



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO  
**INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS**

sid.inpe.br/mtc-m21b/2014/07.19.19.02-TDI

## **HEURÍSTICA BASEADA EM MODELO PARA PROBLEMAS DE LOCALIZAÇÃO DE CONCENTRADORES CAPACITADOS**

Wesley Gomes de Almeida

Tese de Doutorado do Curso de Pós-Graduação em Computação Aplicada, orientada pelos Drs. Edson Luiz França Senne, e Horacio Hideki Yanasse, aprovada em 04 de agosto de 2014.

URL do documento original:

<<http://urlib.net/8JMKD3MGP5W34M/3GMA3NP>>

INPE  
São José dos Campos  
2014

**PUBLICADO POR:**

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GB)

Serviço de Informação e Documentação (SID)

Caixa Postal 515 - CEP 12.245-970

São José dos Campos - SP - Brasil

Tel.:(012) 3208-6923/6921

Fax: (012) 3208-6919

E-mail: pubtc@sid.inpe.br

**CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO DA PRODUÇÃO INTELLECTUAL DO INPE (RE/DIR-204):****Presidente:**

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

**Membros:**

Dr. Gerald Jean Francis Banon - Coordenação Observação da Terra (OBT)

Dr. Amauri Silva Montes - Coordenação Engenharia e Tecnologia Espaciais (ETE)

Dr. André de Castro Milone - Coordenação Ciências Espaciais e Atmosféricas (CEA)

Dr. Joaquim José Barroso de Castro - Centro de Tecnologias Espaciais (CTE)

Dr. Manoel Alonso Gan - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

Dr<sup>a</sup> Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação

Dr. Plínio Carlos Alvalá - Centro de Ciência do Sistema Terrestre (CST)

**BIBLIOTECA DIGITAL:**

Dr. Gerald Jean Francis Banon - Coordenação de Observação da Terra (OBT)

**REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:**

Maria Tereza Smith de Brito - Serviço de Informação e Documentação (SID)

Yolanda Ribeiro da Silva Souza - Serviço de Informação e Documentação (SID)

**EDITORAÇÃO ELETRÔNICA:**

Maria Tereza Smith de Brito - Serviço de Informação e Documentação (SID)

André Luis Dias Fernandes - Serviço de Informação e Documentação (SID)



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO  
**INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS**

sid.inpe.br/mtc-m21b/2014/07.19.19.02-TDI

## **HEURÍSTICA BASEADA EM MODELO PARA PROBLEMAS DE LOCALIZAÇÃO DE CONCENTRADORES CAPACITADOS**

Wesley Gomes de Almeida

Tese de Doutorado do Curso de Pós-Graduação em Computação Aplicada, orientada pelos Drs. Edson Luiz França Senne, e Horacio Hideki Yanasse, aprovada em 04 de agosto de 2014.

URL do documento original:

<<http://urlib.net/8JMKD3MGP5W34M/3GMA3NP>>

INPE  
São José dos Campos  
2014

Dados Internacionais de Catalogação na Publicação (CIP)

---

Almeida, Wesley Gomes de.

Al64h      Heurística baseada em modelo para problemas de localização de concentradores capacitados / Wesley Gomes de Almeida. – São José dos Campos : INPE, 2014.

xxiv + 65 p. ; (sid.inpe.br/mtc-m21b/2014/07.19.19.02-TDI)

Tese (Doutorado em Computação Aplicada) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2014.

Orientadores : Drs. Edson Luiz França Senne, e Horacio Hideki Yanasse.

1. Localização de concentradores. 2. Local branching. 3. Heurística baseada em modelo. 4. Metaheurística. 5. Otimização combinatória. I.Título.

CDU 004.023

---



Esta obra foi licenciada sob uma Licença [Creative Commons Atribuição-NãoComercial 3.0 Não Adaptada](https://creativecommons.org/licenses/by-nc/3.0/).

This work is licensed under a [Creative Commons Attribution-NonCommercial 3.0 Unported License](https://creativecommons.org/licenses/by-nc/3.0/).

Aprovado (a) pela Banca Examinadora  
em cumprimento ao requisito exigido para  
obtenção do Título de **Doutor(a)** em  
**Computação Aplicada**

Dr. Solon Venâncio de Carvalho



Presidente / INPE / SJC Campos - SP

Dr. Edson Luiz França Senne



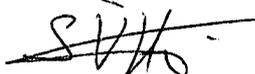
Orientador(a) / UNESP/GUARA / Guaratinguetá - SP

Dr. Horacio Hideki Yanasse



Orientador(a) / INPE / SJC Campos - SP

Dr. Stephan Stephany



Membro da Banca / INPE / SJC Campos - SP

Dr. Antônio Augusto Chaves



Convidado(a) / UNIFESP / São José dos Campos - SP

Dr. André Gustavo dos Santos



Convidado(a) / UFV / Minas Gerais - MG

Este trabalho foi aprovado por:

( ) maioria simples

unanimidade

Aluno (a): **Wesley Gomes de Almeida**

São José dos Campos, 04 de Agosto de 2014



*“O sofrimento se torna a maior das alegrias, quando a gente o busca como o mais precioso dos tesouros”.*

*Sta. Terezinha do Menino Jesus (1873-1897)*



*À minha esposa Angélica  
E a meu filho Wesley Gabriel.*



## **AGRADECIMENTOS**

Agradeço a Deus, por todas as bênçãos derramadas em minha vida durante este trabalho, à mãe Aparecida por todas as graças. Agradeço, também, à Sta. Teresinha do Menino Jesus e a todos os santos pela intercessão e por todas as graças pelas quais tornaram viável a realização deste trabalho.

A minha esposa Angélica que sempre me apoiou nos momentos de alegria e nos momentos difíceis. A meus pais e a meu sogro e minha sogra, por todo incentivo para a realização deste curso. Agradeço, também, a meu filho Wesley Gabriel pelas alegrias e os momentos de descontração que ele trouxe para minha vida nesses dois últimos anos de doutorado do curso de Computação Aplicada.

Ao Prof. Dr. Edson Luiz França Senne e ao Prof. Dr. Horacio Hideki Yanasse, pela paciência que tiveram comigo, pelos conhecimentos transmitidos e pela competência, dedicação e amizade demonstrados o tempo todo.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - CAPES, pelo auxílio financeiro.

Agradeço, também, ao Prof. Dr. Luiz Antônio Nogueira Lorena pela amizade e pelos conhecimentos passados em sua disciplina. Ao Prof. Dr. Stephan Stephany pelos conselhos sobre trancamento. Aos docentes do curso de Computação Aplicada - CAP, pesquisadores do Laboratório Associado de Computação e Matemática Aplicada - LAC, e colegas de curso.

Em memória do Prof. Dr. José Demísio Simões, agradeço muito pelas suas contribuições e sugestões.

A meu irmão e colega de curso Wanderson, por todo auxílio quando precisei para realizar matrícula e não podia comparecer ao INPE.

A todas as secretárias da CAP, pela dedicação com que atenderam cada solicitação durante esses anos que se passaram. Ao Instituto Nacional de Pesquisas Espaciais - INPE, pela oportunidade de estudos e utilização de suas instalações.

Aos professores e colegas do Instituto Federal de Minas Gerais – IFMG campus São João Evangelista-MG, pelo apoio e permissão de afastamento de minhas atividades para conclusão deste trabalho.

E a todos que de forma direta ou indiretamente contribuíram para o crescimento e a realização deste trabalho.

## RESUMO

Este trabalho tem como objetivo propor uma estratégia de solução para problemas de localização de concentradores com restrições de capacidade. A técnica denominada *Local Branching* (LB) foi aplicada para o desenvolvimento desta estratégia de solução. Tal técnica baseia-se em um modelo de Programação Matemática e consiste de uma heurística de melhoramento que utiliza a estratégia *branch-and-cut*, mas incorpora ideias presentes em técnicas de busca local e metaheurísticas. Na técnica LB, a busca por soluções inicia-se com uma solução de referência, que pode ser obtida por uma metaheurística. O método alterna-se entre ramificações estratégicas para definir vizinhanças de solução e ramificações táticas para explorar estas vizinhanças. Para obter as soluções de referência, foram desenvolvidas quatro metaheurísticas. Os resultados obtidos por estas metaheurísticas foram comparados e a de melhor desempenho foi utilizada como geradora da solução inicial de referência para o método LB. Neste trabalho apresentam-se estudos comparativos de resultados obtidos com a estratégia LB proposta e resultados obtidos pelo *solver* CPLEX com e sem a utilização de solução inicial, aplicados a problemas de localização de concentradores capacitados com alocação simples e múltipla.



# **MODEL-BASED HEURISTICS FOR CAPACITATED HUB LOCATION PROBLEMS**

## **ABSTRACT**

This work aims to propose a solution strategy for hub location problems with capacity constraints. A technique called Local Branching (LB) was applied to the development of this solution strategy. This technique is based on a Mathematical Programming model and consists of an improvement heuristic that uses the branch-and-cut strategy, but incorporates ideas present in local search techniques and metaheuristics. In the LB technique, the search for solutions begins with a reference solution which can be obtained by a metaheuristic. The method alternates between strategic ramifications for defining neighborhoods solution branches and tactical ramifications for exploring these neighborhoods. In order to generate the reference solutions, four metaheuristics were developed. The results obtained by these metaheuristics were compared and the best performance metaheuristic was used as a generator of the initial reference solution for the LB method. Comparative studies with the proposed LB strategy and the solver CPLEX with and without the use of initial solution applied to hub location problems with single and multiple allocation are presented.



