Algoritmo de descida em vizinhança variável aplicado ao problema de cobertura máxima de p-eixos não capacitados com alocação simples

Matheus de Araujo Butinholi, Alexandre Xavier Martins,
Paganini Barcellos de Oliveira, Diego Perdigão Martino
Universidade Federal de Ouro Preto, Instituto de Ciências Exatas e Aplicadas,
Departamento de Engenharia de Produção, João Monlevade, Brasil
Rua 36, 115 - Loanda, João Monlevade - MG, 35931-008
matheus.butinholi@aluno.ufop.edu.br, xmartins@ufop.edu.br,
paganini@ufop.edu.br, diego.martino@aluno.ufop.edu.br

RESUMO

O presente artigo aborda o problema de cobertura máxima de p-eixos não capacitados com alocação simples (*Uncapacitated Single Allocation p-hub Maximal Covering Problem* - USApHMCP), que objetiva maximizar a cobertura de um conjunto de nós de uma rede através de p-eixos. Uma heurística baseada na estratégia de busca em descida com vizinhança variável (*Variable Neighborhood Descent* - VND) foi desenvolvida para o problema. Dois diferentes conjuntos de instâncias, *Civil Aeronautics Board* (CAB) e *Australian Post* (AP), são utilizados para avaliar e comparar o desempenho do VND à metaheurística Busca Tabu (*Tabu Search* - TS) encontrada na literatura. Como resultado, o VND apresentou limites superiores melhores para instâncias de grande porte (AP), bem como um desempenho médio ligeiramente superior em tempo computacional de resolução para as instâncias CAB, de menor porte.

PALAVRAS CHAVE. Problema de cobertura máxima. P-eixos não capacitados com alocação simples. Descida em vizinhança variável.

MH – Metaheurísticas, L&T – Logística e Transportes

ABSTRACT

This paper addresses the Uncapacitated Single Allocation p-hub Maximal Covering Problem (USApHMCP), which aims to determine the best allocation for p-hubs within a node network in order to maximize the network coverage. We proposed a search strategy-based heuristic VND (Variable Neighborhood Descent) to solve the problem. Two different sets of literature test instances, Civil Aeronautics Board (CAB) and Australian Post (AP), were used to evaluate the performance of VND and to compare with the Tabu Search metaheuristic (TS) found in the literature. As a result, the VND presented better upper bounds for large-scale AP instances, as well as its performance was slightly greater than the TS, on average, in computational time performance for CAB instances, with smaller size.

KEYWORDS. Maximum coverage problem. Uncapacitated single allocation p-hub. Variable Neighborhood Descent.

MH - Metaheuristics, L&T - Logistics and Transportation

1. Introdução

As redes eixo-raio (*hub-and-spokes*) são canais de interligação entre nós concentradores de fluxo (eixos ou *hubs*) e nós periféricos. Dado um conjunto de nós, os eixos são escolhidos estrategicamente em função do volume de fluxo de atividade/serviço que se deseja transportar entre os nós, ou seja, deve-se estabelecer uma configuração de conexão em rede capaz de cobrir as demandas entre os nós e, ao mesmo tempo, levar em consideração o seu custo de transporte [O'Kelly, 1986]. [Janković et al., 2017] destacam que, na maioria dos casos, é inviável criar uma rede que possibilite o trajeto direto entre todos os pares de nós devido ao seu alto custo.

Aplicações reais de redes eixo-raio são facilmente encontradas em ambientes como o de transporte terrestre, sistemas de telecomunicação, sistemas logísticos e de transporte aéreo [Cunha e Silva, 2017]. Adicionalmente, [Janković et al., 2017] destacam que os problemas eixo-raio incluem também redes de satélites, industriais e de entregas postais. Para todos esses casos, [Cunha e Silva, 2017] afirmam que os eixos são empregados de modo a consolidar o tráfego dos nós periféricos, permitindo, assim, uma conexão eficiente e eficaz entre os arcos por meio de rotas simplificadas de baixo custo. Por outro lado, sempre que o total de eixos (p) e conexões é limitado, o *trade-off* entre o custo do tráfego e a cobertura dos nós periféricos pelos eixos se torna um desafio.

A ideia de se avaliar como objetivo do problema a máxima cobertura de um número pré-definido de p-eixos foi introduzido na literatura por [Campbell, 1994]. Em seu trabalho, Campbell propõe quatro diferentes abordagens e formulações de programação inteira para o problema. São elas: o problema de p-medianas; o problema de localização de facilidades não-capacitadas; o problema de p-centros e o problema de cobertura de eixos. O primeiro visa a abertura de p facilidades de forma a minimizar a soma das distâncias entre as facilidades e os nós periféricos, enquanto o segundo objetiva minimizar, ao mesmo tempo, a soma dos custos de instalação das facilidades e do suprimento dos nós (clientes), ambos pré-definidos por meio de parâmetros de custos de instalação e transporte, considerando que as facilidades tem capacidade infinita de suprimento. O terceiro é uma variação do problema de clusterização de dados que visa particionar um conjunto de nós em p *clusters* em função da minimização das distâncias entre os centros (eixos) e os pontos a serem atendidos. O quarto e último busca maximizar o percentual de cobertura dos p eixos, considerando que eles têm capacidade ilimitada de fluxo de serviços e ainda permite apenas alocações simples entre os nós.

