possui duas decisões a serem tomadas. A primeira consiste em escolher p hubs em H para serem localizados. A segunda decisão é alocar cada ponto não hub a um único hub localizado (alocação simples). Suponha por exemplo que o ponto i seja atribuído a um hub localizado k e o ponto j seja atribuído ao hub localizado m. Neste caso, o par O/D (i, j) é associado ao par de hubs (k, m), o que gera o custo c_{ij}^{km}, que é calculado de acordo com a Equação (1).

$$c_{ij}^{km} = d_{ik} + \alpha d_{km} + d_{mj} \tag{1}$$

onde d representa a matriz de distância ou tempo entre os nós do grafo e $\alpha \in [0, 1]$ é o fator de desconto para o uso do transporte entre hubs. Após o cálculo do custo de cada par O/D (i, j), é verificado se este não excede um raio de cobertura pré-definido β_{ij}. Caso não exceda, o fluxo w_{ij} deste par O/D é dito estar coberto. As decisões de localização e alocação são tomadas de modo a maximizar a soma dos fluxos cobertos.

2.2. Formulação do $PCMpH$

A formulação com melhor performance computacional para o PCMpH foi proposta por Peker e Kara [2015] e é apresentada a seguir. Esta formulação utiliza a notação descrita na Seção 2.1 e também as constantes a_{ij}^{km}, definidas para cada $i, j \in N$ e cada $k, m \in H$, que são calculadas da seguinte maneira:

$$a_{ij}^{km} = \begin{cases} 1 & \text{se } c_{ij}^{km} \leq \beta_{ij} \\ 0 & \text{c.c} \end{cases}$$

A formulação faz uso ainda de dois tipos de variável de decisão:
$Z_{ij} \rightarrow$ Variável declarada como contínua mas que na prática assume valor 1 ou valor 0, indicando se fluxo w_{ij} está ou não coberto respectivamente.
$X_{ik} \rightarrow$ Variável binária indicando que o ponto i foi alocado a hub loclizado k ($X_{ik} = 1$) ou não ($X_{ik} = 0$). Quando $i = k$, a variável indica também se um hub é localizado em k ($X_{kk} = 1$) ou não ($X_{kk} = 0$).

A formulação completa é a que segue:

$$\max \sum_{i \in N} \sum_{j \in N} w_{ij} Z_{ij} \tag{2}$$

$$s.a. \sum_{k \in H} X_{ik} = 1 \qquad \forall i \in N \tag{3}$$

$$\sum_{k \in H} X_{kk} = p \tag{4}$$

$$Z_{ij} \leq \sum_{k \in H} a_{ij}^{km} X_{ik} + 1 - X_{jm} \qquad \forall i, j \in N, m \in H \tag{5}$$

$$Z_{ij} \leq \sum_{k \in H} (a_{ij}^{km} - 1) X_{ik} + 1 \qquad \forall i, j \in N, m \in H \tag{6}$$