## LISTA DE FIGURAS

	<u>Pág.</u>
Figura 1.1– Rede do tipo <i>hub-and-spoke</i> .....	2
Figura 2.1– Exemplo de rede com alocação única e múltipla, Fonte: Ernst e Krishnamoorthy (1998) .....	17
Figura 3.1 – Algoritmo R&F .....	23
Figura 3.2 – Algoritmo F&O .....	25
Figura 3.3– Árvore de enumeração LB .....	28
Figura 3.4– Algoritmo <i>Local Branching</i> .....	30
Figura 3.5– Algoritmo VNSM .....	26
Figura 4.1– Algoritmo do método AG .....	32
Figura 4.2– Algoritmo do método SA .....	33
Figura 4.3– Algoritmo do método VNS.....	33
Figura 4.4 – Diagrama conceitual do CS, Fonte: Chaves (2009) .....	35
Figura 4.5 – Exemplo de <i>path-relinking</i> aplicado ao CSAHLP .....	37
Figura 4.6 – Algoritmo do método MD .....	37
Figura 4.7 – Representação de uma solução CSAHLP .....	38
Figura 4.8 – Solução visual para o CSAHLP .....	39
Figura 4.9 – Algoritmo do método VND .....	41
Figura 4.10 – Representação de uma solução CMAHLP .....	43
Figura 4.11 – Algoritmo para obtenção do menor custo $C_{ijkl}$ .....	44



## LISTA DE TABELAS

	<b><u>Pág.</u></b>
Tabela 1.1 – Tipos de problemas de localização de concentradores.....	8
Tabela 5.1 – Resultados das heurísticas candidatas .....	46
Tabela 5.2 – Resultados obtidos para o PLCC com alocação única.....	49
Tabela 5.3 – Detalhes dos resultados obtidos pelos métodos CPLEX-CS e LB- CS .....	50
Tabela 5.4 – Resultados dos métodos LB-CS e CPLEX-CS para exemplares maiores .....	51
Tabela 5.5 – Resultados obtidos com a heurística CSSA .....	52
Tabela 5.6 – Resultados obtidos para o PLCC com alocação múltipla.....	53
Tabela 5.7 – Resultados para exemplares maiores do PLCC com alocação múltipla.....	54



## LISTA DE SIGLAS E ABREVIATURAS

AG	Algoritmo Genético
AP	<i>Australian Post</i>
CAB	<i>Civil Aeronautics Board</i>
CMAHLP	<i>Capacitated Multiple Allocation Hub Location Problem</i>
CMAHLP-F	Modelo matemático linear para o problema CMAHLP
CPLEX	Software resolvidor de modelo de programação linear, inteira e mista - versão 12.6
CS	<i>Clustering Search Simulated Annealing</i>
CSSA	<i>Clustering Search Simulated</i>
CSAHLF	<i>Capacitated Single Allocation Hub Location Problem</i>
ECS	<i>Evolutionary Clustering Search</i>
HCoP	<i>Hub Covering Problem</i>
LB	<i>Local Branching</i>
PLC	Problema de Localização de Concentradores
SATL	<i>Simulated Annealing Tabu List</i>
UMAHLP	<i>Uncapacitated Multiple Allocation Hub Location Problem</i>
USAHLP	<i>Uncapacitated Single Allocation Hub Location Problem</i>
USApHMP	<i>Uncapacitated Single Allocation p-Hub Median Problem</i>



## LISTA DE SÍMBOLOS

$\alpha$	Custo de transferência entre concentradores
$\beta$	Custo crítico máximo para um caminho entre um par origem-destino
$\lambda$	Custo de coleta entre um nó de origem e seu concentrador
$\delta$	Custo de distribuição entre um concentrador e seu destino
$C_{ijkl}$	Custo total entre dois clientes $i$ e $j$ passando pelos concentradores $k$ e $l$ .
$d_{ij}$	Distância entre os nós $i$ e $j$
$E_i$	Quantidade de fluxo que chega no ponto $i$
$f_k$	Custo fixo para que um <i>hub</i> seja localizado no ponto $k$
$H_k$	Variável de decisão que define se o ponto $k$ é um concentrador
$n$	Número de nós da rede
$p$	Número de concentradores a serem localizados
$Q_k$	Capacidade do concentrador $k$
$S_i$	Quantidade de fluxo que deixa o ponto $i$
$V$	Conjunto de nós da rede
$w_{ij}$	Quantidade de fluxo transferido entre os nós $i$ e $j$
$x_{ik}$	Variável de decisão tal que $x_{ik} = 1$ se o nó $i$ está alocado ao concentrador $k$ ( $x_{ik} = 0$ , caso contrário). Se $x_{kk} = 1$ , significa que o nó $k$ é um concentrador, caso contrário, $x_{kk} = 0$
$x'_{ij}$	Quantidade de fluxo com origem em $i$ e destino em $j$ que passam pelo concentrador $l$
$Y'_{kl}$	Quantidade de fluxo transferido entre os concentradores $k$ e $l$ originados a partir do nó $i$



## SUMÁRIO

	<u>Pág.</u>
<b>1</b>	<b>INTRODUÇÃO ..... 1</b>
1.1.	Contextualização do Problema de Localização de Concentradores 1
1.2.	Objetivos e Delimitações do Trabalho ..... 4
1.3.	Justificativas e Contribuições do Trabalho ..... 5
1.4.	Organização do Trabalho ..... 5
<b>2</b>	<b>PROBLEMAS DE LOCALIZAÇÃO DE CONCENTRADORES ..... 7</b>
2.1.	Revisão da Literatura ..... 9
2.2.	Problema de Localização de Concentradores Capacitado com Alocação Única ..... 15
2.3.	Problema de Localização de Concentradores Capacitado com Alocação Múltipla ..... 16
<b>3</b>	<b>HEURÍSTICAS BASEADAS EM MODELO ..... 21</b>
3.1.	O Método <i>Relax-and-Fix</i> ..... 22
3.2.	O Método <i>Fix-and-Optimize</i> ..... 24
3.3.	O Método LB Associado ao Modelo VNS..... 25
3.4.	O Método <i>Local Branching</i> ..... 26
<b>4</b>	<b>HEURÍSTICAS BASEADAS EM MODELO PARA O PLCC ..... 31</b>
4.1.	Heurísticas Candidatas para a Geração de Soluções de Referência ..... 31
4.2.	Método de Solução para o Problema de Localização de Concentradores Capacitado com Alocação Única ..... 38
4.3.	Método de Solução para o Problema de Localização de Concentradores Capacitado com Alocação Múltipla ..... 41

<b>5</b>	<b>RESULTADOS COMPUTACIONAIS</b> .....	45
5.1.	Resultados para o Problema de Localização de Concentradores Capacitado com Alocação Única (CSAHL P).....	48
5.2.	Resultados para o Problema de Localização de Concentradores Capacitado com Alocação Múltipla (CMAHL P) .....	51
<b>6</b>	<b>CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS</b> .....	55
	<b>REFERÊNCIAS BIBLIOGRÁFICAS</b> .....	59

# 1 INTRODUÇÃO

Neste capítulo é definida a contextualização dos problemas de Localização de Concentradores estudados e são apresentados os objetivos e delimitações do trabalho. Em seguida são mostradas as justificativas, contribuições e a estrutura organizacional deste trabalho.

## 1.1. Contextualização do Problema de Localização de Concentradores

Em alguns problemas de comunicação definidos em redes, a comunicação entre os nós acontece por meio 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.

O Problema de Localização de Concentradores (PLC) em uma rede consiste em determinar o número de instalações de consolidação (concentradores), a localização de cada um dos concentradores e a alocação dos demais nós da rede (denominados como *spokes*, ou nós de demanda) aos concentradores, de

forma a minimizar o custo total de operação, que pode incluir os custos variáveis de transporte e os custos fixos de localização das instalações. Essas instalações podem ser fábricas, portos, pontos de venda de produtos, armazéns, postos de serviço de rotina ou de emergência, postos de correio, pontos de incineração de lixo, centros de atendimento médico, aeroportos, antenas de comunicação, escolas, bibliotecas, dentre muitas outras. Quando o PLC possui uma restrição de capacidade este problema é denominado Problema de Localização de Concentradores Capacitado (PLCC).

A Figura 1.1 mostra um exemplo de rede do tipo *hub-and-spoke*, em que os concentradores (representados por quadrados) atendem aos nós de demanda (representados por círculos). Um fluxo  $w_{ij}$  com origem no nó  $i$  e destino no nó  $j$ , é transportado da seguinte forma: inicialmente, a carga de  $i$  é enviada para o nó  $k$ , onde é consolidada com outros fluxos de origens diferentes, e enviada para o concentrador  $l$ . A partir daí, os fluxos são distribuídos para os nós atendidos por  $l$ , inclusive  $j$ .

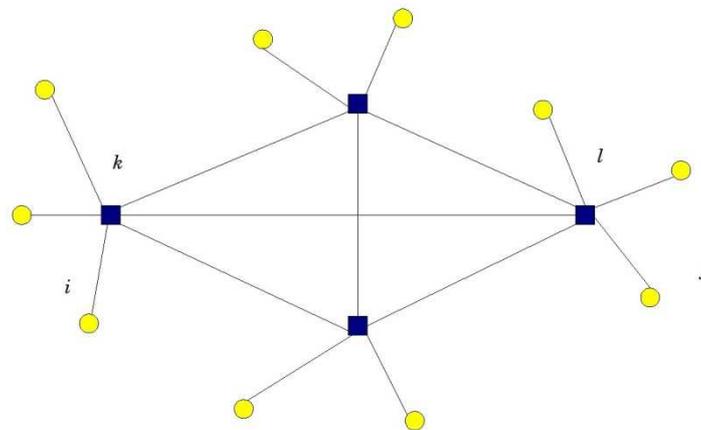


Figura 1.1– Rede do tipo *hub-and-spoke*

Dado o fluxo entre cada par de nós origem-destino, e os custos fixos de abertura das instalações dos concentradores, o problema de localização de concentradores (PLC) consiste em minimizar o custo total da rede, que inclui o

custo de transporte de alguma entidade entre os nós de demanda. A solução do problema busca encontrar os nós que devem se tornar concentradores e a alocação dos demais nós a estes concentradores.

Neste trabalho pretende-se restringir o estudo a problemas de localização de concentradores capacitados para dois casos: problemas com alocação única e problemas com alocação múltipla. No problema com alocação múltipla um cliente pode ser associado a mais que um concentrador (*hub*) enquanto que no problema com alocação única o modelo matemático restringe esta associação a um único concentrador.

Para o desenvolvimento deste trabalho estuda-se uma técnica denominada *Local Branching* (LB), que foi proposta por Fischetti e Lodi (2003). A ideia principal desta técnica consiste em buscar soluções cada vez melhores a partir de uma solução inicial de referência, de modo que se o método for interrompido em algum momento do processo de busca a estratégia funciona como uma heurística.

Para um problema de Otimização Combinatória em que se pretende obter soluções de custo mínimo, como no caso do PLCC, os métodos exatos exploram todas as possibilidades de solução, por meio de uma árvore de busca e, ao final, são capazes de exibir a solução ótima do problema, ou seja, a solução de menor custo. Isto, em geral, dada a natureza combinatória do problema, exige um grande esforço computacional, demandando grande consumo de memória e/ou alto tempo de processamento. Uma heurística pode ser definida com uma estratégia de solução capaz de solucionar rapidamente um problema difícil, empregando algum grau de aleatoriedade para encontrar as melhores soluções, mas sem garantir que a solução encontrada é uma solução ótima. Uma metaheurística pode ser considerada como uma estratégia de solução mais ampla. Uma metaheurística explora o espaço de soluções de um problema utilizando uma heurística específica para aquele problema e emprega estratégias inteligentes para escapar de mínimos locais. Esta

estratégia de solução, geralmente, se aplica a uma grande variedade de problemas.

A técnica LB tem o espírito dos métodos exatos, explorando uma árvore de busca. O algoritmo LB aqui proposto, no entanto, funciona como uma heurística, pois não pretende explorar toda a árvore de busca. Durante o processo de busca, uma solução é explorada dentro de uma vizinhança obtida através de uma restrição *local branching* que é inserida no modelo toda vez que uma nova solução é encontrada. Com o uso dessas ramificações a estratégia funciona como um método do tipo *branch-and-cut* (PADBERG e RINALDI, 1987; PADBERG e RINALDI, 1991). No entanto, usa uma solução inicial de referência para acelerar a busca por soluções ótimas e explorar vizinhanças de solução.