Neste contexto, o presente artigo aborda o problema de cobertura máxima de p-eixos não capacitados com alocação simples, também conhecido como *Uncapacitated Single Allocation p-hub Maximal Covering Problem* - (USApHMCP). O problema busca maximizar a cobertura dos nós concentradores de fluxo, os eixos, não capacitados, por meio das diferentes possibilidades de organização das conexões em uma rede, desde que sejam conexões de atribuição simples entre cada nó e seu respectivo eixo. Trata-se de um problema da classe NP-difícil (NP-*hard*), isto é, para obter uma resolução do problema, o esforço computacional cresce exponencialmente à medida em que aumenta o número total de nós [Kara e Tansel, 2003]. Portanto, é previsto que formulações de programação inteira tenham dificuldade para alcançar a solução ótima do USApHMCP quando do aumento da dimensão do problema.

Motivado pela dificuldade de resolução exata de instâncias do USApHMCP de grande porte, [Cunha e Silva, 2017] propuseram uma abordagem aproximada para o problema via um algoritmo baseado na metaheurística Busca Tabu (*Tabu Search* - TS), introduzida por [Glover e Laguna, 1998]. O TS é um procedimento adaptativo que orienta as buscas locais em vizinhanças e que tem como característica principal a criação de uma lista tabu de soluções não permitidas, normalmente associadas às n últimas soluções visitadas. No algoritmo TS, dada uma solução inicial, são realizados procedimentos de descida completa em uma vizinhança e, paralelamente, as soluções visitadas vão sendo adicionadas à lista tabu. Dessa forma, evita-se reexplorar soluções anteriores, possibilitando encontrar novas vizinhanças até que um critério de parada seja satisfeito.

Visando estabelecer uma estratégia aproximada até então não explorada para o problema, este trabalho propõe uma heurística construtiva baseada na estratégia de busca em descida com vizinhança variável - VND, proposta por [Mladenović e Hansen, 1997] para o USApHMCP. Trata-se de um algoritmo que permite a descida completa em diferentes estruturas de vizinhança (ambiente de diversificação de soluções sob um mesmo critério), com ordem pré-definida. Além disso, sempre que uma nova solução de melhora é identificada, o procedimento de busca retorna à primeira vizinhança pré-definida. Para avaliar o desempenho computacional do VND, dois diferentes conjuntos de instâncias padrão da literatura, *Civil Aeronautic Board* - CAB e *Australian Post* - AP, são resolvidos e comparados aos resultados obtidos pelo TS desenvolvido por [Cunha e Silva, 2017].

Baseado na metodologia proposta, o artigo está organizado da seguinte forma: a seção 2 descreve a formulação matemática do problema; a seção 3 apresenta o algoritmo heurístico proposto; a seção 4 apresenta os experimentos computacionais realizados e a seção 5 apresenta as conclusões e sugestões para continuidade da pesquisa.

2. Problema de cobertura máxima de p-eixos não capacitados com alocação simples

Na literatura, é possível encontrar formulações matemáticas para o problema de alocação de p-eixos. Mais especificamente, no escopo deste trabalho, optou-se pelo uso da formulação do USApHMCP proposta por [Kara e Peker, 2015], conforme o apresentado a seguir.

Seja um conjunto de nós N e parâmetros ρ, W_{ij} e C_{ij} para todo par ordenado $(i, j) \in N$ correspondente ao total de eixos, o fluxo de serviços e o custo de transporte entre os nós, respectivamente. É possível estabelecer uma rota entre os nós $(i- > k- > m- > j)$, formada por dois nós periféricos i e j e dois eixos k e m, desde que o custo total da rota seja menor ou igual a β, ou seja, $\chi C_{ik} + \alpha C_{km} + \delta C_{mj} \leq \beta$, tal que χ e δ são fatores de desconto entre eixo e raio e α é o fator de desconto das rotas entre eixos. Todos os fatores de desconto são valores entre 0 e 1. Para avaliar a possibilidade de cobertura das rotas, o parâmetro parâmetro binário $a_{i,j}^{k,m}$ é introduzido à formulação.

$$a_{i,j}^{k,m} = \begin{cases} 1 : \chi C_{ik} + \alpha C_{km} + \delta C_{mj} \ \leq \ \beta \\ 0 : \text{caso contrário.} \end{cases}$$

Portanto, dada uma rede de nós, o problema consiste em alocar de forma eficiente os nós eixos aos nós não eixos de modo a maximizar a cobertura de todo o fluxo de serviço da rede. Para isso, cada nó periférico somente poderá estar conectado por um único eixo (concentrador de fluxo). O eixo, por sua vez, pode ser conectado a todos os nós periféricos, bem como deve ser conectado a todos os outros eixos da rede. A Figura 1 ilustra uma possível alocação eixo-raio em uma rede de nós que possui um total de dez nós e ρ igual a 2, ou seja, apenas dois concentradores de fluxo são selecionados.