$$X_{ik} \leq X_{kk}, \qquad \forall i \in N, k \in H \tag{7}$$

$$0 \leq Z_{ij} \leq 1, \qquad \forall i, j \in N \tag{8}$$

$$X_{ik} \in \{0, 1\}, \qquad \forall i \in N, k \in H \tag{9}$$

O propósito da função objetivo (2) é maximizar o fluxo total coberto considerando todos os pares O/D. As restrições (3) garantem que cada ponto não hub vai ser alocado a um único hub localizado (alocação simples). A restrição (4) garante a localização de apenas p hubs. As restrições (5) e (6) calculam o a fração do fluxo Z_{ij}. De acordo com tais restrições, tal fluxo vai assumir valor 1 quando existirem k e m tais que $X_{ik} = X_{jm} = 1$ e $a_{ij}^{km} = 1$. Caso contrário, este fluxo vai ser zero. Apenas as restrições (5) são suficientes para determinar o valor de Z_{ij}. No entanto, Peker e Kara [2015] mostraram que uso das restrições (6) fortalece a formulação. As restrições (7) garantem que a localização de um hub em um nó k possibilita a atribuição de um nó i ao hub. As restrições (8) correspondem a não negatividade das variáveis de cobertura Z enquanto que as restrições (9) garantem que as variáveis X são binárias. É válido ressaltar que a formulação original proposta por Peker e Kara [2015] considera a constante $\lambda_{ij} = max_{k,m \in H} a_{ij}^{km}$, que está implícita nas restrições (5), (6) e (8). Neste artigo, essa constante é considerada com o valor igual a 1 pelo fato do mesmo não abordar o caso da cobertura parcial, sendo considerado apenas o caso da cobertura binária.

3. Contribuições deste trabalho

Este trabalho apresenta um conjunto de restrições fortalecidas que podem ser incorporadas a formulação (2) - (9) proposta por Peker e Kara [2015]. O número de tais restrições é exponencial. Assim, este trabalho também propõe uma estratégia de inclusão de restrições sob demanda até um determinado nível da árvore de *branch-and-bound*. A Seção 3.1 apresenta as restrições propostas enquanto que a Seção 3.2 descreve a estratégia de inclusão das restrições.

3.1. Restrições fortalecidas

Este trabalho propõe um conjunto de cortes fortalecidos para a formulação (2) - (9). As desigualdades propostas partem do princípio que a variável Z_{ij} pode ser vista como o fluxo máximo que vai de i até j passando por pares de hub que cobrem (i, j), tomando a capacidade dos arcos entre um ponto não hub e um ponto hub como sendo os valores assumidos pelas variáveis X e a capacidade de arcos entre hubs como unitária. A Figura 1 ilustra esta representação em fluxos para um exemplo onde um dador par O/D (i, j) é coberto apenas pelos pares de hub $(k_1, m_3), (k_2, m_1), (k_2, m_2)$ e (k_3, m_3).

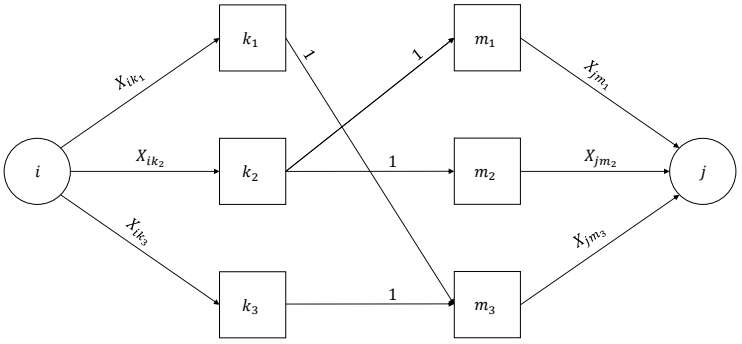


Figura 1: Representação de Z_{ij} como fluxo de i até j.

Assim, este trabalho propõe adaptar o clássico Teorema do Fluxo Máximo e Corte Mínimo [Dantzig e Fulkerson, 1964] para encontrar um conjunto de desigualdades válidas para o PCMpH. O Teorema do Fluxo Máximo e Corte Mínimo possui o seguinte enunciado adaptado pelos autores deste trabalho:

Teorema 1 (Fluxo Máximo - Corte Mínimo): Seja $G = (V, A)$ um grafo onde V é um conjunto de vértices e A é um conjunto de arcos direcionados. Para cada $a \in A$, c_a representa o maior fluxo que pode passar por este arco. Sejam s, t dois vértices de V e $f(s, t)$ o fluxo máximo que vai de s até t respeitando as capacidades dos arcos. Seja $S \subseteq V$ tal que $s \in S$ e $t \notin S$, tem-se então que:

$$f(s,t) \leq \sum_{a \in S \times V \setminus S} c_a$$

Este trabalho propõe adaptar o Teorema do Fluxo Máximo e Corte Mínimo para o PCMpH da seguinte forma. Considere o grafo para um dado par O/D (i, j), cujo conjunto de vértices V_{ij} é composto pelo i, pelo j e por todos os vértices que pertencem a algum par de hubs que cobre (i, j). O conjunto de arcos A_{ij} é formado da seguinte forma. Existe um arco de i para cada outro vértice que está associado ao primeiro hub de um par de hubs que cobre (i, j). De maneira similar, existe um arco chegando em j para cada vértice que está associado ao segundo hub de um par de hubs que cobre (i, j). As capacidades dos arcos prévios são dadas pelos valores assumidos pelas variáveis X associadas. Existe ainda um arco com capacidade unitária para cada par de hubs que cobre (i, j) conectando o primeiro ao segundo componente deste par. O fluxo máximo de i até j que respeita as

capacidades dos arcos é dado pelo valor da variável Z_{ij}. Seja S_{ij} o conjunto composto de todos os subconjuntos $S \subseteq V_{ij}$ satisfazendo três propriedades: $i \in S$, $j \notin S$ e para todo par (k, m) cobrindo (i, j) tem-se $k \in S$ $m \in S$ ou $k \notin S$ e $m \notin S$. Pelo Teorema do Fluxo Máximo e Corte Mínimo, as seguintes desigualdades são válidas para o PCMpH:

$$Z_{ij} \leq \sum_{h \in S \wedge (h,j) \in A_{ij}} X_{jh} + \sum_{h \notin S \wedge (i,h) \in A_{ij}} X_{ih}, \forall i, j \in N, \forall S \in S_{ij}. \quad (10)$$

Para ilustrar a desigualdade, considere novamente o exemplo da Figura 1. Suponha que durante a resolução de uma relaxação da formulação, a solução fracionária presente na Figura 2 foi encontrada. De acordo com as restrições (5), tem-se:

$Z_{ij} \leq X_{ik_2} + 1 - X_{jm_1}$, para $m = m_1$
$Z_{ij} \leq X_{ik_2} + 1 - X_{jm_2}$, para $m = m_2$
$Z_{ij} \leq X_{ik_1} + X_{ik_3} + 1 - X_{jm_2}$, para $m = m_3$

Substituindo os valores das variáveis fracionárias, tem-se:

$Z_{ij} \leq 1,3$, para $m = m_1$
$Z_{ij} \leq 1,3$, para $m = m_2$
$Z_{ij} \leq 1$, para $m = m_3$

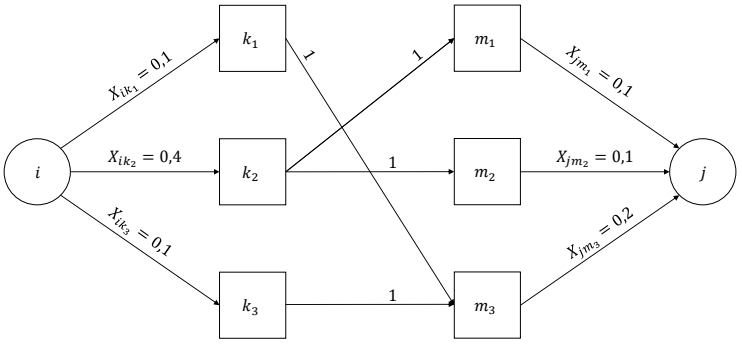


Figura 2: Hubs k e m servindo um par O/D (i, j) e as possibilidades de rotas

De maneira similar, de acordo com as restrições (6), tem-se:

$Z_{ij} \leq -X_{ik_1} - X_{ik_3} + 1$, para $m = m_1$
$Z_{ij} \leq -X_{ik_1} - X_{ik_3} + 1$, para $m = m_2$
$Z_{ij} \leq -X_{ik_2} + 1$, para $m = m_3$

Substituindo os valores das variáveis fracionárias, tem-se:

$Z_{ij} \leq 0,8$, para $m = m_1$
$Z_{ij} \leq 0,8$, para $m = m_2$
$Z_{ij} \leq 0,6$, para $m = m_3$

Como o PCMpH é um problema de maximização, então Z_{ij} assume valor $0,6$ ($Z_{ij} = 0,6$). Agora, considere o conjunto $S = \{i, k_2, h_1, h_2\}$. A Figura 3 destaca este conjunto.