O método *branch-and-cut* (PADBERG e RINALDI (1987, 1991) é uma especialização da técnica *branch-and-bound* (B&B), pois seu funcionamento baseia-se na mesma lógica do método B&B, mas com a adição de cortes em cada nó da árvore de ramificação, capazes de gerar limitantes mais precisos antes de podá-los e ramificá-los.

A estratégia *Local Branching* (LB) define subproblemas através da adição de cortes *local branching* isolando uma vizinhança de uma determinada solução factível. Ao isolar essas regiões, o objetivo consiste em melhorar as soluções factíveis antes de continuar o processo comum do método *branch-and-cut*.

## **1.2. Objetivos e Delimitações do Trabalho**

Em geral, um método do tipo *branch-and-cut* requer um estudo aprofundado das propriedades topológicas do espaço de soluções do problema, para utilizar cortes que sejam efetivos. Assim, os cortes são específicos para cada problema a ser resolvido. Existem, no entanto, vários métodos de planos de corte que são genéricos, como por exemplo, os cortes de Gomory (1958).

O objetivo deste trabalho consiste em demonstrar que o método *Local Branching* com cortes genéricos é efetivo para os problemas de localização de concentradores.

### **1.3. Justificativas e Contribuições do Trabalho**

Até o momento não foram encontradas na literatura trabalhos que utilizam a técnica LB a problemas de localização de concentradores. A principal contribuição, aqui proposta é mostrar que a utilização de restrições *local branching* em métodos baseados em modelos de Programação Matemática para problemas de localização de concentradores capacitados é efetivo. O método proposto utiliza algumas metaheurísticas para a determinação de bons limitantes superiores e, com isso, acelerar a busca por soluções.

### **1.4. Organização do Trabalho**

Este trabalho está organizado da forma descrita a seguir.

O Capítulo 2 contém uma revisão bibliográfica sobre o problema em estudo, em que são apresentados conceitos e definições básicas para o desenvolvimento dos próximos capítulos, além dos tipos de problemas de localização de concentradores a serem abordados.

No decorrer do Capítulo 3 são apresentados os principais conceitos sobre heurísticas baseadas em modelo (também conhecidas como *matheuristics*), e também são apresentadas algumas dessas heurísticas já existentes.

O Capítulo 4 detalha os métodos implementados para os problemas de localização de concentradores capacitados com alocação única e múltipla e apresenta também algumas metaheurísticas bem conhecidas, selecionadas para trabalhar com as estratégias propostas.

O Capítulo 5 mostra um estudo comparativo entre as metaheurísticas estudadas para a solução do problema de localização de concentradores com

restrições de capacidade. Em seguida são apresentadas comparações entre as soluções obtidas pelas técnicas propostas e pelo software CPLEX (IBM ILOG, 2013).

Por fim, as conclusões e sugestões para trabalhos futuros são apresentadas no Capítulo 6.

## 2 PROBLEMAS DE LOCALIZAÇÃO DE CONCENTRADORES

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 com Alocação Única (do inglês, *Uncapacitated Single Allocation Hub Location Problem*, USAHLP). 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 USA $p$ HMP (do inglês, *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), para o caso em que um cliente só pode ser alocado a um concentrador e capacitado com alocação múltipla (*Capacitated Multiple Allocation Hub Location Problem*, CMAHLP) quando um cliente pode ser alocado a mais que um concentrador. A Tabela 1.1 apresenta as principais diferenças entre os diferentes tipos de problemas apresentados.

Tabela 1.1 – Tipos de problemas de localização de concentradores

<b>Problema</b>	<b>Critério</b>	<b>Número de concentradores</b>	<b>Tipo de Alocação</b>	<b>Capacidade do Concentrador</b>	<b>Custo de Localizar um Concentrador</b>
USAHLP	Minimizar	Endógeno	Única	Ilimitado	Sim
CSAHLP	Minimizar	Endógeno	Única	Limitado	Sim
UMAHLP	Minimizar	Endógeno	Múltipla	Ilimitado	Sim
CMAHLP	Minimizar	Endógeno	Múltipla	Limitado	Sim
USApHMP	Minimizar	Exógeno	Única	Ilimitado	Não

Na Tabela 1.1 todos os problemas possuem como função-objetivo a minimização dos custos de transporte da rede. Quanto ao número de concentradores, em alguns casos o número de concentradores é uma variável de decisão (problema endógeno) e, em outros casos, o número de concentradores é pré-estabelecido (problema exógeno). O tipo de alocação pode ser única ou múltipla, quanto à quantidade de concentradores aos quais um cliente pode ser alocado. Quanto à capacidade dos concentradores, em alguns problemas os concentradores possuem limitação de capacidade, e em outros casos os concentradores são ilimitados quanto à capacidade. Nos problemas em que o número de concentradores são pré-determinados (exógeno), normalmente não são determinados custos de abertura de uma instalação, como no caso do USApHMP.

Outra versão do problema de localização de concentradores é o problema de cobertura (*Hub Covering Problem*, HCoP) em que o objetivo é determinar quantas instalações devem ser abertas e a localização de cada uma delas de modo que o maior caminho entre dois nós de demanda não exceda um determinado raio de cobertura (HAMACHER e MEYER, 2006).

Na Seção 2.1 serão apresentados os principais trabalhos da literatura relacionados aos problemas de localização de concentradores. Entre esses, são citados os principais modelos matemáticos e heurísticas apresentadas

para os casos em que não existe restrição de capacidade e a alocação só é permitida a um único concentrador.

Já nas Seções 2.2 e 2.3 são apresentados, de uma forma mais específica, os principais métodos e modelos para problemas capacitados, com alocação única e múltipla, respectivamente.

## **2.1. Revisão da Literatura**

O'Kelly (1987) apresentou o conjunto de testes CAB (*Civil Aeronautics Board*) para o problema de localização de concentradores. Neste conjunto, as instâncias estão baseadas no fluxo aéreo de passageiros entre as 25 maiores cidades dos Estados Unidos no ano de 1970. Os dados de teste disponíveis neste conjunto se agrupam em problemas definidos em redes de tamanhos  $n = \{10, 15, 20, 25\}$ .

Ernst e Krishnamoorthy (1996) tornaram disponível o conjunto AP (*Australian Post*), derivado do fluxo de entrega postal de Sydney, Austrália. Este conjunto consiste de problemas definidos em redes de até 200 nós que representam distritos postais. Neste conjunto de dados foram definidas duas classes de problemas, que variam de acordo com o custo e a capacidade, que podem ser do tipo “frouxo” (*Loose - L*) e “apertado” (*Tight - T*).

Os conjuntos de dados de teste CAB e AP para problemas de localização de concentradores têm sido referenciados em diversos trabalhos (Sasaki e Fukushima (2002), Silva (2004), Topcuoglu (2005) e Chen (2007, 2008)). A utilização destas instâncias de teste tem facilitado a comparação de diferentes abordagens propostas para o problema.

O'Kelly (1987) apresentou um modelo matemático com função objetivo quadrática para o problema de localização de  $p$ -concentradores não-capacitado (USApHMP). Sua formulação pode ser descrita da seguinte forma:

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

Sujeito a:

$$\sum_k x_{kk} = p \quad (2.2)$$

$$\sum_k x_{ik} = 1, \quad \forall i \in V \quad (2.3)$$

$$x_{kk} - x_{ik} \geq 0, \quad \forall i, k \in V, \quad (2.4)$$

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

Em que:

- $V$  é conjunto de nós da rede;
- $p$  é o número de concentradores a serem localizados;
- $d_{ij}$  é a distância entre os nós  $i$  e  $j$ ;
- $w_{ij}$  é a quantidade de fluxo que saí de  $i$  e chega em  $j$ ;
- $\lambda$ ,  $\alpha$ ,  $\delta$  são, respectivamente, os custos de coleta, transferência e distribuição;
- $x_{ik}$  é uma variável de decisão, tal que  $x_{ik} = 1$  se o nó  $i$  está alocado ao concentrador  $k$ , e  $x_{ik} = 0$ , caso contrário. Deve-se observar que se  $x_{kk} = 1$  então o nó  $k$  é um concentrador; Caso contrário,  $x_{kk} = 0$ .

Nessa formulação, a função-objetivo (2.1) estabelece o custo total a ser minimizado, que corresponde à soma dos custos de coleta, transferência e distribuição de uma rede, a restrição (2.2) fixa o número de concentradores igual a  $p$ , as restrições (2.3) garantem que cada nó de demanda será alocado a um único concentrador, as restrições (2.4) asseguram que as alocações serão

feitas apenas para nós que são concentradores, e as restrições (2.5) correspondem às condições de integralidade das variáveis de decisão.

Campbell (1994) mostrou um modelo de Programação Linear Inteira para o USApHMP. No entanto, este modelo exige um número muito grande de variáveis e restrições

Em seguida, vários outros modelos foram propostos, dentre os quais se destaca o de Ernst e Krishnamoorthy (1996) com o qual os autores relatam resultados considerados melhores que os obtidos por outros modelos. Ernst e Krishnamoorthy (1996) formularam o problema de localização de p-concentradores como:

$$f(x) = \text{Min} \sum_i \sum_k d_{ik} x_{ik} (\lambda E_i + \delta S_i) + \sum_i \sum_k \sum_l \alpha d_{kl} Y^{i_{kl}} \quad (2.6)$$

Sujeito a: (2.2)-(2.5) e

$$\sum_l Y^{i_{kl}} - \sum_l Y^{i_{lk}} = E_i x_{ik} - \sum_j w_{ij} x_{jk} \quad \forall i, k \in V \quad (2.7)$$

$$Y^{i_{kl}} \geq 0 \quad \forall i, k, l \in V \quad (2.8)$$

Em que: os símbolos  $V$ ,  $p$ ,  $d_{ij}$ ,  $w_{ij}$ ,  $\lambda$ ,  $\alpha$ ,  $\delta$  e  $x_{ik}$  têm os mesmos significados já apresentados para o modelo (2.1)-(2.5),  $Y^{i_{kl}}$  significa a quantidade de fluxo transferido entre os concentradores  $k$  e  $l$  originado a partir do nó  $i$ ,  $E_i$  é a quantidade de fluxo que chega no nó de demanda  $i$ , ou seja,  $E_i = \sum_j w_{ji}$ , e  $S_i$  é a quantidade de fluxo que deixa o nó  $i$ , ou seja,  $S_i = \sum_j w_{ij}$ .

A restrição (2.7) faz o balanceamento do nó  $i$  para o nó  $k$  em que a quantidade de fluxo que entra e o fluxo que sai são determinados pela variável de decisão  $x_{ik}$ . A partir deste modelo, Ernst e Krishnamoorthy (1996) propuseram um algoritmo *simulated annealing* para o USApHMP capaz de determinar bons

limitantes superiores para um método do tipo *branch-and-bound*. Com esse algoritmo, os autores encontraram soluções ótimas para exemplares do problema definidos em redes de até 50 nós.

