

## METAHEURÍSTICA HÍBRIDA COM BUSCA POR AGRUPAMENTO APLICADO AO PROBLEMA DE LOCALIZAÇÃO DE HUBS COM RESTRICÇÕES DE CAPACIDADE

**Wesley Gomes de Almeida**

Instituto Nacional de Pesquisas Espaciais – INPE  
Av. dos Astronautas 1758, Jardim da Granja  
12227-010 São José dos Campos, São Paulo, Brasil  
wesgomes2005@gmail.com

**Edson Luiz França Senne**

Universidade Estadual Paulista – UNESP  
Faculdade de Engenharia – Campus de Guaratinguetá  
Av. Ariberto Pereira da Cunha, 333, Portal das Colinas  
12516-410 Guaratinguetá, SP, Brasil  
elfsenne@feg.unesp.br

### RESUMO

Neste trabalho estuda-se o problema de localização de concentradores com restrições de capacidade. Um método *simulated annealing* é proposto e em seguida é apresentada uma estratégia utilizando agrupamentos para identificar regiões do espaço de busca que são mais promissoras para obtenção de boas soluções. Testes computacionais foram realizados na tentativa de verificar os resultados do uso métodos com busca por agrupamentos.

**PALAVRAS-CHAVE.** Otimização Combinatória, Problemas de Localização-Alocação, Localização de Concentradores Capacitado, Busca por Agrupamentos.

### ABSTRACT

In this work the capacitated hub location problem is studied. A simulated annealing method is proposed and then is present strategy using clustering to identify the regions of the search space which are more promising for obtaining good solutions. Computational tests were made in the trying to verify the results of use to clustering search methods.

**KEYWORDS.** Combinatorial Optimization, Location Allocation Problems, Capacited Single Hub Location Allocation Problem, Clustering Search.

## 1. Introdução

Existem diversos problemas definidos em redes, na maioria dos casos a comunicação entre os nós da rede não acontece de forma direta, mas através de nós especiais denominados concentradores. Isto ocorre com frequência, por exemplo, em redes de transporte e em redes de telecomunicação. Nestes casos, diz-se que a rede é do tipo *hub-and-spoke* (Aykin, 1994).

Um modelo desse tipo de rede pode ser exemplificado imaginando-se um serviço de transporte rodoviário de cargas em que a demanda individual dos clientes não é suficiente para lotar um veículo em uma única viagem. Por esse motivo, as cargas são agregadas e transportadas em conjunto. Para isto, empresas que operam este tipo de serviço possuem instalações físicas localizadas em diversas regiões para consolidar as cargas oriundas de diversas origens. Portanto, este tipo de serviço compreende as operações de coleta (de um cliente até um terminal de consolidação de origem), transferência (de um terminal de consolidação de origem para um terminal de consolidação de destino) e distribuição (do terminal de consolidação de destino até o cliente final). Para uma empresa deste tipo, um bom planejamento da rede de transporte, com os terminais de consolidação (concentradores) bem localizados, pode implicar em ganhos financeiros significativos.

Diversas aplicações práticas em redes de transporte e de telecomunicações, como o transporte de pessoas, produtos, materiais ou de dados, ocorrem em redes deste tipo. A Figura 1 mostra um exemplo de rede do tipo *hub-and-spoke*, onde os concentradores são representados por quadrados e os demais nós (denominados *spokes* ou nós de demanda) são representados por círculos.

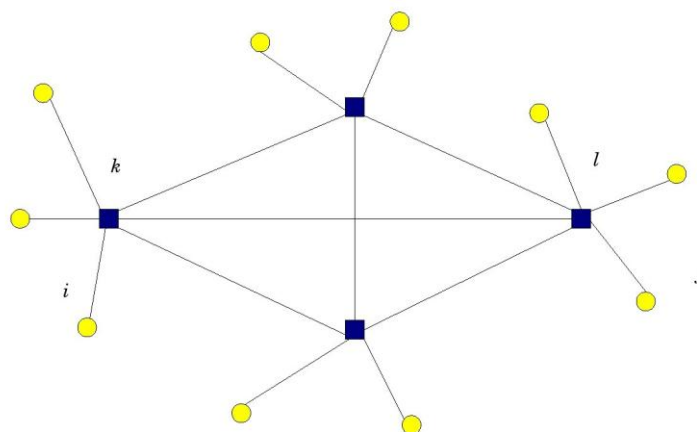


Figura 1 – Rede do tipo *hub-and-spoke* com  $n=200$

Existem diferentes versões do problema de localização de concentradores (*hubs*). Alguns casos podem apresentar restrições de capacidade (Aykin, 1994), ou seja, um limitante no volume de informações que um concentrador consegue transportar, ou ainda um custo fixo associado a cada concentrador, além dos custos de alocação dos nós de demanda da rede aos concentradores.