A Figura 1 apresenta a conexão simples dos nós periféricos $n = \{1, 3, 7, 9, 10\}$ ao eixo concentrador de fluxo 5 e, ao mesmo tempo, a conexão dos nós $n = \{2, 4, 8\}$ ao eixo 6. Além disso, é preciso destacar que os eixos fazem conexões com todos os raios e ainda possuem ligações com todos os outros eixos da rede, como representado pelo arco entre os nós 5 e 6, visando a concentração de fluxo de transição.

Neste contexto, ao introduzir as variáveis de decisão X_{ij} e Z_{ij} ao problema, responsáveis por indicar se um nó i está conectado (1) a um eixo j ou não (0) e estabelecer a fração de fluxo

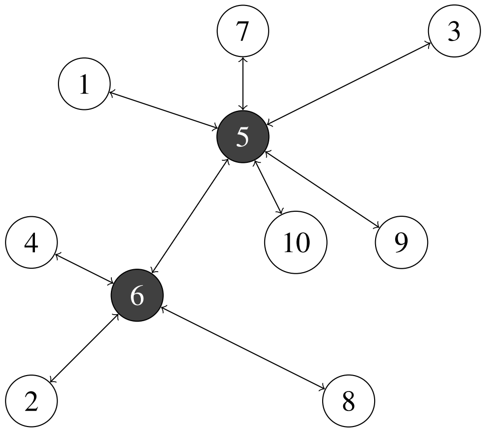

Figura 1: Representação de uma rede eixo-raio.

coberto do nó i até o nó j, respectivamente, é possível estruturar a formulação matemática descrita pelas Equações de (1) a (7).

$$\max \sum_{i\in N}\sum_{j\in N} W_{ij}Z_{ij} \tag{1}$$

$$\text{s.a:} \sum_{k\in N} X_{kk} = p \tag{2}$$

$$\sum_{k\in N} X_{ik} = 1 \qquad \forall\, i \in N \tag{3}$$

$$X_{ik} \leq X_{kk} \qquad \forall\, i,k \in N \tag{4}$$

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

$$X_{ij} \in \{0,1\} \qquad \forall\, i,j \in N \tag{6}$$

$$Zij \geq 0 \qquad \forall\, i,j \in N \tag{7}$$

A função objetivo do problema (1) visa maximizar o fluxo de cobertura da rede. A restrição (2) assegura que exatamente p-eixos serão alocados, enquanto as restrições (3) e (4) garantem que cada raio tenha apenas um eixo a ele associado e que cada nó seja alocado a um único eixo, respectivamente. Por fim, as restrições (5) são utilizadas para indicar o percentual de fluxo coberto entre os pares de nós i e j. Note que os parâmetros λ_{ij} são utilizados para apertar as restrições (5), e esses assumem valores que são indicados pela Equações (8).

$$\lambda_{ij} = max_{km} a_{i,j}^{k,m} \qquad \forall\, (i,j) \;\in\; N \tag{8}$$

Contudo, verifica-se que quando o nó periférico j está ligado ao eixo m, $X_{jm} = 1$, as restrições (5) são limitadas a $Z_{ij} \leq \sum_{k\in N} a_{i,j}^{k,m} X_{ik}$. Por outro lado, quando $X_{jm} = 0$, as restrições assumem a configuração $Z_{ij} \leq \sum_{k\in N} a_{i,j}^{k,m} X_{ik} + \lambda_{ij}$. Por fim, o domínio das variáveis X_{ij} e Z_{ij} é dado pelas Equações (6) e (7), respectivamente.

3. Algoritmo de descida em vizinhança variável

Para representar computacionalmente a solução do problema, foi utilizada uma matriz com dimensões $2 \times n$, ou seja, 2 linhas e n colunas, referentes à quantidade de nós presentes em cada instância. A Tabela 1 mostra como seria representada a solução ilustrada na Figura 1.

0	0	0	0	1	1	0	0	0	0
5	6	5	6	5	6	5	6	5	5

Tabela 1: Representação da solução

Note que uma solução viável para o problema mostrado na Figura 1 pode ser obtida a partir da seleção de dois eixos iniciais, no caso, os nós 5 e 6. Portanto, para se obter um solução inicial qualquer viável para o problema, basta associar um conjunto de ρ nós, para cobrir o fluxo de serviço da rede. Neste sentido, optou-se pela elaboração de um procedimento de construção de solução inicial que se apropriasse dessa propriedade, o qual seriam avaliados o percentual de cobertura máxima referente à soluções que contenham ρ nós da rede. No exemplo mostrado na Tabela 1, foi realizado um procedimento sistemático de testes, dois a dois, que resultou na seleção do par de eixos (5 e 6) que obteve a maior cobertura.

Nos casos em que $\rho > 2$, após encontrar uma solução com dois eixos, os demais eixos vão sendo incorporados à solução corrente de forma a maximizar a cobertura da configuração da rede, ou seja, enquanto o total de eixos for menor do que ρ, novos eixos vão sendo adicionados, um a um, tendo como critério de seleção aquele que possibilita o aumento mais expressivo no percentual de cobertura na rede. Portanto, uma vez inseridos ρ eixos, obtém-se uma solução inicial para o problema.