Ernst e Krishnamoorthy (1998) propuseram um algoritmo do tipo *branch-and-bound* para o USApHMP. Essa proposta utiliza soluções de caminho mais curto obtidas por meio do algoritmo *Floyd–Warshall* para obtenção dos limites inferiores para o método *branch-and-bound*. Com essa estratégia os autores conseguiram obter soluções exatas para exemplares do problema definidos em redes de até 100 nós.

Kratica *et al.* (2007) apresentaram dois algoritmos genéticos com estruturas diferentes para a representação dos indivíduos. Estes algoritmos trabalham apenas com soluções factíveis, ou seja, caso uma solução infactível apareça durante a busca, o algoritmo se encarrega em viabilizá-la. Para o trabalho em questão os autores apresentaram um conceito denominado bits congelados (nome dado aos bits de um indivíduo que não se alteram devido à convergência prematura do método). Uma vez identificados esses bits o método permite uma probabilidade maior de mutação a eles, para que assim o método obtenha um aumento da diversidade do material genético. Com essa estratégia os autores obtiveram soluções ótimas para exemplares do conjunto de dados AP com até 50 nós, para o problema USApHMP.

Chen (2007) apresenta um algoritmo híbrido composto por um algoritmo *Simulated Annealing* e uma Lista Tabu. Além disso, acrescentou-se um procedimento heurístico para o cálculo de um limitante superior para o número de concentradores. Com essa idéia, a metaheurística implementada gerou soluções muito boas em termos de eficiência e qualidade de solução para o USAHLP. Chen (2007) conclui que a boa qualidade das soluções pode ser devido à inclusão dos limitantes superiores para o número de concentradores.

Chen (2008) propõe uma heurística híbrida semelhante à proposta por Chen (2007), para resolver o USApHMP. Tal método é composto por um algoritmo

*simulated annealing* e uma lista tabu. Com este trabalho o autor obteve soluções ótimas para problemas do conjunto AP com até 50 nós, e boas soluções em um tempo razoável para problemas com mais de 100 nós.

O'Kelly (1992) apresentou um modelo para o problema de localização de concentradores, no qual cada instalação possui um custo fixo de abertura de do concentrador que depende do ponto escolhido. Duas possibilidades foram consideradas: o problema de localização de concentradores capacitado, em que cada concentrador pode ter uma capacidade máxima de atendimento (alocação) e o problema não-capacitado, em que não existem limites para a capacidade de atendimento dos concentradores.

O problema não-capacitado foi formulado como:

$$f(x) = \text{Min.} \sum_i \sum_j w_{ij} \sum_k \lambda d_{ik} x_{ik} + \sum_i \sum_j w_{ji} \sum_l \delta d_{jl} x_{jl} \quad (2.9)$$

$$+ \sum_i \sum_k x_{ik} \sum_j \sum_l \alpha x_{jl} d_{kl} w_{ij} + \sum_k f_k x_{kk}$$

Sujeito às restrições: (2.3)-(2.5).

Deve-se observar que a função-objetivo (2.9) diferencia-se da função (2.1) apenas quanto ao acréscimo do somatório dos custos fixos de localização dos concentradores, aqui representados por  $f_k$ . Com isso, o número de concentradores passa a ser uma variável de decisão, ou seja, a restrição (2.2) não é mais necessária visto que a escolha da quantidade de concentradores pode influenciar no custo total da rede.

Existem vários estudos relacionados a problemas de localização de concentradores não capacitados, entre esses o de Topcuoglu *et al.* (2005) que apresentam um algoritmo genético e comparam seus resultados com um algoritmo *simulated annealing*, implementado pelos mesmos autores, e com um algoritmo genético com busca tabu (ABDINNOUR-HELM, 1998) para os

conjuntos CAB e AP. Este trabalho resultou em um método eficiente e robusto, capaz de obter soluções boas em tempos computacionais reduzidos.

Cunha e Silva (2007) propuseram um algoritmo genético híbrido agregado a um método *simulated annealing* com a finalidade de melhorar a função de adaptação de cada indivíduo. Os autores testaram o método para problemas CAB e aplicaram o método a um problema real da companhia de caminhões LTL do Brasil, em que os custos não são simétricos, ou seja, os custos de ida são diferentes dos custos de volta. O algoritmo foi aplicado a uma rede de 46 nós, em que cada nó representa uma determinada cidade de uma filial da empresa. Tal trabalho gerou resultados muito bons para a companhia de caminhões. Algumas das melhores soluções ajudaram a empresa brasileira a decidir sobre o uso de uma nova configuração de rede próxima da ótima, com finalidade de reduzir os custos.

De acordo com Alumur e Kara (2008) as melhores heurísticas até o momento para o problema de localização de concentradores não-capacitado são as propostas por Chen (2007) e Cunha e Silva (2007).

Farahani *et al.* (2013) apresentam uma revisão dos modelos e técnicas desenvolvidas recentemente para os problemas de localização de concentradores. Neste trabalho foi realizada uma classificação das variações do problema de localização de concentradores e apresentados os principais trabalhos relacionados a esses problemas. Foi identificado também que a maioria dos problemas não trata de questões de confiabilidade, no caso de ocorrência de falha nos concentradores. Entre os métodos relacionados existem diversas heurísticas como *Simulated Annealing*, Busca Tabu e Algoritmos Genéticos, e os métodos exatos relacionados utilizam *branch-and-cut*, *branch-and-price*, mas em nenhum dos casos foram apresentados métodos que utilizam a estratégia *Local branching*.

## 2.2. Problema de Localização de Concentradores Capacitado com Alocação Única

Aykin (1994) desenvolveu uma formulação permitindo conexões diretas entre clientes (*spokes*), de forma que, em alguns casos, um cliente pode enviar uma encomenda para outro sem passar por um concentrador. Em seguida, Aykin (1995a) apresenta uma formulação com um dado número de concentradores para localizar.

Um modelo quadrático para o CSAHLP pode ser descrito pela mesma função-objetivo (2.9) sujeito às restrições (2.3), (2.4) e (2.5), porém com a inclusão das restrições de capacidade, dadas pelas restrições (2.10):

$$\sum_i E_i x_{ik} \leq Q_k x_{kk} \quad \forall k \in V \quad (2.10)$$

Em que  $Q_k$  é a capacidade do concentrador  $k$ . Deve-se observar que as restrições (2.10) limitam a alocação dos nós de demanda à capacidade máxima de cada concentrador do problema.

Ernst e Krishnamoorthy (1999), através de algumas alterações no modelo proposto pelos mesmos autores para o USApHMP, apresentaram uma linearização do modelo matemático com menos variáveis e restrições para o problema capacitado:

$$f(x) = \text{Min} \sum_i \sum_k d_{ik} x_{ik} (\lambda E_i + \delta S_i) + \sum_i \sum_k \sum_l \alpha d_{kl} Y_{kl}^i + \sum_k f_k x_{kk} \quad (2.11)$$

Sujeito a: (2.3), (2.4), (2.5), (2.7), (2.8) e (2.10).

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 obtiveram soluções ótimas para

problemas do conjunto 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 apertado (*Tight - T*).

### **2.3. Problema de Localização de Concentradores Capacitado com Alocação Múltipla**

O problema de localização de concentradores com alocação múltipla pode ser considerado um dos mais importantes, devido à sua aproximação com a realidade. Nem sempre a alocação única torna uma rede mais barata. Quando se permite que um cliente seja alocado a mais que um terminal, a solução deste novo problema pode trazer ganhos significativos para a rede final.

A Figura 2.1 apresenta dois exemplos de rede do *tipo hub-and-spoke*. Na primeira figura cada cliente (pontos) só pode ser alocado a um único concentrador (quadrados), enquanto que na segunda, um cliente pode ser alocado a mais que um concentrador. Este exemplo foi apresentado por Ernst e Krishnamoorthy (1998) e, neste caso,  $n = 25$  e  $p$  foi restrito a 4 concentradores. Neste exemplo, o custo ótimo para o PLC com alocação simples é 139197,17. No entanto, quando se muda o problema, permitindo a alocação múltipla (ou seja, um cliente ser alocado a mais que um concentrador) o custo ótimo passa a ser 135638,58, um ganho de 3%. Em situações reais, por exemplo, o gerente de transporte de uma empresa poderia optar pelo problema com alocação múltipla.

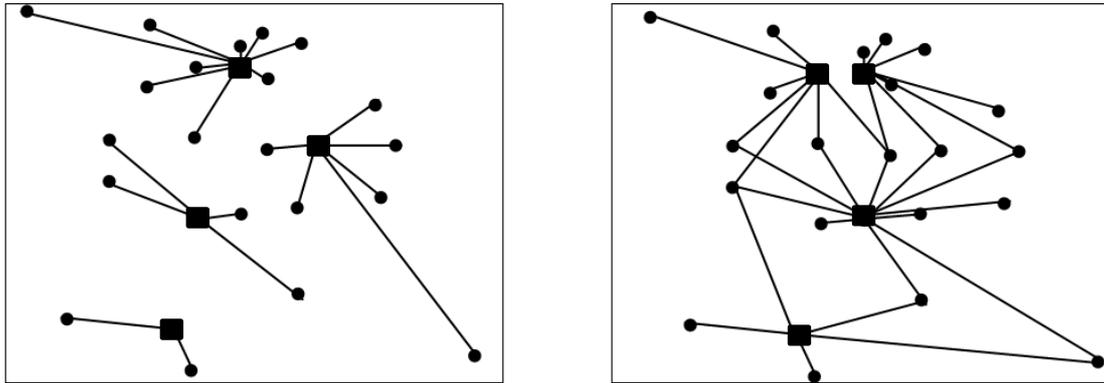


Figura 2.1– Exemplo de rede com alocação única e múltipla  
 Fonte: Adaptada de Ernst e Krishnamoorthy (1998)

Uma variação do modelo quadrático proposto por O’Kelly (1987) para o CMAHLP (*Capacitated Multiple Allocation Hub Location Problem*) pode ser apresentado pelo modelo matemático formado pela função objetivo (2.9), sujeito às restrições (2.4), (2.5) e (2.10).