Quando não existe restrição quanto ao fluxo (de pessoas ou de dados, por exemplo) que passa por um concentrador e cada nó de demanda não pode ser alocado a mais que um concentrador, o problema denomina-se Problema Não-Capacitado de Localização de Concentradores (USAHLP - *Uncapacitated Single Allocation Hub Location Problem*). Neste problema o número de concentradores é uma variável de decisão. No caso do número de concentradores ser fixo (por exemplo, igual a  $p$ ), o problema é denominado de USApHMP (*Uncapacitated Single Allocation p-Hub Median Problem*) (Chen, 2008; Ebery, 2001). No entanto, quando um nó de demanda pode ser alocado a mais do que um concentrador, o problema denomina-se problema de localização de concentradores com alocação múltipla (do inglês,

*Uncapacitated Multiple Allocation Hub Location Problem, UMAHLP*) e quando existe restrição de capacidade quanto ao fluxo máximo de um concentrador, o problema é conhecido como problema de localização de concentradores capacitado (*Capacitated Single Allocation Hub Location Problem, CSAHLP*).

O problema de localização de concentradores pertence à classe de problemas NP-difíceis (Garey e Johnson, 1979). Portanto, a determinação de solução ótima para o problema por meio de algoritmos exatos, deve utilizar métodos enumerativos, que exigem grande esforço computacional e podem ser impraticáveis para exemplares do problema de grandes dimensões. Por isto, diversas propostas usando heurísticas têm sido consideradas, tais como: Busca Tabu (Kliniewicz, 1991), *Simulated Annealing* (Abdinnour-helm e Venkataramanan, 1993), Redes Neurais Artificiais (Smith et al, 1996) e Algoritmos Genéticos (Abdinnour, 1998; Topcuoglu et al, 2005; Cunha e Silva, 2007).

Ernst e Krishnamoorthy (1999) propõem dois algoritmos heurísticos para o problema capacitado: o primeiro baseia-se no método *simulated annealing* e o segundo em um método de descida randômico. Tal método consiste em gerar soluções vizinhas aleatoriamente e só aceitar movimentos de melhora. Com o apoio de um método do tipo *branch-and-bound* e com limitantes superiores providos pelas heurísticas os autores conseguiram obter soluções ótimas para problemas AP definidos em redes de até 50 nós, com exceção do teste em que  $n = 50$  e os custos fixos e a capacidade são do tipo apertados.

Almeida e Senne (2009) realizam um estudo comparativo de dois algoritmos híbridos com busca por agrupamentos (*Clustering Search, CS*) para resolver o USApHMP. Os testes realizados verificaram quais das metaheurísticas propostas se adéquam melhor ao CS.

Almeida e Senne (2010) apresentam três metaheurísticas: um algoritmo genético, um método *simulated annealing* e um método VNS para o problema de localização de concentradores com restrição de capacidade. Um estudo comparativo é realizado através do conjunto de dados AP.

Este trabalho propõe uma metaheurística *simulated annealing* capaz de solucionar o CSAHLP, e em seguida é proposto uma hibridização do método por meio da aplicação de busca por agrupamentos com o intuito de verificar o comportamento do CS.

## 2. Modelo Matemático

Goldman (1969) foi um dos primeiros a apresentar o problema de localização de concentradores. No entanto, o primeiro modelo matemático com função objetivo quadrática para o problema de localização de concentradores é devido a O'Kelly (1987). A formulação matemática para o problema capacitado pode ser escrita como:

$$f(x) = \text{Min.} \sum_i \sum_j w_{ij} \sum_k \lambda d_{ik} x_{ik} + \sum_i \sum_j w_{ji} \sum_l \delta d_{jl} x_{jl} + \sum_i \sum_k x_{ik} \sum_j \sum_l \alpha x_{jl} d_{kl} w_{ij} + \sum_k f_k x_{kk} \tag{2.1}$$

Sujeito à:

$$\sum_k x_{ik} = 1, \quad \forall i \in V \tag{2.2}$$

$$x_{kk} - x_{ik} \geq 0, \quad \forall i, k \in V \tag{2.3}$$

$$\sum_i E_i x_{ik} \leq Q_k x_{kk} \quad \forall k \in V \tag{2.4}$$

$$x_{ik} \in \{0,1\} \quad \forall i, k \in V. \tag{2.5}$$

Em que:

- $V$  é conjunto de nós da rede;
- $d_{ij}$  é a distância entre os nós  $i$  e  $j$ ;
- $w_{ij}$  é a quantidade de fluxo transferido entre os nós  $i$  e  $j$ ;
- $\lambda$ ,  $\alpha$ ,  $\delta$  são, respectivamente, os custos de coleta, transferência e distribuição;
- $Q_i$  é a capacidade do concentrador  $i$ ;
- $x_{ik}$  é uma variável de decisão, tal que  $x_{ik} = 1$  se o nó  $i$  está alocado ao concentrador  $k$  ( $x_{ik} = 0$  caso contrário). Deve-se observar que se  $x_{kk} = 1$  então o nó  $k$  é um concentrador; Caso contrário,  $x_{kk} = 0$ .
- $E_i$  é a soma dos fluxos que chegam em  $i$ . ( $E_i = \sum_j w_{ij}$ )