Para aprimorar a qualidade da solução, foi desenvolvido um procedimento de permuta entre buscas locais para o problema conhecido na literatura como *Variable Neighborhood Descent* (VND). Proposto por [Mladenović e Hansen, 1997], o procedimento de Descida em Vizinhança Variável busca explorar os espaços do conjunto de soluções vizinhas à solução corrente (estruturas de vizinhança) de forma eficiente, através de trocas sistemáticas de informações e mecanismos de buscas. Trata-se de um procedimento que propõe a permuta entre estruturas de vizinhança, ocasionando uma maior proporção de soluções visitadas em comparação a outros métodos que possuem apenas uma única estratégia de busca local. O pseudocódigo do VND proposto é apresentado no Algoritmo 1.

No Algoritmo 1, o VND tem como parâmetros de entrada a quantidade de estruturas de vizinhanças q e a solução corrente s (linha 1). Na linha 3, o algoritmo inicializa a variável k, responsável por controlar a sequência de visita nas estruturas de vizinhanças. O laço que compreende as linhas 4 a 13 irá se repetir até que VND não seja capaz mais de melhorar a solução corrente s, ou seja, quando $k = q$. A linha 5 representa as buscas locais em s geradas na estrutura de vizinhança k, enquanto a linha 6 avalia se a nova solução s' é melhor que s. Caso s' seja melhor que s, deve-se, então, atualizar a solução corrente e reiniciar o processo de buscas locais na primeira estrutura de vizinhança pré-determinada (linhas 6 a 9). Caso contrário, a próxima estrutura de vizinhança será acionada (linhas 10 a 12). Ao final, o pseudocódigo retorna melhor solução s, conforme descrito na linha 14.

O VND foi elaborado por meio da permuta entre cinco diferentes estruturas de vizinhanças, $N^q = \{N^1, N^2, ..., N^5\}$, de modo que cada vizinhança avaliasse diferentes espaços de soluções, ou seja, cada uma permite alcançar diferentes pontos de melhora e/ou piora para o problema. A definição da sequência de exploração das vizinhanças foi feita a partir de testes computacionais pre-

Algoritmo 1: Algoritmo de descida em vizinhança variável

Função *VND*(q, s)
início
    $k \leftarrow 1$
    enquanto $k \leq q$ **faça**
        $s' \leftarrow$ Melhor Vizinho de s em $N^{(k)}$
        se $f(s') > f(s)$ **então**
            $s \leftarrow s'$
            $k \leftarrow 1$
        fim
        senão
            $k \leftarrow k + 1$
        fim
    fim
    retorna s
fim

liminares, a partir dos quais verificou-se que alterações nos eixos seguidas de alocações dos raios resultariam em melhorias significativas no valor da função objetivo. Portanto, a ordem foi definida levando em consideração as características específicas do problema.

As duas primeiras estruturas de vizinhança N^1 e N^2 são bastante semelhantes em relação ao movimento proposto. Ambas realizam trocas em que os nós periféricos assumem a função de um eixo. A diferença entre elas é que em N^1, após alocar um nó periférico como eixo, faz-se uma realocação simples dos raios aos eixos. Essa alocação simples se dá pelo custo, isto é, dado um nó periférico, esse vai se conectar ao eixo que possui o menor custo. Por outro lado, N^2 realiza a realocação de um nó periférico como eixo e, em seguida, efetua uma realocação dos raios visando avaliar o grau de cobertura do fluxo de serviço do problema. A Tabela 2 ilustra o comportamento oriundo do movimento na vizinhança N^1 ou N^2, sendo que o nó periférico 1 substituí o eixo 5 na primeira iteração.

Antes do movimento

0	0	0	0	**1**	1	0	0	0	0
5	6	5	6	5	6	5	6	5	5

Depois do movimento

1	0	0	0	**0**	1	0	0	0	0
1	6	6	6	1	6	1	6	6	6

Tabela 2: Representação da primeira e da segunda estruturas de vizinhança

A terceira estrutura de vizinhança, definida como N^3, permite que todos os raios do problema possam variar o seu respectivo eixo, ou seja, são feitos testes sistemáticos de realocação dos nós periféricos a outros eixos da rede, como mostra a Tabela 3. Note que o nó periférico 1 está inicialmente vinculado ao eixo 5 e que, após a primeira iteração da vizinhança N^3, ele passa a ser associado ao eixo 6. Este procedimento continua até percorrer todo os nós periféricos.

A estrutura N^4 permite variar os raios dois a dois. Para isso, são escolhidos dois raios atendidos por eixos diferentes e realizada a troca de alocação dos raios aos eixos. A Tabela 4 evidencia esse tipo de movimento na estrutura de vizinhança N^4, no qual os nós periféricos 1 e 2 são alocados inicialmente aos eixos 5 e 6 e, posteriormente, são realocados para 6 e 5, respectivamente.

Antes do movimento

0	0	0	0	1	1	0	0	0	0
5	6	5	6	5	6	5	6	5	5

Depois do movimento

0	0	0	0	1	1	0	0	0	0
6	6	5	6	5	6	5	6	5	5

Tabela 3: Representação da terceira estrutura de vizinhança

Antes do movimento

0	0	0	0	1	1	0	0	0	0
5	**6**	5	6	5	6	5	6	5	5

Depois do movimento

0	0	0	0	1	1	0	0	0	0
6	**5**	5	6	5	6	5	6	5	5

Tabela 4: Representação da quarta estrutura de vizinhança

Por fim, a vizinhança N^5 é responsável pela permuta de um par de eixos com um par de nós periféricos, ou seja, conforme mostrado no exemplo da Tabela 5, os eixos 5 e 6 foram substituídos pelos nós periféricos 1 e 2, respectivamente. Após a realização das trocas, os raios do problema são realocados em função do percentual de cobertura dos novos eixos.