Campbel (1994) propôs o seguinte modelo matemático para o CMAHLP com alocação múltipla:

$$f(x) = \text{Min} \sum_i \sum_j \sum_k \sum_m w_{ij} C_{ijkm} x_{ijkm} + \sum_k f_k H_k \quad (2.12)$$

Sujeito a:

$$\sum_k \sum_m x_{ijkm} = 1, \quad \forall i, j \in V, \quad (2.13)$$

$$\sum_m x_{ijkm} \leq H_k, \quad \forall i, j, k \in V, \quad (2.14)$$

$$\sum_k x_{ijkm} \leq H_m, \quad \forall i, j, m \in V, \quad (2.15)$$

$$\sum_i \sum_j w_{ij} \sum_m x_{ijkm} \leq Q_k H_k, \quad \forall k \in V, \quad (2.16)$$

$$H_k \in \{0,1\} \quad \forall k \in V. \quad (2.17)$$

$$0 \leq x_{ijkm} \leq 1 \quad \forall i, j, k, m \in V. \quad (2.18)$$

Neste modelo os dados do problema, que também correspondem aos valores apresentados nos modelos anteriores, são:

- $V$  é conjunto de nós da rede;
- $w_{ij}$  é a quantidade de fluxo transferido entre os nós  $i$  e  $j$ ;
- $\lambda$ ,  $\alpha$ ,  $\delta$  são, respectivamente, os custos de coleta, transferência e distribuição;
- $f_k$  corresponde ao custo fixo para a abertura de um concentrador no ponto  $k$ ;
- $Q_k$  corresponde à capacidade máxima do concentrador  $k$ ;
- $d_{ij}$  corresponde à distância entre um nó  $i$  e um nó  $j$  qualquer.

As variáveis de decisão do problema são:

- $x_{ijkm}$  é a fração do fluxo que é transferida via concentradores  $k$  e  $m$ , entre um cliente  $i$  e um cliente  $j$ ;
- $H_k$  é uma variável de decisão que define se o ponto  $k$  é um concentrador ( $H_k = 1$ ) ou não ( $H_k = 0$ );

O parametro  $C_{ijkm}$  é o custo por unidade de fluxo de  $i$  para  $j$  via concentradores  $k$  e  $m$ . ( $C_{ijkm} = \lambda d_{ik} + \alpha d_{km} + \delta d_{mj}$ ).

A função objetivo (2.12) do PLC capacitado consiste em minimizar o custo total da rede incluindo o custo fixo para abertura de um concentrador. As restrições (2.13) garantem que não vai haver perda, ou seja, a soma das frações de uma encomenda que sai de  $i$  com destino em  $j$ , passando por qualquer rota de

concentradores  $(k, m)$  deve ser igual a 1 (100%). As restrições (2.14) e (2.15) determinam que as transferências somente ocorram via concentradores. As restrições (2.16) definem a capacidade de cada concentrador. As restrições (2.17) definem as restrições de integralidade das variáveis  $H_k$  e as restrições (2.18) definem os limites da fração do fluxo da variável  $x_{ijkm}$ .

Para o desenvolvimento deste trabalho utilizou-se o modelo matemático CMAHLP-F proposto por Ebery *et al.* (2000) que corresponde ao modelo (2.19-2.24) a seguir. Esse modelo foi escolhido por utilizar menos variáveis e por mostrar capacidade de trabalhar com valores de  $n$ , número de pontos da rede, maiores que 40 sem exceder a capacidade da memória.

$$f(x) = \text{Min} \sum_i \sum_k \sum_l Y_{kl}^i (\lambda d_{ik} + \alpha d_{kl}) + \sum_i \sum_k \sum_l \delta d_{ij} x_{ij}^i + \sum_k f_k H_k \quad (2.19)$$

Sujeito à:

$$\sum_l x_{ij}^i = w_{ij}, \quad \forall i, j \in V, \quad (2.20)$$

$$\sum_i \sum_l Y_{kl}^i \leq Q_k H_k, \quad \forall k \in V, \quad (2.21)$$

$$\sum_i \sum_k Y_{kl}^i \leq \sum_i \sum_j w_{ij} H_l, \quad \forall l \in V, \quad (2.22)$$

$$\sum_k Y_{kl}^i = \sum_j x_{ij}^i, \quad \forall i, l \in V, \quad (2.23)$$

$$Y_{ij}^i, x_{kl}^i \geq 0, H_k \in \{0,1\} \quad \forall i, j, k, l \in V, \quad (2.24)$$

Neste modelo as variáveis possuem as seguintes definições:

- $Y_{kl}^i$  corresponde ao fluxo do nó  $i$  que passa via concentradores  $k$  e  $l$ ;
- $x_{ij}^i$  determina a quantidade de fluxo com origem em  $i$  e destino em  $j$  que passa pelo concentrador  $l$ ;

- $H_k$  é uma variável de decisão que define se o ponto  $k$  é um concentrador ( $H_k = 1$ ) ou não ( $H_k = 0$ );
- E os outros dados do problema possuem o mesmo significado do modelo matemático (2.12-2.18).

Para o modelo matemático (2.19-2.24) a função objetivo (2.19) determina que o custo de localização e alocação dos concentradores deve ser o menor possível, considerando os custos de transporte e os custos fixos para localização de concentradores. As restrições (2.20) definem que a soma das parcelas dos fluxos com origem em  $i$  e destino em  $j$  que passam pelos concentradores  $l$  deve ser igual ao fluxo total que sai de  $i$  e chega em  $j$  ( $w_{ij}$ ). As restrições (2.21) garantem que a soma dos fluxos que chegam ao concentrador  $k$  não ultrapassa a capacidade máxima deste concentrador. Para as restrições (2.22) observa-se que a soma total dos fluxos que chegam em  $l$  deve ser menor ou igual à quantidade total de fluxo que está associado ao concentrador  $H_l$  escolhido. As restrições (2.23) asseguram que o fluxo total que chega ao concentrador  $l$  deve ser igual à soma de todos os fluxos associados ao concentrador  $l$ . E as restrições (2.24) definem os limites das variáveis de decisão.

Conforme apresentado existem diversas variações do problema de localização de concentradores, e para cada um destes existem variados modelos matemáticos e métodos de solução. No entanto, em nenhum dos trabalhos relatados na literatura estuda-se a técnica *Local Branching* como estratégia de solução.

### 3 HEURÍSTICAS BASEADAS EM MODELO

As heurísticas baseadas em modelos (também conhecidas como *matheuristics*) têm recebido atenção recente devido ao avanço dos *solvers* de problemas de Programação Matemática. Nesta seção serão apresentados quatro tipos de *matheuristics*: o método *Relax-and-Fix* (R&F), o método *Fix-and-Optimize* (F&O), o método *Local Branching* (LB) e o método LB associado ao modelo VNS (*Variable Neighborhood Search*).

A ideia básica dos métodos R&F e F&O é resolver, de forma iterativa, uma série de subproblemas que são obtidos da formulação do problema original. A cada iteração, muitas das variáveis binárias são fixadas a valores previamente estabelecidos, reduzindo o número de variáveis binárias a serem otimizadas no subproblema correspondente àquela iteração. Os subproblemas resultantes são então resolvidos pelo *solver* até a otimalidade. Como o número de variáveis binárias do subproblema é muito menor do que no problema original, o tempo de solução para um subproblema é pequeno. Isso fornece uma nova solução temporária para as variáveis binárias do subproblema atual. Algumas delas são fixadas na próxima iteração, quando um subconjunto diferente de variáveis binárias é otimizado. Na heurística R&F as variáveis binárias são divididas em 3 grupos para cada subproblema: (a) as variáveis que são fixadas; (b) as variáveis que são otimizadas; e (c) as variáveis para as quais as restrições de integralidade são relaxadas. A heurística F&O opera somente com os dois primeiros grupos de variáveis binárias.

A ideia básica do método LB é explorar a estrutura de modelos de Programação Inteira em que um conjunto de variáveis binárias particiona o problema em níveis, de modo que a fixação do valor das variáveis de um nível produz um subproblema mais fácil de ser resolvido. O procedimento tem o espírito das metaheurísticas de busca local, mas as vizinhanças são obtidas por meio da inserção de cortes (denominados restrições LB ou restrições de ramificação local) na formulação de Programação Matemática do problema

original. Portanto, o método LB tem o espírito do método *branch-and-cut*, em que a inserção de restrições no modelo matemático que representa o problema restringe vizinhanças a serem exploradas por meio de um *solver*.

O método LB associado ao modelo VNS consiste de uma estratégia de busca em vizinhança por meio da inserção de restrições LB no modelo matemático do problema. Com isso, a restrição LB se encarrega de estabelecer a vizinhança máxima a ser explorada e um *solver* é utilizado para a exploração dessas vizinhanças.

Esses métodos baseados em modelos de Programação Matemática têm sido utilizados para resolver diversas classes de problemas complexos de otimização, como: projeto de redes de telecomunicação (Fischetti *et al.*, 2004), problema de sequenciamento de guindastes de terminais marítimos (Legato e Trunfio, 2014), problemas de sequenciamento de projetos (Escudero e Salmeron, 2005), problemas de dimensionamento de lotes (Sahling *et al.*, 2009; Lang e Shen, 2011; Moraes e Santos, 2012), problemas integrados de dimensionamento e sequenciamento de lotes (Araújo *et al.*, 2007; Ferreira *et al.*, 2010; Kawamura e Ronconi, 2010). Essas técnicas têm sido usadas também em algoritmos híbridos, em conjunto com metaheurísticas como, por exemplo, busca tabu (Pedroso e Kubo, 2005) e algoritmos genéticos (Toledo *et al.*, 2011).

### **3.1. O Método *Relax-and-Fix***

A técnica R&F é um método iterativo que decompõe um problema de programação inteira mista de difícil solução em subproblemas menores, que podem ser resolvidos rapidamente. A Figura 3.1 mostra o fluxo de controle desta técnica.

Neste algoritmo,  $X$  é o conjunto das variáveis binárias do problema,  $S$  é o conjunto de subproblemas referentes a uma decomposição,  $X_s^o$  é o conjunto das variáveis binárias a serem otimizadas em um dado subproblema  $s \in S$ . O algoritmo começa transformando o problema original de forma que todas as variáveis binárias sejam relaxadas continuamente. Em seguida, os subproblemas referentes a uma determinada decomposição são ordenados de acordo com um critério conveniente. A cada iteração, o algoritmo, em primeiro lugar, chama a rotina `LimparAlteracoes()` para remover as relaxações e fixações das variáveis binárias a serem otimizadas, ou seja, incluindo-as em  $X_s^o$ . Em seguida, chama `ResolverModelo()` para executar um *solver* ao problema com as atuais fixações e relaxações de variáveis binárias. Se nenhuma solução viável puder ser encontrada para o subproblema, o algoritmo termina e retorna uma solução vazia (`status = 0`). Caso contrário, chama `FixarValores()` para fixar todas as variáveis binárias otimizadas no subproblema a seus valores atuais. O algoritmo termina normalmente (`status = 1`) quando todos os subproblemas tiverem sido resolvidos.

<ol style="list-style-type: none"> <li>1. <b>Algoritmo Relax&amp;Fix()</b></li> <li>2.     <code>RelaxarContinuamente(X);</code></li> <li>3.     <code>LS ← OrdenarSubproblemas(S);</code></li> <li>4.     <b>para cada</b> <math>s \in LS</math> <b>fazer</b></li> <li>5.         <code>LimparAlteracoes(<math>X_s^o</math>);</code></li> <li>6.         <code>(status, solução) ← ResolverModelo();</code></li> <li>7.         <b>se</b> (<code>status = 0</code>) <b>então</b></li> <li>8.             <b>retornar</b> <math>\phi</math>;</li> <li>9.         <b>fim-se</b></li> <li>10.        <code>FixarValores(<math>X_s^o</math>);</code></li> <li>11.     <b>fim-para</b></li> <li>12.     <b>retornar</b> solução;</li> <li>13. <b>fim-algoritmo</b></li> </ol>
---

Figura 3.1 – Algoritmo R&F

### 3.2. O Método *Fix-and-Optimize*

A técnica F&O também decompõe um problema de programação inteira mista em subproblemas menores.