Nessa formulação, a função-objetivo (2.1) estabelece o custo total a ser minimizado que corresponde a soma dos custos de coleta transferência e distribuição de uma rede, a restrição, as restrições (2.2) garantem que cada nó de demanda será alocado a um único concentrador, as restrições (2.3) asseguram que as alocações serão feitas apenas para nós que são concentradores, as condições (2.4) restringem que o fluxo total que chega ao concentrador  $k$  não pode exceder a capacidade máxima de cada concentrador  $k$  e as restrições (2.5) correspondem às condições de integralidade das variáveis de decisão.

### 3. A Busca por Agrupamentos

O método de busca evolutiva por agrupamentos (ou ECS - *Evolutionary Clustering Search*), proposto por Oliveira e Lorena (2004, 2007), pode ser definida como uma metaheurística que se baseia no agrupamento (*cluster*) de soluções geradas por um determinado algoritmo e na busca local dentro dos *clusters* mais promissores.

Chaves (2009) propôs uma generalização do método ECS e, devido a isto, o nome da técnica foi simplificado para busca por agrupamentos (CS, *Clustering Search*). Na busca por agrupamentos, um *cluster*  $c$  é caracterizado por uma tripla  $c = (C, \gamma, r)$ , em que:  $C$  é a solução que representa o centro do *cluster*  $c$ ,  $\gamma$  representa a quantidade de soluções pertencentes ao *cluster*  $c$  e  $r$  é uma variável de controle que armazena o número de vezes consecutivas que a busca local foi aplicada ao *cluster*  $C$  e não melhorou a solução.

O método de busca por agrupamentos consiste basicamente de três componentes: um gerador de soluções factíveis, um processo de agrupamento e um método de busca local.

O gerador de soluções pode ser qualquer heurística ou metaheurística capaz de gerar soluções com diversidade. Sua execução não depende dos outros componentes. No entanto, o algoritmo gerador de soluções deve garantir que as soluções serão geradas continuamente para o processo de agrupamento.

Na Figura 2, ilustra-se em resumo o funcionamento do método de busca por agrupamentos.