Antes do movimento

0	**0**	0	0	**1**	**1**	0	0	0	0
5	6	5	6	5	6	5	6	5	5

Depois do movimento

1	**1**	0	0	**0**	**0**	0	0	0	0
1	2	1	1	2	1	2	2	1	2

Tabela 5: Representação da quinta estrutura de vizinhança

4. Experimentos computacionais

O algoritmo VND desenvolvido neste trabalho foi implementado utilizando a linguagem C++ compatível com o sistema operacional Windows 10 em um *notebook* com processador Intel Core i7-8550 1.80 GHz, 4 Cores (8 núcleos) e 8 GB de memória RAM. O resolvedor comercial ILOG CPLEX 12.6 foi utilizado como estratégia exata de resolução das instâncias avaliadas, sendo esse implementado na linguagem de AMPL. Já os experimentos realizados por [Cunha e Silva, 2017], que são aqui comparados, foram realizados em um computador com processador Intel Xeon E7-2870 2.40 GHz, 10 Cores (20 núcleos) e 512 GB de memória RAM.

É válido ressaltar que a comparação proposta neste trabalho não é a ideal, tendo em vista a diferença entre a capacidade de processamento entre as máquinas, seja pela comparação numérica da quantidade de Cores, memória RAM ou pela frequência de relógio (GHz). Para se ter uma comparação justa e coerente, o trabalho proposto também deveria ter implementado o TS de [Cunha e Silva, 2017] e o comparado com o VND, embora seja perceptível que a configuração da máquina adotada por [Cunha e Silva, 2017] seja superior. Essa atividade faz parte de uma etapa posterior desta pesquisa, ainda em processo de desenvolvimento.

Todos as instâncias avaliadas foram executadas uma única vez, por algoritmo, tendo em vista que o VND proposto não possui parâmetros a serem calibrados e, no caso do CPLEX, uma única semente padrão do resolvedor foi previamente fixada.

4.1. Características das instâncias

Dois diferentes conjuntos de instâncias teste da literatura foram avaliadas neste trabalho, são elas: *Civil Aeronautics Board* (CAB) e *Australian Post* (AP). As instâncias CAB foram introduzidas inicialmente por [O'Kelly, 1987] e são comumente utilizadas em problemas eixo-raio. As instâncias CAB são agrupadas em função do número de nós, $n = \{10, 15, 20, 25\}$, número de eixos a serem fixados, $p = \{2, 3, 4, 5\}$ e o fator de desconto entre os eixos, $\alpha = \{0.2, 0.4, 0.6, 0.8, 1.0\}$. Os fatores de desconto χ e δ em todos os casos são setados em 1, ou seja, não há desconto pelo transporte entre raios, uma vez que o problema não permite conexões entres eles. Como alternativa de comparação, os valores máximos de custo da rota β foram os mesmos utilizados por [Cunha e Silva, 2017].

Em se tratando da instâncias AP, introduzidas por [Ernst e Mohan, 1996], foram avaliadas instâncias com número de nós n assumindo os valores $n = \{10, 20, 25, 40, 50, 100\}$. Quanto aos fatores de desconto, em todos os casos, α é fixado em 0.75, enquanto que χ e δ são fixados em 1. Novamente, para padronizar as comparações, ficou estabelecido que o β seria o mesmo proposto por [Cunha e Silva, 2017] para cada uma das instâncias.

4.2. Resultados e discussões

Nessa seção, cada uma das tabelas apresentadas contém o número n de nós, a quantidade de eixos ρ, o fator de desconto α e o custo máximo permitido β. Além disso, para as instâncias CAB, de menor porte, as Tabelas 6-9 também apresentam o valor da solução ótima obtida pelo CPLEX (optsol), o percentual de cobertura dos fluxos da rede (cov(%)) e tempo computacional de execução dos algoritmos, em segundos (t(s)), para se encontrar a solução dos problemas, bem como seus respectivos os valores médios geométricos. Em todos os casos foram utilizados a média geométrica deslocada [Achterberg, 2007], visando diminuir a distorção entre o desempenho médio dos algoritmos em função da dimensão dos problemas. Para isso, foi estabelecido que o deslocamento dos valores de t(s) seriam iguais a 10, conforme sugerido por [Achterberg, 2007].

Tabela 6: Resultados obtidos por meio das instâncias CAB com n = 10.