A Figura 3.2 mostra o fluxo de controle desta técnica. Este algoritmo começa gerando uma solução inicial viável. Caso não seja possível encontrar uma solução viável inicial, o algoritmo termina com status = 0. Do contrário, chama `FixarValores()` para fixar as variáveis binárias aos seus valores nesta solução inicial. Em seguida, esses valores são salvos pela rotina `SalvarValores()`, para que possam ser restaurados mais tarde. A cada iteração, o algoritmo remove as relaxações e fixações das variáveis binárias a serem otimizadas e resolve o subproblema fixando todas as variáveis binárias aos valores obtidos no subproblema resolvido anteriormente. Com esses valores especificados, uma solução viável pode ser encontrada resolvendo o problema de Programação Linear restante. Se uma solução viável for encontrada, o algoritmo verifica se ela deve ser aceita. Caso seja aceita, as variáveis otimizadas são fixadas aos seus valores na nova solução. Caso contrário, a solução é descartada e as fixações de variáveis que foram removidas para resolver o subproblema são restauradas. No caso (improvável) de uma solução viável não ser encontrada, a última solução viável encontrada é restaurada, fixando-se as variáveis aos seus valores anteriores.

Deve-se observar que o algoritmo R&F pode ser usado para gerar uma solução inicial para o algoritmo F&O. Neste caso, tem-se um algoritmo em dois estágios denominado *Relax-and-Fix-and-Optimize* (R&F&O).

```

1. Algoritmo Fix&Optimize
2.   status ← GerarSolucaoInicial();
3.   se (status = 0) então
4.     retornar;
5.   fim-se
6.   FixarValores(X);
7.   SalvarValores(X);
8.   LS ← OrdenarSubproblemas(S);
9.   para cada s ∈ LS fazer
10.    LimparAlteracoes(Xso);
11.    (status, solução) ← ResolverModelo();
12.    se (status = 1) então
13.      se Aceitar(solução) então
14.        SalvarValores(X);
15.      senão
16.        RestaurarValores(X);
17.      fim-se
18.      FixarValores(Xso);
19.    senão // status = 0
20.      RestaurarValores(X);
21.      FixarValores(X);
22.    fim-se
23.  fim-para
24.  retornar solução;
25. fim-algoritmo

```

Figura 3.2 – Algoritmo F&O

### 3.3. O Método LB Associado ao Modelo VNS

A método LB associado ao modelo VNS (*VNS Matheuristic* - VNSM) utiliza a ideia da restrição *local branching* como maneira de reduzir uma vizinhança a ser explorada por meio de um *solver* em cada iteração de uma heurística VNS (DELLA CROCE e SALASSA, 2012).

Della Croce e Salassa (2012) utilizaram o método VNSM para a solução de um problema de escalas de enfermeiras. A Figura 3.5 ilustra o funcionamento do método VNSM. Inicialmente, o método gera uma solução inicial viável para o problema. Esta solução pode ser gerada aleatoriamente ou por outro método

trabalhando de forma híbrida. Em seguida uma restrição *local branching* é inserida no modelo matemático do problema. Esta restrição limita uma vizinhança a ser explorada pelo *solver* a uma distância máxima  $k$  em relação à solução corrente. Após a execução do *solver*, se uma nova solução melhor que a solução corrente for encontrada, a última restrição *local branching* é removida, e uma nova restrição *local branching* é inserida. O processo continua e se repete, até que o *solver* não seja capaz de encontrar uma nova solução melhor que a solução corrente. Por fim, o método VNSM retorna a melhor solução encontrada do processo de busca. A estratégia apresentada é bem parecida com o método *Local Branching*. Sua principal diferença está na eliminação das restrições inseridas.

<ol style="list-style-type: none"> <li>1. <b>Algoritmo VNSM()</b></li> <li>2.     gere uma solução inicial viável <math>x'</math> para o problema;</li> <li>3.     <math>x^* \leftarrow \infty</math>;</li> <li>4.     <b>enquanto</b> Critério de Parada Não Satisfeito <b>faça</b></li> <li>5.         adicione a restrição <math>\Delta(x', x^*) \leq k</math> ao modelo MIP;</li> <li>6.         resolva o subproblema resultante;</li> <li>7.         <b>se</b> uma solução viável <math>x'</math> melhor que <math>x^*</math> for encontrada <b>então</b></li> <li>8.             remova a restrição <math>\Delta(x', x^*) \leq k</math> anterior do modelo MIP;</li> <li>9.             <math>x^* \leftarrow x'</math>;</li> <li>10.         <b>fim-enquanto</b></li> <li>11.         <b>retorne</b> <math>x^*</math>;</li> <li>12. <b>fim-algoritmo</b></li> </ol>
--

Figura 3.5– Algoritmo VNSM

### 3.4. O Método *Local Branching*

A estratégia LB, embora criada como uma abordagem de alta generalidade, pode ser usada para explorar a estrutura específica de alguns modelos de Programação Inteira em que um conjunto de variáveis binárias particiona naturalmente o problema em dois níveis, com a propriedade de que a fixação do valor das variáveis do primeiro nível produz subproblemas menores para serem resolvidos.

Esta ideia tem sido usada, com sucesso, para a solução de problemas de localização de facilidades em projetos de redes de telecomunicação (FISCHETTI *et al.*, 2004). Os problemas de localização de concentradores abordados neste trabalho, no entanto, diferem dos problemas de localização de facilidades abordados, pois além da minimização dos custos de localização (abertura) dos concentradores, incluem também a minimização do custo total da rede, que representam os custos variáveis de transporte, calculados por meio dos custos de coleta, transferência e distribuição.

Para a aplicação da estratégia LB deve-se dispor de uma solução inicial  $\bar{x}$  binária, denominada solução de referência (Fischetti e Lodi, 2003). Seja  $S = \{j \in B \mid \bar{x}_j = 1\}$ , para um dado parâmetro inteiro  $k$ , pode-se definir a vizinhança  $N(\bar{x}, k)$  da solução de referência  $\bar{x}$  como o conjunto de soluções viáveis  $x$  do problema que satisfazem à seguinte restrição adicional, denominada restrição de ramificação local:

$$\sum_{j \in S} (1 - x_j) + \sum_{j \notin S} x_j \leq k \quad (3.1)$$

Nesta restrição, os dois termos do lado esquerdo contam o número de variáveis binárias de uma solução  $x$  que mudaram de valor (de 1 para 0 e de 0 para 1, respectivamente), em relação à solução de referência  $\bar{x}$ . Esta restrição, portanto, impõe que  $k$  é a maior distância entre vizinhos viáveis de  $\bar{x}$ . Tal restrição de ramificação local pode ser usada em um método enumerativo como um critério de ramificação, considerando  $\Delta(x, \bar{x}) = \sum_{j \in S} (1 - x_j) + \sum_{j \notin S} x_j$ , o método utiliza:

- $\Delta(x, \bar{x}) \leq k$ , para o ramo esquerdo;
- $\Delta(x, \bar{x}) \geq k+1$ , para o ramo direito.

Os vizinhos definidos pelas restrições de ramificação local podem ser explorados (com um *solver*) usando-se, por exemplo, o critério de ramificação nas variáveis fracionárias. A Figura 3.3 apresenta a ideia básica da técnica LB.

O método começa a busca a partir de uma solução de referência inicial. A cada nível da árvore é adicionado uma nova restrição *local branching* para que um *solver* possa explorar vizinhanças da solução apresentadas em cada ramificação. A árvore adiciona as restrições locais enquanto ocorrer melhora nas soluções correntes. Nesse caso o método é considerado exato, no entanto, quando se considera restrições de tempo ou nem todas as ramificações da árvore de enumeração LB são exploradas, a estratégia passa a se comportar como uma heurística.

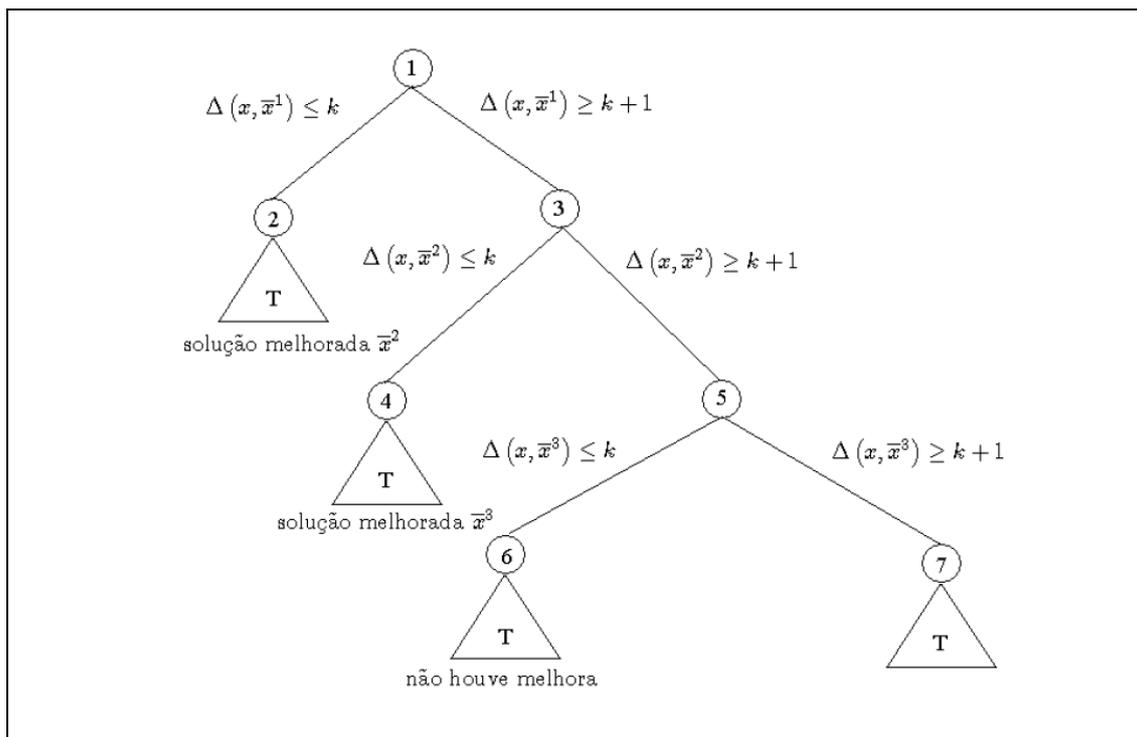


Figura 3.3– Árvore de enumeração LB

Como neste trabalho pretende-se utilizar a estratégia LB como heurística, o método desenvolvido não explora todas as possíveis ramificações da árvore de enumeração LB. Uma das principais justificativas é que o método LB antecipa melhorias das soluções submetidas a ele, e normalmente como o método já

inicia com uma boa solução, ou seja, com um limitante superior próximo do ótimo, a exploração de apenas algumas ramificações é, normalmente, suficiente em termos de desempenho e qualidade.