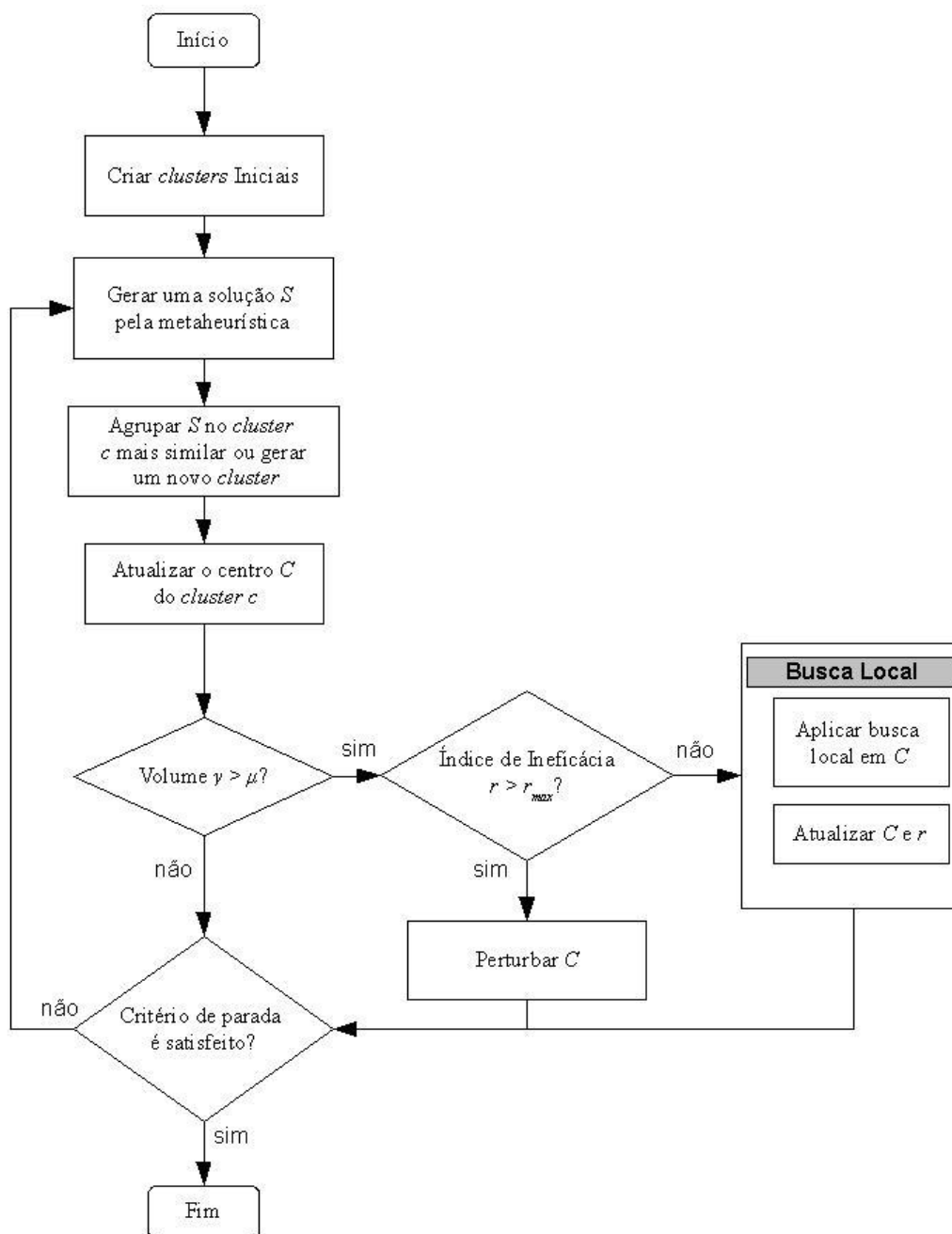


Figura 2 – Diagrama conceitual do CS (Chaves, 2009)

### 3.2. Metaheurística (SA)

O método *Simulated Annealing*, introduzido por Kirkpatrick *et al.* (1983), foi baseado no trabalho de Metropolis *et al.* (1953), nesta técnica a “temperatura” não é constante como no Metropolis. O processo consiste inicialmente em “fundir” o sistema a uma alta temperatura (este estado tem uma maior probabilidade de se aceitar soluções que pioram a função objetivo) e então, resfriar lentamente o sistema até que ele se “congele” e nenhuma mudança posterior possa ocorrer. Um pseudocódigo do método é apresentado pela Figura 3.

```

Procedimento Simulated Annealing
  Entrada:  $T_0$ ,  $T_f$ ,  $L$ ,  $\alpha$  (entre 0 e 1)
   $T \leftarrow T_0$ ;  $S_0$ 
  gera solução inicial;
   $S \leftarrow S_0$ ;
   $S^* \leftarrow S_0$ 
  enquanto  $T > T_f$  faça (temperatura alta)
    para  $cont \leftarrow 1$  até  $L(T)$  faça (iterações para equilíbrio)
       $S' \leftarrow$  seleciona uma solução vizinha de  $S$ 
       $D_{custo} \leftarrow custo(S') - custo(S)$ 
      se  $D_{custo} < 0$  ou  $U[0,1] < \exp(-D_{custo}/T)$ 
        então  $S \leftarrow S'$ 
        se  $(S < S^*)$  então  $S^* \leftarrow S$ 
      fim do para
       $T \leftarrow \alpha T$ 
    fim-enquanto
fim-procedimento

```

Figura 3 – Método *Simulated Annealing*

A idéia desta metaheurística consiste em simular o recozimento de metais. O processo começa com a temperatura inicial elevada e aos poucos a temperatura vai diminuindo de acordo com uma taxa de resfriamento. Essa característica permite que o algoritmo aceite diversos movimentos de piora no espaço de busca. À medida que a temperatura diminui, o método tende a se estabilizar em uma determinada região do espaço de busca.

### 3.3. Processo de Agrupamento

O processo de agrupamento tem como principal tarefa agrupar soluções similares dentro de um mesmo cluster e criar novos clusters, caso não exista um cluster similar a uma determinada solução. Com o objetivo de limitar o número de clusters a serem criados, definiu-se um limitante superior  $N_c$  para este número. Tal componente também é responsável por uma perturbação (assimilação) no centro de um cluster toda vez que uma nova solução é incluída neste cluster.

Para que este componente funcione adequadamente, é necessário estabelecer uma métrica de distância entre soluções. A métrica estabelecida para o CSAHLP corresponde ao número de alocações diferentes para os *hubs*. Assim, pode-se medir a distância entre uma dada solução e o centro (que também corresponde a uma solução) de um *cluster*. No processo de assimilação (agrupamento) utiliza-se o método *path-relinking* (Glover, 1996), que realiza movimentos exploratórios na trajetória que interconecta uma solução gerada pelo SM e o centro de um *cluster*.