Instâncias				CPLEX			VND	TS
n	ρ	α	β	optsol	cov(%)	t(s)	t(s)	t(s)
10	2	0.75	1425	994540	99.55	0.91	0.00	0.00
10	3	0.75	1117	999026	100.00	0.55	0.00	0.05
10	4	0.75	811	999026	100.00	0.36	0.02	0.08
10	5	0.75	736	991270	99.22	0.45	0.03	0.09
10	2	0.75	1627	994540	99.55	0.86	0.02	0.02
10	3	0.75	1185	990542	99.15	0.44	0.02	0.03
10	4	0.75	970	999026	100.00	0.36	0.02	0.08
10	5	0.75	863	999026	100.00	0.34	0.02	0.09
10	2	0.75	1671	987490	98.85	0.59	0.02	0.02
10	3	0.75	1387	984530	98.55	0.41	0.02	0.03
10	4	0.75	1148	999026	100.00	0.36	0.02	0.08
10	5	0.75	1079	999026	100.00	0.34	0.02	0.13
10	2	0.75	1744	999026	100.00	0.38	0.02	0.00
10	3	0.75	1589	999026	100.00	0.36	0.02	0.03
10	4	0.75	1457	999026	100.00	0.38	0.03	0.06
10	5	0.75	1413	999026	100.00	0.38	0.02	0.09
10	2	0.75	1839	984836	98.58	0.41	0.02	0.02
10	3	0.75	1791	999026	100.00	0.33	0.02	0.03
10	4	0.75	1770	999026	100.00	0.36	0.02	0.06
10	5	0.75	1766	999026	100.00	0.34	0.02	0.13
Média						0.44	0.01	0.01

Nos casos em que algum dos algoritmos não alcançou a solução ótima dos problemas (Tabelas 8-9), são apresentados também a distância percentual que a solução encontrada está em relação à solução ótima do problema (*gaps*). Em contrapartida, sempre que houver $gap = 0$ nas Tabelas 8-9, significa que a solução apresentada pelo VND foi igual a do CPLEX.

As Tabelas 6 e 7 mostram que os dois menores conjuntos de instâncias CAB, com nós entre $n = \{10 \text{ e } 15\}$, respectivamente, o VND e o TS atingiram a solução ótima em todos os casos com um tempo médio inferior ao resolvedor comercial CPLEX. Por outro lado, o VND se demonstrou mais eficiente pelo fato de obter as mesmas soluções com um tempo de processamento menor quando comparado ao algoritmo TS.

Tabela 7: Resultados obtidos por meio das instâncias CAB com n = 15.

Instâncias				CPLEX			VND	TS
n	ρ	α	β	optsol	cov(%)	t(s)	t(s)	t(s)
15	2	0.75	2004	2358068	99.71	3.06	0.03	0.03
15	3	0.75	1638	2358068	99.71	1.88	0.05	0.19
15	4	0.75	1324	2364942	100.00	0.63	0.09	0.34
15	5	0.75	1149	2353712	99.53	0.88	0.16	0.58
15	2	0.75	2019	2364942	100.00	1.02	0.02	0.05
15	3	0.75	1741	2364942	100.00	0.66	0.06	0.19
15	4	0.75	1436	2364942	100.00	0.52	0.09	0.33
15	5	0.75	1287	2364942	100.00	0.41	0.13	0.59
15	2	0.75	2103	2364942	100.00	0.44	0.03	0.03
15	3	0.75	1844	2304218	97.43	1.20	0.06	0.19
15	4	0.75	1756	2364942	100.00	0.47	0.13	0.33
15	5	0.75	1560	2320434	98.12	0.44	0.11	0.59
15	2	0.75	2424	2364942	100.00	0.45	0.03	0.06
15	3	0.75	2165	2320434	98.12	0.69	0.05	0.19
15	4	0.75	2100	2364942	100.00	0.56	0.09	0.34
15	5	0.75	2080	2320434	98.12	0.50	0.11	0.59
15	2	0.75	2611	2364942	100.00	0.47	0.03	0.06
15	3	0.75	2610	2364942	100.00	0.55	0.02	0.16
15	4	0.75	2605	2364942	100.00	0.45	0.06	0.36
15	5	0.75	2600	2320434	98.12	0.52	0.08	0.56
Média						0.77	0.09	0.28

Tabela 8: Resultados obtidos por meio das instâncias CAB com n = 20.

Instâncias				CPLEX			VND		TS
n	ρ	α	β	optsol	cov(%)	t(s)	gap(%)	t(s)	t(s)
20	2	0.75	1851	5747720	99.88	8.95	0.00	0.09	0.19
20	3	0.75	1549	5743058	99.80	2.06	0.00	0.20	0.56
20	4	0.75	1356	5754594	100.00	4.70	0.00	0.27	1.08
20	5	0.75	1162	5722742	99.45	4.02	0.00	0.67	1.86
20	2	0.75	2067	5737094	99.70	3.27	0.00	0.09	0.17
20	3	0.75	1744	5739610	99.74	1.14	0.00	0.22	0.53
20	4	0.75	1473	5754594	100.00	1.05	0.00	0.38	1.11
20	5	0.75	1386	5754594	100.00	3.22	0.00	0.73	1.84
20	2	0.75	2255	5748824	99.90	3.17	0.00	0.09	0.16
20	3	0.75	1996	5719090	99.48	1.31	0.00	0.28	0.56
20	4	0.75	1835	5754594	100.00	0.67	0.00	0.53	1.09
20	5	0.75	1663	5754594	100.00	2.13	0.00	0.77	1.84
20	2	0.75	2493	5754594	100.00	0.89	0.00	0.11	0.17
20	3	0.75	2264	5754594	100.00	0.70	0.56	0.22	0.55
20	4	0.75	2154	5754594	100.00	0.84	0.00	0.44	1.11
20	5	0.75	2118	5752254	99.96	1.09	0.00	0.91	1.81
20	2	0.75	2611	5754594	100.00	1.11	0.00	0.06	0.19
20	3	0.75	2605	5754594	100.00	1.19	0.77	0.11	0.53
20	4	0.75	2601	5754594	100.00	1.11	0.00	0.33	1.08
20	5	0.75	2600	5710086	99.23	1.1	0.00	0.38	1.81
Média						2.05	0.06	0.34	0.89