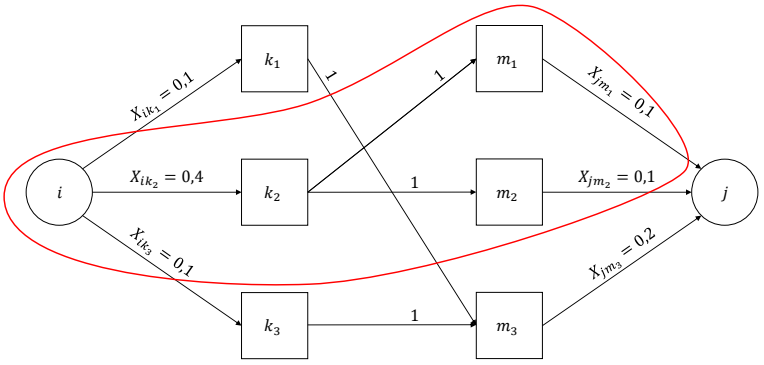


Figura 3: Exemplo de conjunto S usado para o corte.

De acordo com (10), tem-se:

$$Z_{ij} \leq X_{jh_1} + X_{jh_2} + X_{ik_1} + X_{ik_3}.$$

Substituindo os valores da solução, tem-se $Z_{ij} \leq 0,4$ e, portanto, se este corte for inserido, a variável Z_{ij} não poderia assumir valor $0,6$ que é o que acontece quando se utiliza apenas as restrições da formulação original.

3.2. Algoritmo proposto

O número de desigualdades em (10) é exponencial, o que torna inviável a geração das mesmas antes da resolução do modelo. Assim, este trabalho propõe resolver a formulação (2) - (9) através do resolvedor CPLEX com o uso de um *callback* de cortes visando incluir os cortes (10) sob demanda durante a execução do algoritmo. O Algoritmo 1 mostra os passos do nosso método, que é chamado de B&C por se tratar de um algoritmo do tipo *branch-and-cut*. Este algoritmo se inicia na linha 12, onde a formulação (2) - (9) é resolvida com a inclusão de um *callback* de cortes chamado SEPARAÇÃO(). Este *callback* é executado após a resolução de cada relaxação na árvore de *branch-and-bound*, recebendo o valor das variáveis X e Z, simbolizados com uma barra em cima das variáveis. Quando o *callback* é executado, é verificado se a diferença percentual relativa entre o valor relaxado da árvore e a melhor solução inteira (*gap*) é maior do que ϵ (linha 2), onde ϵ é um valor entre 0 e 1 pré-definido. Se sim, para todo par todo par O/D (i, j), é executada a função CORTEMINIMO() (linha 5) cujo objetivo é resolver o clássico problema do Corte Mínimo, encontrando o conjunto S que minimiza o lado direito de (10). Se o conjunto S encontrado viola (10) (linha 6), então insere o corte (linha 7). O valor ϵ utilizado é calibrado de modo a balancear o tradeoff entre o custo computacional de se incluir cortes e o ganho obtido na relaxação.