Na Figura 4, ilustra-se o funcionamento do método *path-relinking* para o CSAHLP. Neste caso, tem-se uma rede com 4 nós. Cada solução é representada por um vetor  $v$  tal que  $v_i = 0$ , se o nó  $i$  corresponde a um *spoke* e  $v_i = 1$ , se o nó  $i$  corresponde ao um *hub*. Para a aplicação do método deve-se, inicialmente, gerar um conjunto de soluções vizinhas à solução inicial. Para o CSAHLP, as soluções vizinhas foram obtidas trocando-se um valor de  $v_i$  da solução inicial pelo correspondente  $v_i$  da solução guia. Com isto, foram obtidas 4 novas soluções, mostradas no primeiro nível da Figura 4. O método escolhe então uma dessas novas soluções, neste caso, a de menor custo, que corresponde à solução (1, 1, 0, 1). Para esta solução escolhida, aplica-se o mesmo procedimento de troca de elementos com a solução guia, gerando novas soluções. Para este exemplo, foram geradas 3 novas soluções, mostradas no segundo nível da Figura 4. Este procedimento prossegue até que a solução guia seja encontrada.

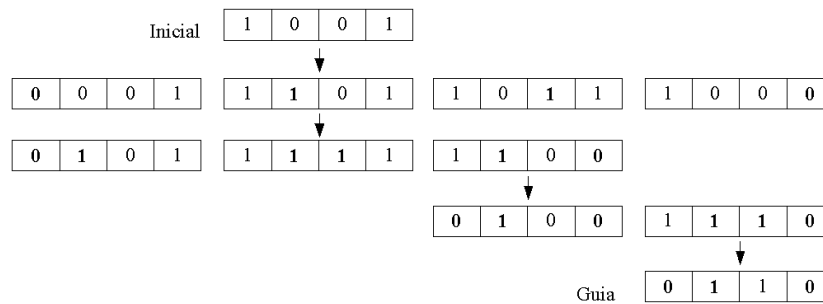


Figura 4 – Exemplo de *path-relinking* aplicado ao USAHLP

Com este método, define-se um “caminho” entre uma solução inicial e uma solução guia. A melhor solução encontrada em qualquer nível deste caminho é utilizada pelo componente IC para atualizar o centro do *cluster*.

### 3.4. Método de Busca Local

Heurísticas de busca local são métodos que partem de uma solução inicial viável, este método consiste basicamente na tentativa de melhorar tal solução por meio de operações de troca, remoção ou inserção, até que não seja mais possível a melhoria ou algum outro critério de parada seja satisfeito. A solução encontrada por esta heurística é considerada ótima local. Tal ótimo local pode, eventualmente, ser o ótimo global, ou seja, a melhor solução possível para o problema, mas não há garantias que isto seja uma verdade, como em todos os procedimentos heurísticos. O gráfico da Figura 5 mostra esta relação. A heurística de busca local inicia a partir de uma solução inicial, que é representada por um ponto no gráfico. A partir deste momento, através do critério de vizinhança adotado, ele procura soluções ao seu redor e, no caso do gráfico apresentado pela Figura 5, e reduz o custo da solução até o segundo ponto em destaque, o *ótimo local*.

Como é possível observar, todas as soluções vizinhas ao ótimo local possuem um custo superior à própria solução. Desta forma, o algoritmo para e exibe como resposta o ótimo local. Essa decisão faz o algoritmo perder todas as possibilidades que levariam o método a encontrar um *ótimo global*.

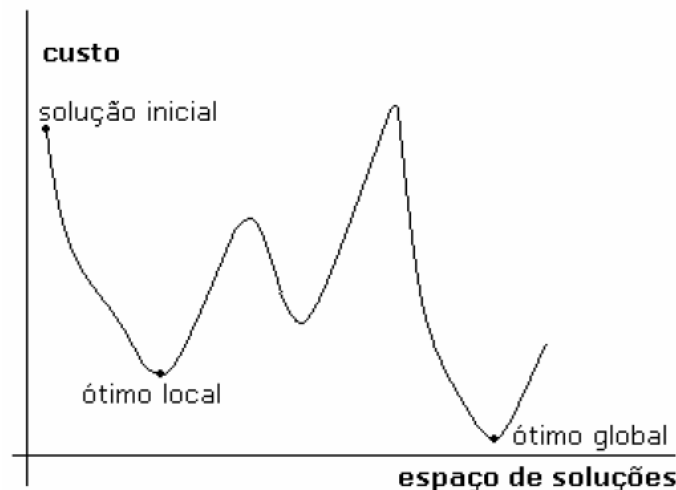


Figura 5 - Mínimos locais e globais

Um algoritmo de busca local básico denominado de método de subida/descida da encosta (do inglês, *hill climbing*) é apresentado na Figura 6.

<p><b>Procedimento</b> Descida</p> <p>Selecione a solução inicial <math>s</math></p> <p>Escolha o melhor vizinho <math>s' \in N(s)</math></p> <p><b>Enquanto</b> <math>s'</math> melhor que <math>s</math> <b>faça</b></p> <p>    <math>s \leftarrow s'</math></p> <p>    Escolha o melhor vizinho <math>s' \in N(s)</math></p> <p><b>fim-enquanto</b></p> <p><b>retorne</b> <math>s</math></p> <p><b>Fim-procedimento</b></p>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Figura 6 – Método de Descida

A implementação deste método requer a definição de uma vizinhança e a escolha de um método de busca local. No entanto um vizinho deve ser definido de acordo com a representação da solução.