De maneira similar, mas para instâncias um pouco maiores, com número de nós $n = \{20$ e $25\}$, o TS se mostrou mais efetivo que o VND por conseguir atingir a solução ótima em todos os casos, embora tenha um desempenho médio pior em tempo do que o VND, conforme mostram as Tabelas 8 e 9. Esse cenário indica que, embora o VND seja eficiente em obter boas soluções, em pouco tempo, para as instâncias CAB, ele ainda pode ser aprimorado visando explorar novas soluções e fugir de ótimos locais, seja pela incorporação de novas estruturas de vizinhanças, pela utilização de uma outra solução inicial como ponto de partida ou, ainda, estabelecendo critérios de diversificação da ordem de pesquisa das buscas.

Para o segundo conjunto de instâncias (AP), em sua maioria de grande porte, optou-se por

Tabela 9: Resultados obtidos por meio das instâncias CAB com n = 25.

Instâncias				CPLEX			VND		TS
n	ρ	α	β	optsol	cov(%)	t(s)	gap(%)	t(s)	t(s)
25	2	0.75	2136	8540006	100.00	20.70	0.00	0.33	0.42
25	3	0.75	1913	8533986	99.93	49.42	0.00	0.77	1.34
25	4	0.75	1617	8533986	99.93	19.23	0.00	0.95	2.67
25	5	0.75	1346	8540006	100.00	4.81	0.00	2.16	4.52
25	2	0.75	2401	8536326	99.96	16.66	0.00	0.36	0.44
25	3	0.75	2099	8540006	100.00	13.55	0.04	0.52	1.48
25	4	0.75	1881	8517004	99.73	14.55	0.02	1.56	2.69
25	5	0.75	1597	8526490	99.84	2.73	0.00	2.83	4.52
25	2	0.75	2557	8536326	99.96	6.34	0.00	0.36	0.42
25	3	0.75	2336	8536326	99.96	4.70	0.00	0.53	1.36
25	4	0.75	2184	8540006	100.00	2.58	0.10	1.64	2.67
25	5	0.75	2002	8524146	99.81	6.28	0.28	2.81	4.5
25	2	0.75	2713	8536326	99.96	3.78	0.00	0.34	0.41
25	3	0.75	2552	8536326	99.96	2.92	0.00	1.22	1.34
25	4	0.75	2457	8540006	100.00	2.70	0.14	2.86	2.69
25	5	0.75	2307	8490176	99.42	3.38	0.00	4.16	4.48
25	2	0.75	2806	8527758	99.86	1.98	0.00	0.34	0.45
25	3	0.75	2762	8540006	100.00	1.84	0.00	0.83	1.3
25	4	0.75	2726	8536326	99.96	1.58	0.00	1.61	2.69
25	5	0.75	2725	8536326	99.96	1.72	0.00	2.44	4.53
Média						7.07	0.02	1.38	2.15

não apresentar os dados referentes ao valor da solução ótima dos problemas, pelo CPLEX, tendo em vista que, em alguns casos, obter os valores ótimos demandariam um tempo computacional superior a 24 horas de processamento. Os resultados em negrito inferem a melhor solução encontrada para cada instância e, ainda, são apresentadas as distâncias percentuais (gap^b(%)) entre os resultados encontrados pelos algoritmos VND e TS, conforme mostra a Tabela 10.

Tabela 10: Resultados das instâncias da AP (n = 10,20,25,40,50,100).