Na Figura 3.4 é apresentado o pseudocódigo do método *Local Branching*. Inicialmente o método gera uma solução inicial para servir como solução de referência, em seguida, uma restrição *local branching* da forma  $\Delta(x, \bar{x}) \leq k$  é adicionada ao modelo de programação inteira mista (MIP, *Mixed Integer Programming*) do PLC. Em seguida, um *solver* entra em funcionamento e realiza a busca por uma nova solução. Caso uma solução de menor custo seja encontrada, a última restrição LB da forma  $\Delta(x, \bar{x}) \leq k$  é excluída e uma restrição LB da forma  $\Delta(x, \bar{x}) \geq k+1$  é incluída no modelo, o que corresponde à explorar uma ramificação do lado direito da árvore de enumeração LB. Em seguida, uma nova restrição LB  $\Delta(x, \bar{x}) \leq k$  baseada na nova solução encontrada é inserida e esse processo se repete enquanto uma nova solução de menor custo seja encontrada.

Neste método, o resultado final da estratégia LB depende fortemente da solução de referência inicial e do valor de  $k$ . Se o valor de  $k$  for muito pequeno o método pode não ser capaz de melhorar a solução inicial. Ao contrário, se o valor de  $k$  é muito grande a exploração de uma ramificação da árvore pode ser muito demorada. Assim, é importante que o valor de  $k$  e uma boa solução inicial sejam bem estabelecidos para que o método seja capaz de alcançar um bom desempenho e uma boa solução.

O desempenho do método pode ser melhorado incorporando-se mecanismos de diversificação, como os que ocorrem em metaheurísticas. Um ramo esquerdo da árvore de enumeração que não leva a soluções de referência melhores pode ser reavaliado, por exemplo, explorando-se uma vizinhança maior ou aplicando-se uma busca como na metaheurística VNS (*Variable Neighborhood Search*) (HANSEN *et al.*, 2006). Outras possibilidades de melhorias são o tratamento de soluções tabus e a aplicação das restrições

locais como cortes que levem em conta a estrutura do problema (HAMACHER *et al.*, 2004).

```
1. Algoritmo Local Branching()
2.   gere uma solução inicial viável  $x'$  para o problema;
3.    $x^* \leftarrow x'$ ;
4.   loop
5.     adicione a restrição  $\Delta(x', x^*) \leq k$  ao modelo MIP;
6.     resolva o subproblema resultante;
7.     se uma solução viável  $x'$  melhor que  $x^*$  for encontrada então
8.       remova a restrição  $\Delta(x', x^*) \leq k$  do modelo MIP;
9.       adicione a restrição  $\Delta(x', x^*) \geq k+1$  ao modelo MIP;
10.       $x^* \leftarrow x'$ ;
11.     senão
12.       remova a restrição  $\Delta(x', x^*) \leq k$  do modelo MIP;
13.       adicione a restrição  $\Delta(x', x^*) \geq k+1$  ao modelo MIP;
14.     saia do loop;
15.   fim se
16. fim loop
17. resolva o subproblema resultante;
18. se uma solução viável  $x'$  melhor que  $x^*$  for encontrada então
19.    $x^* \leftarrow x'$ ;
20. fim se
21. retorne  $x^*$ ;
22. fim-algoritmo
```

Figura 3.4– Algoritmo *Local Branching*

Neste trabalho pretende-se utilizar o método *Local Branching* para a solução de problemas de localização de concentradores. A estratégia LB foi escolhida por ser um método capaz de encontrar soluções de boa qualidade nos estágios iniciais da árvore de enumeração e de trabalhar com buscas em vizinhanças limitadas, mas sem perder informações do espaço de soluções já explorado, o que não ocorre com o método VNSM, que perde a cada iteração informações sobre o que já foi explorado. Além disso, a estratégia LB ainda não foi aplicada a problemas de localização de concentradores. No entanto, a aplicação desta estratégia a problemas de redes de telecomunicação, que se assemelham aos problemas abordados neste trabalho, tem gerado boas soluções.

## **4 HEURÍSTICAS BASEADAS EM MODELO PARA O PLCC**

Este capítulo apresenta os métodos propostos para os problemas de localização de concentradores capacitados (PLCC) apresentados nas seções 2.2 e 2.3. Como comentado anteriormente, foram desenvolvidos métodos heurísticos baseados na técnica denominada *local branching* (LB).

Na técnica LB emprega-se um *solver* comercial para explorar (em um nível tático) de forma efetiva, subespaços de solução convenientes, definidos e controlados (em um nível estratégico) por uma estrutura de ramificação local. O procedimento tem o espírito das metaheurísticas de busca local, mas as vizinhanças são obtidas por meio da inserção de cortes de ramificação local no modelo de Programação Inteira que descreve o problema. Esta estratégia de solução se alterna entre ramificações estratégicas para definir vizinhanças de solução e ramificações táticas para explorar estas vizinhanças. O resultado é um esquema completamente geral que antecipa melhorias em soluções incumbentes e, portanto, produz soluções de alta qualidade nos estágios iniciais da árvore de enumeração, visando reduzir o tempo computacional. Este método depende de uma boa solução inicial de referência. Assim, são apresentadas a seguir algumas heurísticas candidatas a trabalhar de forma híbrida com o método proposto para o PLCC.

### **4.1. Heurísticas Candidatas para a Geração de Soluções de Referência**

Nesta seção serão apresentados alguns métodos candidatos a trabalhar como geradoras de solução inicial para a técnica LB.

O método Algoritmo Genético (AG) é uma metaheurística de busca inspirada na teoria da evolução, capaz de encontrar boas soluções para um problema. Tal método, introduzido por Holland (1975), baseia-se em determinar em uma população de indivíduos (possíveis soluções para o problema), aqueles que,

por serem mais adaptados, irão se reproduzir e gerar descendentes para novas gerações. A Figura 4.1 apresenta o algoritmo do método AG.

1. **procedimento** Algoritmo Genético
2. Crie uma população inicial aleatória de indivíduos  $Np$ .
3. Aplique a função de avaliação (*fitness*) a cada indivíduo.
4. **enquanto** (critério de parada não foi satisfeito) **faça**
5. Aplique os operadores evolutivos como: *crossover*, mutação e elitismo.
6. Aplique a função de avaliação aos novos indivíduos.
7. Selecione as soluções mais adaptadas, ou seja, as soluções com os melhores valores da função-objetivo.
8. **fim-enquanto**
9. **retorne** a melhor solução encontrada
10. **fim-procedimento**

Figura 4.1– Algoritmo do método AG

O método *Simulated Annealing* (SA), introduzido por Kirkpatrick *et al.* (1983), foi baseado no trabalho de Metropolis *et al.* (1953). Nesta técnica a “temperatura” não é constante como no trabalho de Metropolis *et al.* (1953). 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. O algoritmo do método SA é apresentado na Figura 4.2.

O método de busca em vizinhança variável (VNS, do inglês, *Variable Neighborhood Search*) proposto por Mladenovic e Hansen (1997) é uma técnica de busca local que explora o espaço de soluções através de trocas sistemáticas de estruturas de vizinhança. Sua metodologia consiste em explorar vizinhanças gradativamente mais “distantes” da solução atual. A exploração de uma nova região somente acontece se um movimento de melhora é realizado. O algoritmo do método VNS é apresentado pela Figura 4.3.

```

1. Procedimento Simulated Annealing
2.    $T \leftarrow T^0$ ;
3.   gera solução inicial  $S^0$ ;
4.    $S \leftarrow S^0$ ;
5.    $S^* \leftarrow S^0$ ;
6.   enquanto  $T > T_f$  faça (temperatura alta)
7.     para  $cont \leftarrow 1$  até  $L(T)$  faça (iterações para equilíbrio)
8.        $S' \leftarrow$  seleciona uma solução vizinha de  $S$ 
9.        $D_{custo} \leftarrow custo(S') - custo(S)$ 
10.      se  $D_{custo} < 0$  ou  $U[0,1] < \exp(-D_{custo}/T)$ 
11.        então  $S \leftarrow S'$ 
12.      se  $(S < S^*)$  então  $S^* \leftarrow S$ 
13.    fim do para
14.     $T \leftarrow \alpha T$ 
15.  fim-enquanto
16.  retorne  $S^*$ 
17. fim-procedimento

```

Figura 4.2– Algoritmo do método SA

```

1. procedimento VNS
2.   Seja  $S^0$  uma solução inicial e  $r$  o número de estruturas de vizinhança
3.    $S \leftarrow S^0$  {Solução corrente}
4.   enquanto (Critério de parada não satisfeito) faça
5.      $k \leftarrow 1$ ; {Tipo de estrutura de vizinhança}
6.     enquanto  $(k \leq r)$  faça
7.       Gere um vizinho qualquer  $s' \in N^{(k)}(s)$ 
8.        $s'' \leftarrow$  BuscaLocal( $s'$ )
9.       se  $(f(s'') < f(s))$  então
10.         $s \leftarrow s''$ ;  $k \leftarrow 1$ 
11.      senão  $k \leftarrow k + 1$ 
12.    fim-se
13.  fim-enquanto
14.  fim-enquanto
15.  retorne  $s$ 
16. fim-procedimento

```

Figura 4.3– Algoritmo do método VNS

O método de busca evolutiva por agrupamentos (ECS, do inglês, *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, do inglês, *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 CS consiste de 4 componentes conceitualmente independentes: um gerador de soluções factíveis, um processo de agrupamento, o módulo analisador e um método de busca local. A Figura 4.4 mostra, em resumo, o funcionamento do método CS.

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.

O processo de agrupamento do CS 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. Pode-se definir um limitante superior para o número de *clusters* a serem criados. 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 PLCC 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 gerador de soluções e o centro de um *cluster*.