Para o problema considerado neste trabalho, a representação das soluções (ou, indivíduos do AG) baseiam-se na proposta de Topcuoglu *et al.* (2005) e corresponde a dois vetores de tamanho  $n$ : um para armazenar a localização dos *hubs* e outro para armazenar as alocações dos *spokes* aos *hubs*. Nestes vetores, denominados *HubArray* e *AssignArray*, cada posição corresponde a um nó da rede. O *HubArray* corresponde a um vetor binário em que cada posição armazena o valor 0, no caso do nó correspondente a esta posição ser um *spoke*, ou 1, no caso deste nó ser um *hub*. O *AssignArray* equivale a um vetor em que cada posição armazena o índice do *hub*, ao qual o nó correspondente está associado.

As vizinhanças definida para este trabalho corresponde aos seguintes movimentos:

- *Swap Nodes*: reponsável pela escolha de dois nós não-hub para a troca de suas alocações;
- *Swap Hubs*: obtém dois concentradores e troca as associações de um concentrador com as associações de outro;
- *Reallocate Node*: escolhe um nó não hub e aloca a outro concentrador diferente de sua alocação original;
- *Reallocate Hubs*: pega um nó não hub e seu concentrador associado, e faz uma troca entre eles. Neste caso, o nó não hub passa a ser concentrador e o concentrador um nó não hub, com isso todas as associações ao antigo concentrador passa a se conectar ao concentrador atual.
- *New Hub*: escolhe um nó não hub e o transforma em um concentrador;
- *Delete Hub*: retira um concentrador da solução, e o transforma em um nó não hub. Os nós não hub alocados a este são realocados a outros concentradores de modo aleatório.

Em cada vizinhança definida o critério de escolha acontece de forma aleatória. Além disso, as soluções inviáveis em que o fluxo total ultrapassam a capacidade do concentrador são tratadas em todos os métodos aqui definidos através de penalização, por meio da atribuição de um custo elevado a solução. Com isso o método evita a escolha de soluções inviáveis.

#### 4. Testes e Resultados

A verificação da eficiência dos métodos implementados, basearam-se no conjunto de dados AP (Ernest e Krishnamoorthy, 1996). Este conjunto tem sido referenciado em diversos trabalhos, como em Sasaki e Fukushima (2003) e Chen (2007).



O conjunto de dados AP (*Australian Post*), proposto por Ernest e Krishnamoorthy (1996), derivado do fluxo de e-mails na Austrália, compõe um conjunto de testes que vão de 10 a 50 nós para exemplares pequenos, e de 100 a 200 nós para exemplares grandes. Os valores dos custos de coleta, transferência e distribuição são respectivamente 3.0, 0.75 e 2. Os fluxos não são simétricos, ou seja,  $w_{ij} \neq w_{ji}$ , além disso, um cliente (*spoke*) pode enviar um e-mail a si próprio ( $w_{ii} \neq 0$ ).

Para ambas as metaheurísticas implementadas, os testes foram executados em um computador com processador Core2-Duo 2.0 GHz, com 2 GB de memória RAM, sob o sistema operacional Linux.

Os parâmetros utilizados para SA e CSSA, foram os seguintes:

- Temperatura inicial: 100000;
- Taxa de resfriamento: 0,95;
- Critério de parada: 10000 execuções sem redução de custo da melhor solução corrente;
- Raio do cluster: 3;
- Número de clusters: 20.

Os testes apresentados nesta seção consideram que os custos fixos e a capacidade dividem-se em duas categorias: frouxos (L) e apertados (T). Para o CSAHLP cada método foi executado 10 vezes. Nestas tabelas são utilizadas as seguintes legendas para as colunas:

- **Ótimo** representa a solução ótima conhecida para o problema;
- **Sol** representa a melhor solução encontrada pelo método implementado;
- **TE** representa o tempo médio de execução total do algoritmo, até que o critério de parada seja alcançado;
- **Desv** corresponde à variação das soluções encontradas em relação à média;
- **Gap** corresponde ao erro da melhor solução encontrada pela metaheurística em relação à melhor solução encontrada na literatura;

Os valores de **Desv** e **Gap** foram calculados com o objetivo de analisar a robustez e eficácia dos métodos implementados. Os valores de *Desv* e *Gap* são calculados pelas equações (4.1) e (4.2), em que  $Sol_{média}$  corresponde à solução média, *Sol* corresponde à solução de menor custo encontrada pela meta-heurística nas dez execuções e *Melhor* equivale a solução ótima ou a solução de menor custo encontrada na literatura.