Instâncias				VND			TS		
n	ρ	α	β	sol	t(s)	gap^b(%)	sol	t(s)	gap^b(%)
10	2	0.75	40383	**3978.92**	0.02	0.00	**3978.92**	0.02	0.00
10	3	0.75	34772	**3937.31**	0.02	0.00	**3937.31**	0.05	0.00
10	4	0.75	32574	**3954.53**	0.02	0.00	**3954.53**	0.10	0.00
10	5	0.75	32531	**3954.53**	0.03	0.00	**3954.53**	0.13	0.00
20	2	0.75	45954	**3973.21**	0.16	0.00	**3973.21**	0.20	0.00
20	3	0.75	43400	**3973.20**	0.38	0.00	**3973.20**	0.60	0.00
20	4	0.75	38607	**3974.27**	0.74	0.00	**3974.27**	1.21	0.00
20	5	0.75	37868	**3973.20**	1.50	0.00	**3973.20**	2.01	0.00
25	2	0.75	53207	**3976.57**	0.61	0.00	**3976.57**	0.46	0.00
25	3	0.75	46608	**3972.51**	1.45	0.00	**3972.51**	1.47	0.00
25	4	0.75	45552	**3976.68**	2.69	0.00	**3976.68**	2.95	0.00
25	5	0.75	45552	**3976.68**	3.90	0.00	**3976.68**	5.06	0.00
40	2	0.75	61683	**3978.92**	10.51	0.00	**3978.92**	3.29	0.00
40	3	0.75	58193	**3978.92**	38.37	0.00	**3978.92**	10.13	0.00
40	4	0.75	52265	3976.19	49.96	0.03	**3977.28**	20.82	0.00
40	5	0.75	49741	3977.28	89.97	0.02	**3977.97**	35.62	0.00
50	2	0.75	65523	**3978.69**	40.66	0.00	**3978.69**	7.90	0.00
50	3	0.75	60132	3978.39	98.16	0.00	**3978.42**	24.16	0.00
50	4	0.75	52906	3978.03	209.25	0.02	**3978.92**	49.05	0.00
50	5	0.75	50708	3977.62	715.82	0.03	**3978.92**	82.87	0.00
100	2	0.75	65915	**3978.92**	2561.54	0.00	3978.69	7.90	0.01
100	3	0.75	60659	**3978.79**	10061.17	0.00	3978.42	24.16	0.01
100	4	0.75	56125	**3978.92**	13679.10	0.00	**3978.92**	49.05	0.00
100	5	0.75	54243	3978.72	47504.96	0.00	**3978.92**	82.87	0.00
Média					60.72	0.00		9.89	0.00

Note pela Tabela 10 que, para a maior parte das instâncias, os algoritmos encontraram as mesmas soluções para os problemas. Além disso, à medida em que o problema cresce, o VND obtém melhores limites superiores que o TS, especialmente em se tratando das instâncias com $n = 100$. Por outro lado, o esforço computacional gasto pelo VND para atingir soluções melhores que o TS é suficientemente maior, principalmente por se tratar de um procedimento de busca exaustiva, o qual não é utilizado um critério de parada baseado no tempo de processamento.

5. Conclusão e próximos estudos

Este trabalho estudou o problema de cobertura máxima de p-eixos não capacitados com alocação simples, também conhecido como *Uncapacitated Single Allocation p-hub Maximal Covering Problem* (USApHMCP). O problema consiste em selecionar p-eixos dentro de uma rede de nós de forma a maximizar a cobertura da rede e considerando que os nós periféricos (que não são eixos) devem ser alocados de forma simples a apenas um eixo. Foi proposta uma estratégia de resolução aproximada para o USApHMCP, por meio de uma heurística baseada na estratégia de busca em descida com vizinhança variável (VND). Foram utilizados dois conjuntos de instâncias teste contidos na literatura, *Civil Aeronautics Board* (CAB) e *Australian Post* (AP), para avaliar e comparar o desempenho do VND em relação à metaheurística Busca Tabu (TS) [Cunha e Silva, 2017].

Os resultados computacionais realizados comprovaram que tanto o VND quanto o TS são boas alternativas para a resolução do problema. Ambos algoritmos atingem as soluções ótimas, como comprovado pelo CPLEX, para grande maioria das instâncias avaliadas. Para as instâncias denominadas CAB, o TS encontra a solução ótima para todos os casos, porém, o tempo de processamento médio é maior do que o apresentado pelo VND. Por outro lado, o VND somente não atingiu a solução ótima em 8.75% (7 em 80) das instâncias CAB com gap(%) médio inferior a 0.8% no pior caso. Em relação às instâncias de maior porte do AP, ou seja, as instâncias que possuem um número maior de nós, verificou-se também que o VND obteve limites superiores, em média, melhores que a TS.

Por fim, é importante ressaltar que os resultados aqui apresentados são informações preliminares provenientes de um projeto de iniciação científica. Desta forma, como etapa seguinte da pequisa, novas estratégias de geração de solução inicial e estruturas de vizinhança, bem como outras metaheurísticas serão implementadas, visando aprimorar os resultados até então obtidos.

Agradecimentos

Os autores agradecem a Universidade Federal de Ouro Preto (UFOP), a Fundação de Amparo à Pesquisa de Minas Gerais (Fapemig), a Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (Capes) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pelo apoio e suporte financeiro para realização desta pesquisa.

Referências

Achterberg, T. (2007). *Constraint Integer Programming*. PhD thesis, Technische Universität Berlin, Fakultät II - Mathematik und Naturwissenschaften.

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

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

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

Glover, F. e Laguna, M. (1998). Tabu search. In *Handbook of combinatorial optimization*. Springer.

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*, 259:191–216.

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

Kara, B. Y. e Tansel, B. C. (2003). The single-assignment hub covering problem: Models and linearizations. *Journal of the Operational Research Society*, 54:59–64.

Mladenović, N. e Hansen, P. (1997). Variable neighborhood search. *Computers & operations research*, 24:1097–1100.

O'Kelly, M. E. (1986). Activity levels at hub facilities in interacting networks. *Geographical Analysis*, 18:343–356.

O'Kelly, M. E. (1987). A quadratic integer program for the location of interacting hub facilities. *European Journal of Operational Research*, 32:393–404.