4. Experimentos Computacionais

Todos os experimentos foram realizados em um computador com processador Intel Core i5-7200U 2.70GHz CPU e 8 GB de memória RAM tendo o Windows 10 como sistema operacional. A linguagem de programação utilizada foi Julia. Para uma comparação justa com a formulação

Algoritmo 1 B&C

procedimento SEPARAÇÃO($\bar{X}, \bar{Z}$)
se $gap > \epsilon$ **então**
  para todo par O/D (i, j) **faça**
    se $\bar{X}$ é fracionário **então**
      $S \leftarrow$ CORTEMINIMO()
      se $\sum_{h \in S \wedge (h,j) \in A_{ij}} \bar{X}_{jh} + \sum_{h \notin S \wedge (i,h) \in A_{ij}} \bar{X}_{ih} < \bar{Z}_{ij}$ **então**
        Insere a restrição do tipo (10) associada (i, j) e a S
      fim se
    fim se
  fim para
fim se
Resolve a Formulação (2) - (9) tomando SEPARAÇÃO() como um callback de cortes.
Retorna X e Z ótimos

proposta por Peker e Kara [2015], todos os tempos apresentados aqui são todos obtidos na mesma máquina. A resolução das formulações bem como todo o gerenciamento da árovre de *branch-and-bound* foram feitos pelo CPLEX 12.8. As resoluções dos problemas de corte mínimo são feitas através da biblioteca *SparseMaxFlowMinCut* [Pessoa, 2019].

4.1. Geração das instâncias

Nesta seção são descritas as instâncias usadas neste trabalho. Os testes foram realizados utilizando uma base de dados muito comum em problemas envolvendo localização de hubs, denominada AP (Australian Post) proposto por Ernst e Krishnamoorthy [1996]. O conjunto AP é baseado no sistema de entregas postais da cidade de Sydney, na Austrália. O conjunto original possui 200 pontos e nesta base são fornecidos o fluxo e a distância entre todos os possíveis pares de de pontos. Porém, neste trabalho, visto que o conjunto com 200 pontos ficaria intratável sob o ponto de vista computacional, as instâncias AP utilizadas são conjuntos reduzidos de 40 e 50 pontos, gerados de acordo com o esquema de agregação de pontos propostos pelos mesmos autores.

Foram geradas 27 instâncias para cada conjunto de pontos (AP40 ou AP50) variando $\alpha \in \{0,6; 0,8; 0,9\}$ e $p \in \{2, 3, 4, 5, 6, 7, 8, 9, 10\}$. Além disso, em cada instância, foi utilizado mesmo raio de cobertura β_{ij} para todos os pares O/D, mas podendo ser diferentes para cada instância. Tais valores são explicitados nas tabelas de resultados. A base dados AP possui uma matriz de distâncias simétrica. Assim, para todo $i, j \in N$ com $i \neq j$, os pares O/D (i, j) e (j, i) são agrupados em único par O/D com fluxo $w_{ij} + w_{ji}$. Além disso, os fluxos foram normalizados, onde cada fluxo foi dividido pela soma total de todos eles. As instâncias AP40 e AP50 possuem respectivamente 820 e 1275 pares O/D e respectivamente 40 e 50 candidatos a hub. É válido ressaltar que durante a pesquisa também foram testadas instâncias mais fáceis do tipo TR [Tan e Kara, 2007]. No entanto, as estatísticas associadas a este tipo de instância foram omitidas por falta de espaço e farão parte de uma futura versão estendida do presente trabalho.

4.2. Resultados

Nesta seção são apresentados os resultados computacionais deste trabalho. Para todos os testes, o valor de ϵ utilizado foi $\epsilon = 0,07$. As Tabelas 1 e 2 mostram para as instâncias AP40 e AP50 respectivamente uma comparação entre o método proposto e a formulação proposta por Peker e Kara [2015]. Tais tabelas usam o seguinte cabeçalho. As colunas *opt* mostram o valor ótimo da função objetivo para cada instância. As colunas *Tempo(s)* mostram uma comparação entre os tempos de

CPU em segundos demandados pelos métodos. O menor dos tempos é marcado em negrito para cada instância. As colunas *Gap raiz(%)* mostram uma comparação da diferença percentual relativa entre o valor da relaxação no nó raiz e o valor da solução ótima.