$$Desv = 100 \times \frac{(Sol_{média} - Sol)}{Sol} \quad (4.1)$$

$$Gap = 100 \times \frac{(Sol - Melhor)}{Melhor} \quad (4.2)$$

Para o cálculo da variação do CSSA em relação ao SA utilizou-se a seguinte equação:

$$Var = 100 \times \frac{(Sol_{CSSA} - Sol_{SA})}{Sol_{CSSA}} \quad (4.3)$$

Tabela 1 – Resultados SA × CSSA

Teste	Ótima	SA			TE	CSSA			TE	Var
		Sol	Gap	Desvio		Sol	Gap	Desvio		
ap10ll	224250,05	225979,17	0,01	0,00	0,03	224250,05	0,00	0,08	1,46	0,01
ap10lt	250992,26	264173,83	0,05	0,00	0,02	250992,26	0,00	0,00	1,34	0,05
ap10tl	263399,94	271332,74	0,03	1,69	0,02	263399,95	0,00	0,00	0,98	0,03
ap10tt	263399,94	272625,71	0,04	0,00	0,02	263399,95	0,00	0,00	1,28	0,04
ap20ll	234690,96	275146,38	0,17	0,00	0,01	234690,97	0,00	0,00	3,08	0,17
ap20lt	253517,40	253517,40	0,00	3,50	0,06	253517,40	0,00	0,00	3,96	0,00
ap20tl	271128,18	315568,06	0,16	0,00	0,02	271128,18	0,00	0,00	1,82	0,16
ap20tt	296035,40	329139,36	0,11	0,00	0,07	296035,42	0,00	0,25	5,18	0,11
ap25ll	238977,95	292025,93	0,22	0,00	0,02	238977,96	0,00	0,00	5,81	0,22
ap25lt	276372,50	276372,49	0,00	7,67	0,10	276372,49	0,00	0,05	8,21	0,00
ap25tl	310317,64	368528,29	0,19	0,00	0,02	310317,65	0,00	0,00	4,90	0,19
ap25tt	348369,15	352325,27	0,01	4,04	0,09	348369,16	0,00	0,21	8,05	0,01
ap40ll	241955,71	242114,04	0,00	1,14	0,23	241955,70	0,00	0,00	15,54	0,00
ap40lt	272218,32	287214,14	0,06	2,11	0,42	272218,32	0,00	0,00	22,46	0,06
ap40tl	298919,01	305552,00	0,02	6,13	0,20	298919,00	0,00	0,00	15,16	0,02
ap40tt	354874,10	356873,01	0,01	7,10	0,27	354874,10	0,00	1,20	33,93	0,01
ap50ll	238520,59	238520,58	0,00	0,00	0,34	238520,58	0,00	0,00	24,23	0,00
ap50lt	272897,49	286328,03	0,05	2,96	0,51	272897,49	0,00	0,38	48,37	0,05
ap50tl	319015,77	328189,44	0,03	1,25	0,34	319015,78	0,00	0,00	33,92	0,03
ap50tt	417440,99	419169,14	0,00	6,82	0,59	418215,07	0,00	0,84	61,35	0,00
Média			0,06	2,22	0,17		0,00	0,15	15,05	

Como pode ser analisado na Tabela 1 pela coluna “Var” a maior parte dos casos o método CSSA apresentou melhoras em relação ao SA. Outro casos importantes podem ser vistos através das colunas GAP e Desvio, nestes casos o CSSA mostra-se se mais eficiente e mais robusto na busca de soluções ótimas ou próximas da ótima. Pela Tabela 3 é possível verificar um resumo dos resultados da Tabela 1.

Tabela 2 – Resumo dos testes SA × CSSA

	SA	CSSA
Desvio médio total	2,22	0,15
% das soluções ótimas obtidas	25,00	100,00

## 5. Conclusões

O problema de localização de concentradores é um problema de Otimização Combinatória relevante, pois ocorre em diversas situações práticas em que o transporte de alguma entidade (pessoas, dados, produtos, etc) precisa passar por um processo de agregação antes de ser distribuída ao seu destino. Boas soluções para o problema podem representar ganhos econômicos significativos para muitos setores empresariais. A solução do problema consiste na determinação da quantidade necessária e na localização dos concentradores, além da alocação das demais instalações a estes concentradores, minimizando os custos envolvidos.

Pelos testes realizados pode-se concluir que a aplicação da busca por agrupamentos permite obter soluções de melhor qualidade, no entanto a escolha de uma boa metaheurística para trabalhar com esse método é de grande importância, neste trabalho analisou-se os resultados gerados por um método *simulated annealing* devido a suas simplicidades de entendimento e implementação.

Como sugestão para trabalhos futuros fica um estudo sobre problemas de cobertura de concentradores, neste problema o objetivo principal consiste em localizar concentradores de modo que cubra clientes em um determinado raio.

## Referências

- Abdnour-Helm, S.** (1998) A hybrid heuristic for the uncapacitated hub location problem. *European Journal of Operations Research*, v. 106, p. 489-499.
- Abdnour-Helm, S.; Venkataramanan, M.A.** (1993) *Using simulated annealing to solve the p-hub location problem*. In: IRMIS Working paper, Decision and Information Systems Department, School of Business.
- Almeida, W. G. ; Senne, E. L. F.** (2009) Meta-heurísticas para o problema não-capacitado de localização de p-concentradores. In: XL Simpósio Brasileiro de Pesquisa Operacional. Porto Seguro, BH. Anais do XLI SBPO. Rio de Janeiro, RJ : SOBRAPO. p. 2001-2012.
- Almeida, W. G. ; Senne, E. L. F.** (2010) Soluções meta-heurísticas para o problema de localização de concentradores com restrições de capacidade. In: Encontro Mineiro de Engenharia de Produção, Coronel Fabriciano - MG. Anais do EMEPRO.
- Aykin, T.** (1994) Lagrangian relaxation based approaches to capacitated hub-and-spoke network design problem. *European Journal of Operational Research*, v. 79, n. 33, p. 501-523.
- Chaves, A. A.** (2009) *Meta-heurística híbrida com busca por agrupamentos aplicada a problemas de otimização combinatória*. Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 197 p.
- Chen, J.F.** (2007) A hybrid heuristic for the uncapacitated hub location problem. *Omega, The International Journal of Management Science*, v. 35, p. 211-220.
- Chen, J.F.** (2008) A note on solution of the uncapacitated single allocation p-hub median problem. *Journal of Chinese Institute of Industrial Engineers*, v. 25, p. 11-17.
- Cunha, C.B.; Silva, M.R.** (2007) A genetic algorithm for the problem of configuring a hub-and-spoke network for a LTL trucking company in Brazil. *European Journal of Operational Research*, v. 179, p. 747-758.
- Ebery, J.** (2001). Solving Large Single Allocation p-hub problems with two or three hubs. *European Journal of Operational Research*, v. 128, p. 447-458.
- Ernest, A.; Krishnamoorthy, M.** (1996) Efficient algorithms for the uncapacitated single allocation p-hub median problem. *Location Science*, v. 4, n. 3, p. 139-154.
- Ernst, A.T.; Krishnamoorthy, M.** (1999) Solution algorithms for the capacitated single allocation hub location problem, *Annals of Or*, v. 86, p. 141-159.
- Garey, M. R.; Johnson, D. S.** (1979) *Computers and Intractability: A Guide to the Theory of NP-Completeness*, San Francisco: W.H. Freeman.
- Glover, F.** (1996) Tabu Search and adaptative memory programming: Advances, applications and challenges. *Interfaces in Computer Science and Operational Research*, Kluwer, p. 1-75.

- Goldman, A.J.** (1969) Optimal location for centers in a network. *Transportation Science*, v. 3, p. 352-360.
- Kirkpatrick, S.; Gelatt Junior, C.D.; Vecchi, M.P.** (1983) Optimization by Simulated Annealing. *Science*, New York, v.220, p.671-680, 1983.
- Klincewicz, J.** (1991) Heuristics for the p-hub median problem. *European Journal of Operations Research*, v. 79, p. 25-37.
- Metropolis, N.; Rosenbluth, A. W.; Rosenbluth, M.; Teller, A. H.; and Teller, E.** (1953) Equation of state calculations by fast computing machines. *J. Chem. Phys.* **21**, 1087-1092.
- Oliveira, A.C.M.; Lorena, L.A.N.** (2004) Detecting promising areas by evolutionary clustering search. *Advances in Artificial Intelligence*, Springer Lecture Notes in Artificial Intelligence Series, p. 193-207.
- Oliveira, A.C.M.; Lorena, L.A.N.** (2007) *Hybrid Evolutionary Algorithms and Clustering Search*. Hybrid Evolutionary Algorithms, Studies in Computational Intelligence, Volume 75, Springer Berlin, p. 77-99.
- O’Kelly, M.** (1987) A quadratic integer program for the location of interacting hub facilities. *European Journal of Operational Research* , v. 32, p. 393-404.
- Sasaki, M.; Fukushima, M.** (2003) On the hub-and-spoke model with arc capacity constraints. *Journal of the Operations Research Society of Japan*, v. 46, n. 4, p. 409-428.
- Smith, K.; Krishnamoorthy, M.; Palaniswami, M.** (1996) Neural versus traditional approaches to the location of interacting hub facilities. *Location Science (Special Issue on Hub Location)*, v. 4, n. 3, p. 155-171.
- Topcuoglu, H.; Corut, F.; Ermis, M.; Yilmaz, G.** (2005) Solving the uncapacitated hub location problem using genetic algorithms. *Computers and Operations Research*, v. 32, p. 967-984.