Observando as Tabelas 1 e 2, pode se observar que o uso das restrições propostas resultam em um menor tempo computacional para 45 das 54 instâncias testadas. Além disso, tais restrições demonstram uma melhor performance para todas as instâncias com $p \geq 3$ e $\alpha \leq 0,8$. É válido ressaltar que o uso das restrições permite a obtenção da solução ótima de todas as instâncias testadas, utilizando no máximo $1825,16s$. Já a formulação proposta por Peker e Kara [2015] não comprovou a otimalidade em 8 instâncias considerando o tempo limite de $3600s$. Além disso, as restrições (10) reduzem significativamente o *Gap* Raiz em todas as instâncias comparadas.

A Tabela 3 apresenta resultados detalhados para as instâncias AP50 com $\alpha = 0,8$. Os seguintes novos cabeçalhos são utilizados nas colunas desta tabela. *#nós* indica o número de nós da árvore de *branch-and-bound*. *#CM* indica o número de vezes que a função CORTEMINIMO (linha 5 do Algoritmo 1) foi utilizada. *#cortes* indica o número de cortes efetivamente gerados (linha 7 do Algoritmo 1). *Tempo CM* indica o tempo total de CPU em segundos gasto pela função CORTEMINIMO(). *Tempo total* indica o tempo total de CPU em segundos gasto no algoritmo.

Fazendo uma análise da Tabela 3, pode-se notar um baixo número de nós. Além disso, nota-se também que o número de vezes que a função CORTEMINIMO() foi executada foi alto perante ao baixo número de cortes gerados. No entanto, isto não acarreta um aumento significativo no tempo computacional, uma vez que o tempo total gasto nessa função é pequeno perante ao tempo computacional total. É válido ressaltar que foram testados outros valores para ϵ diferentes de $\epsilon = 0,07$, mas que não geraram os mesmos ganhos computacionais.

5. Conclusões

Este trabalhou apresentou um conjunto de restrições fortalecidas para a melhor formulação exata existente para o Problema da Cobertura Máxima por p hubs. As restrições aqui propostas são baseadas no clássico Teorema do Fluxo Máximo e Corte Mínimo. Como o número de tais restrições cresce exponencialmente com o tamanho da instância, este trabalho também propõe uma estratégia de inclusão de cortes sob demanda durante o algoritmo de *branch-and-bound*.

Para comprovar o ganho da inclusão das restrições propostas, diversos experimentos computacionais foram conduzidos sobre 54 instâncias de diversos tamanhos geradas com base no conjunto de pontos hub AP. Os experimentos mostraram que o método proposto apresentou um menor tempo computacional para 45 das 54 instâncias comparadas, sendo 4,14 vezes mais rápido nas instâncias com otimalidade comprovada pelos dois métodos. Além disso, as restrições propostas diminuíram consideravelmente o *gap* do nó raiz, implicando em redução média de 15,90% para as instâncias AP40 e de 21,11% para as instâncias AP50.

Como trabalhos futuros, pretende-se estender os cortes propostos para a variante do problema onde é permitida a cobertura parcial de pares O/D e para a variante com alocação múltipla, ambas discutidas em Peker e Kara [2015].

Referências

Button, K. (2002). Debunking some common myths about airport hubs. *Journal of Air Transport Management*, 8(3):177 – 188. ISSN 0969-6997. Developments in Air Transport Economics.

Campbell, J. F. (1994). Integer programming formulations of discrete hub location problems. *European Journal of Operational Research*, 72(2):387–405.

Dantzig, G. B. e Fulkerson, D. R. (1964). On the max flow min cut theorem of networks. *The RAND Corporation*.

Tabela 1: Resultados comparativos para AP40

		$\alpha = 0,6$				$\alpha = 0,8$				$\alpha = 0,9$			
		Tempo(s)		Gap Raiz(%)		Tempo(s)		Gap Raiz(%)		Tempo(s)		Gap Raiz(%)	
p	β_{ij} $(\times 10^3)$	opt	Este Trab.	Peker e Kara (2015)	Este Trab.	Peker e Kara (2015)	opt	Este Trab.	Peker e Kara (2015)	Este Trab.	Peker e Kara (2015)	Este Trab.	Peker e Kara (2015)
2	26	0,73	116,92	**88,05**	10,76	12,50	0,69	66,50	**34,38**	6,68	8,36	0,67	33,22	**22,67**	5,51	5,61
3	26	0,80	**119,38**	253,86	5,96	8,08	0,74	**77,72**	86,86	6,78	8,36	0,71	59,05	**51,50**	6,26	6,96
4	26	0,84	**170,19**	1181,27	4,27	6,45	0,78	**78,39**	150,47	5,63	5,52	0,75	**46,49**	55,83	4,29	4,98
5	26	0,86	**209,39**	1993,52	3,84	5,70	0,80	**108,86**	146,78	3,83	4,89	0,77	**53,17**	57,61	3,53	4,12
6	24	0,85	**147,80**	2518,38	3,86	5,61	0,77	**88,06**	730,06	4,27	5,27	0,73	**52,78**	59,35	3,43	3,80
7	24	0,86	**171,98**	2408,17	3,43	4,72	0,79	**109,84**	472,94	4,30	4,33	0,75	**39,83**	44,52	2,65	3,01
8	24	0,88	**162,14**	2865,64	2,82	3,78	0,81	**56,11**	73,14	2,41	2,96	0,76	38,88	**32,34**	2,24	2,26
9	24	0,89	**167,42**	1786,78	2,30	3,03	0,83	**41,50**	50,33	1,62	2,08	0,77	32,53	**28,80**	1,71	1,85
10	24	0,90	**89,78**	793,42	2,23	2,20	0,84	**21,80**	23,69	1,41	1,44	0,78	**22,53**	25,47	1,53	1,57
Média		0,85	**150,56**	1543,23	4,39	5,79	0,78	**72,09**	196,52	4,10	4,80	0,74	42,05	**42,01**	3,46	3,80

Galoá { Este trabalho foi publicado utilizando o **Galoá** proceedings

Tabela 2: Resultados comparativos para AP50

		$\alpha = 0,6$					$\alpha = 0,8$					$\alpha = 0,9$				
			Tempo(s)		Gap Raiz(%)			Tempo(s)		Gap Raiz(%)			Tempo(s)		Gap Raiz(%)	
p	β_{ij} $(\times 10^3)$	opt	Este Trab.	Peker e Kara (2015)	Este Trab.	Peker e Kara (2015)	opt	Este Trab.	Peker e Kara (2015)	Este Trab.	Peker e Kara (2015)	opt	Este Trab.	Peker e Kara (2015)	Este Trab.	Peker e Kara (2015)
2	26	0,73	321,23	**288,34**	9,14	11,21	0,69	**103,52**	209,34	7,15	8,40	0,67	112,08	**73,81**	6,16	6,21
3	26	0,79	**813,97**	1198,20	6,86	9,53	0,73	**233,09**	666,53	6,54	8,20	0,70	233,00	**184,76**	6,04	7,07
4	26	0,84	**637,45**	2512,95	4,21	6,52	0,78	**285,33**	599,37	4,38	5,89	0,74	**280,03**	307,69	4,51	5,34
5	26	0,86	**880,78**	$> 3600,00$	3,90	5,76	0,80	**307,95**	1991,19	3,37	4,71	0,76	**400,64**	527,49	4,27	5,01
6	24	0,85	**302,22**	$> 3600,00$	3,06	5,21	0,76	**602,09**	$> 3600,00$	4,77	6,09	0,73	**264,67**	1067,67	3,64	4,14
7	24	0,87	**441,77**	$> 3600,00$	2,74	4,45	0,78	**565,91**	$> 3600,00$	3,99	4,98	0,74	**334,02**	352,11	3,11	3,54
8	24	0,88	**866,94**	$> 3600,00$	2,57	3,97	0,79	**383,99**	2416,50	3,16	4,02	0,75	**325,25**	2677,53	2,73	3,09
9	24	0,89	**1825,16**	$> 3600,00$	2,54	3,47	0,81	**249,55**	1383,50	2,09	2,70	0,77	**293,67**	1011,91	2,04	2,41
10	24	0,90	**477,69**	$> 3600,00$	1,71	2,59	0,83	**96,41**	207,37	1,29	1,81	0,78	**124,33**	179,11	1,72	1,81
Média		0,85	**729,69**	2746,66	4,08	5,86	0,77	**314,20**	1628,48	4,08	5,20	0,74	**263,08**	709,12	3,80	4,29

Galoá { Este trabalho foi publicado utilizando o **Galoá** proceedings

Tabela 3: Resultados detalhados deste trabalho para instâncias AP50 com $\alpha = 0,8$

	Gap					Tempo(s)	
p	opt	Raiz(%)	#nós	#CM	#cortes	CM	total
2	0,69	7,15	84	28689	520	4,69	103,52
3	0,73	6,54	279	87478	1088	13,77	233,09
4	0,78	4,38	374	56480	801	9,04	285,33
5	0,80	3,37	486	41544	671	6,47	307,95
6	0,76	4,77	1956	63243	686	9,19	602,09
7	0,78	3,99	2522	68066	666	9,90	565,91
8	0,79	3,16	1690	87579	660	12,34	383,99
9	0,81	2,09	1003	9172	253	1,49	249,55
10	0,83	1,29	181	60466	419	8,60	96,41

Ernst, A. T. e Krishnamoorthy, M. (1996). Efficient algorithms for the uncapacitated single allocation p-hub median problem. *Location science*, 4(3):139–154.

Farahani, R. Z., Hekmatfar, M., Arabani, A. B., e Nikbakhsh, E. (2013). Hub location problems: A review of models, classification, solution techniques, and applications. *Computers & Industrial Engineering*, 64(4):1096–1109.

Hwang, Y. H. e Lee, Y. H. (2012). Uncapacitated single allocation p-hub maximal covering problem. *Computers & Industrial Engineering*, 63(2):382–389.

Ishfaq, R. e Sox, C. R. (2011). Hub location–allocation in intermodal logistic networks. *European Journal of Operational Research*, 210(2):213 – 230. ISSN 0377-2217.

Janković, O., Mišković, S., Stanimirović, Z., e Todosijević, R. (2017). Novel formulations and VNS-based heuristics for single and multiple allocation p-hub maximal covering problems. *Annals of Operations Research*, 259(1-2):191–216. ISSN 0254-5330, 1572-9338.

Janković, O. e Stanimirović, Z. (2017). A general variable neighborhood search for solving the uncapacitated r -allocation p -hub maximal covering problem. *Electronic Notes in Discrete Mathematics*, 58:23–30. ISSN 15710653.

Kim, H. e O'Kelly, M. E. (2009). Reliable p-hub location problems in telecommunication networks. *Geographical Analysis*, 41(3):283–306. ISSN 1538-4632.

Peker, M. e Kara, B. Y. (2015). The p-hub maximal covering problem and extensions for gradual decay functions. *Omega*, 54:158–172.

Pessoa, A. (2019). Sparsemaxflowmincut. `https://github.com/artalvpes/SparseMaxFlowMinCut/`. Acessado: 2020-06-29.

Silva, M. R. e Cunha, C. B. (2017). A tabu search heuristic for the uncapacitated single allocation p -hub maximal covering problem. *European Journal of Operational Research*, 262(3):954–965. ISSN 03772217.

Tan, P. Z. e Kara, B. Y. (2007). A hub covering model for cargo delivery systems. *Networks*, 49(1): 28–39.