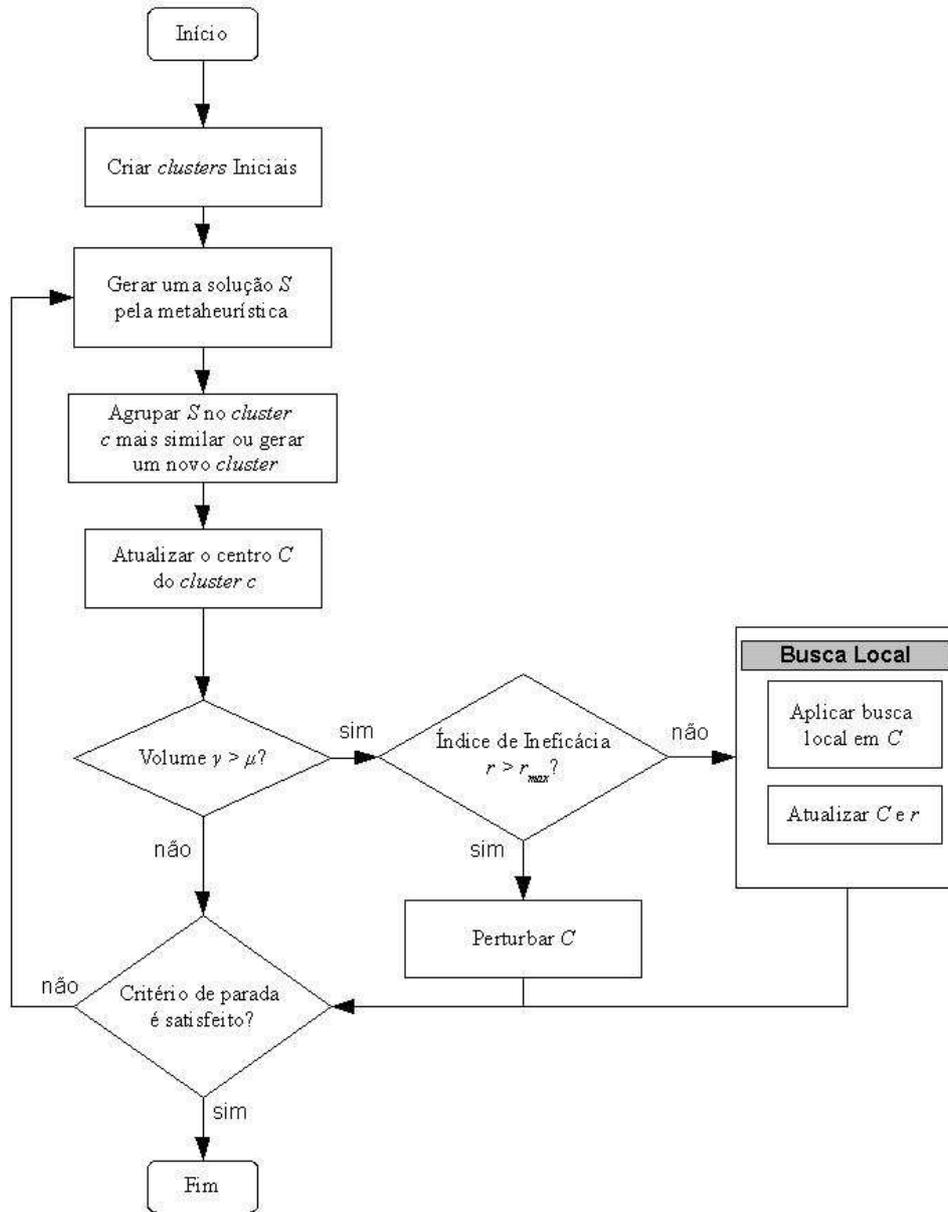


Figura 4.4 – Diagrama conceitual do CS

Fonte: Chaves (2009)

O módulo analisador examina cada *cluster*, em intervalos regulares, com o propósito de identificar um provável *cluster* promissor. O volume de um *cluster*

$(\gamma)$  é uma medida que indica o nível de atividade dentro do *cluster*. Para simplificar,  $\gamma$  pode contar o número de soluções geradas pelo gerador de soluções e agrupadas neste *cluster*. Sempre que  $\gamma$  atinge um certo limite ( $\mu$ ), o que significa que algum padrão de informação torna-se predominantemente gerado pelo gerador de soluções, este *cluster* é considerado promissor e deve ser melhor investigado para acelerar o processo de convergência.

Por fim, o método de busca local do CS é um método de busca interno que realiza a exploração de uma suposta área de busca promissora, estabelecida por um *cluster*.

Na Figura 4.5, 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 a 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 4.5. O método escolhe então uma dessas novas soluções. Supondo que após a avaliação de cada uma dessas soluções geradas, a de menor custo encontrada corresponda à solução  $(1, 1, 0, 1)$ , então, a partir desta solução, 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 4.5. O processo se repete, e este procedimento continua até que a solução guia seja encontrada.

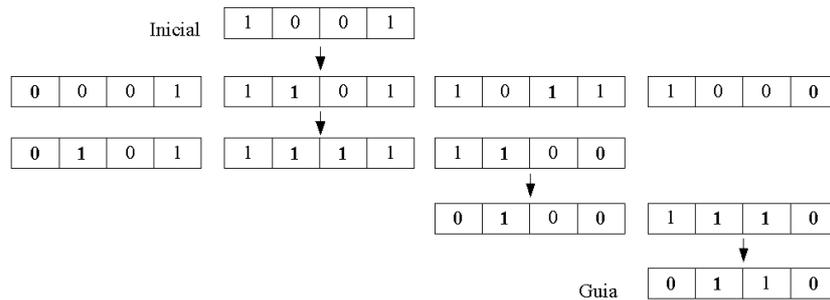


Figura 4.5 – Exemplo de *path-relinking* aplicado ao CSAHLP

Com este método, define-se um “caminho” entre uma solução inicial e uma solução guia. Neste caso, a solução inicial é a nova solução e a guia corresponde ao centro do *cluster*. A melhor solução encontrada em qualquer nível deste caminho é utilizada pelo processo de agrupamento para atualizar o centro do *cluster*.

Heurísticas de busca local utilizada no método CS são métodos que partem de uma solução inicial viável e tentam 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 um ótimo local. Um algoritmo de busca local básico, denominado de método de descida (MD) é apresentado na Figura 4.6.

1. **procedimento** Método de Descida
2.     Selecione a solução inicial  $s$
3.     Escolha o melhor vizinho  $s' \in N(s)$
4.     **enquanto**  $s'$  melhor que  $s$  **faça**
5.          $s \leftarrow s'$
6.     Escolha o melhor vizinho  $s' \in N(s)$
7.     **fim-enquanto**
8.     **retorne**  $s$
9. **fim-procedimento**

Figura 4.6 – Algoritmo do método MD

## 4.2. Método de Solução para o Problema de Localização de Concentradores Capacitado com Alocação Única

Para o PLC com alocação única considerado neste trabalho, a representação das soluções baseia-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*. Pela Figura 4.7 é possível observar que 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. Foram acrescentadas mais duas representações,  $P$  que equivale ao número de concentradores da solução corrente e o vetor *Ind\_Hub* que define os índices dos clientes que foram fixados como concentradores.

Em um PLC com alocação única pretende-se identificar uma rede para um conjunto de clientes/pontos que podem ser ilustrados pelo lado esquerdo da Figura 4.8. Após a geração de uma possível solução, como a apresentada pela Figura 4.7, esta solução pode ser representada pelo lado direito da Figura 4.8. Observe que os concentradores são representados por quadrados, cada cliente está associado a um único concentrador, e todos os concentradores se interconectam entre si formando uma rede do tipo *hub-and-spoke*.

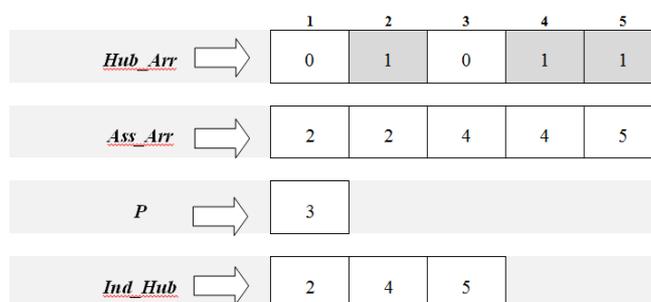


Figura 4.7 – Representação de uma solução CSAHLP

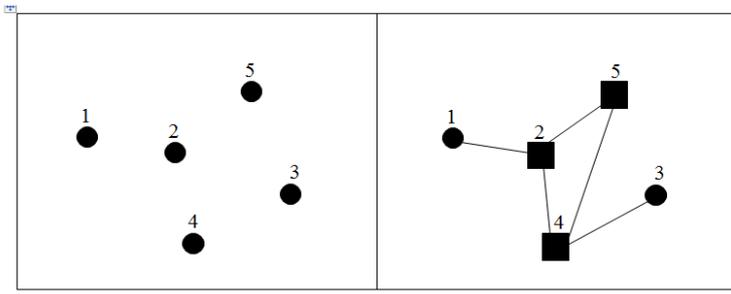


Figura 4.8 – Solução visual para o CSAHLP

As vizinhanças definidas para este trabalho correspondem aos seguintes movimentos:

- *Swap Nodes*: responsável 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*: escolhe 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 passam 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 ultrapassa a capacidade do concentrador são tratadas em todos os métodos aqui definidos através de penalização, por meio da atribuição de um custo elevado à solução. Com isso o método evita a escolha de soluções inviáveis.

A escolha de um modelo matemático também é de grande importância para o bom funcionamento do método proposto. Portanto, o modelo matemático usado neste trabalho para o problema CSAHLP foi baseado na proposta de Ernst e Krishnamoorthy (1999), por ser um modelo capaz de trabalhar com valores de  $n$  maiores que 40 sem exceder a capacidade de memória do computador.

$$f(x) = \text{Min} \sum_i \sum_k d_{ik} x_{ik} (\lambda E_i + \delta S_i) + \sum_i \sum_k \sum_l \alpha d_{kl} Y_{kl}^i + f_k x_{kk} \quad (2.11)$$

Sujeito a:

$$\sum_k x_{ik} = 1, \quad \forall i \in V \quad (2.3)$$

$$x_{kk} - x_{ik} \geq 0, \quad \forall i, k \in V, \quad (2.4)$$

$$\sum_l Y_{kl}^i - \sum_l Y_{lk}^i = E_i x_{ik} - \sum_j w_{ij} x_{jk} \quad \forall i, k \in V \quad (2.7)$$

$$\sum_i E_i x_{ik} \leq Q_k x_{kk} \quad \forall k \in V \quad (2.10)$$

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

$$Y_{kl}^i \geq 0 \quad \forall i, k, l \in V \quad (2.8)$$

Com a solução de referência  $\bar{x}$  para um dado parâmetro inteiro  $k$ , definiu-se a vizinhança  $N(\bar{x}, k)$  da solução de referência como o conjunto de soluções viáveis do CSAHLP que satisfazem às seguintes restrições de ramificação local:

- para o ramo esquerdo:  $\sum_{i,j \in S} (1-x_{ij}) + \sum_{i,j \notin S} x_{ij} \leq k$
- para o ramo direito:  $\sum_{i,j \in S} (1-x_{ij}) + \sum_{i,j \notin S} x_{ij} \geq k + 1$

### 4.3. Método de Solução para o Problema de Localização de Concentradores Capacitado com Alocação Múltipla

Para o PLCC com alocação múltipla foi desenvolvida uma estratégia de busca local, capaz de resolver o problema e gerar a solução inicial de referência.

O método de busca local (BL) proposto baseia-se no método VND (do inglês, *Variable Neighborhood Descent*), que explora o espaço de soluções através de trocas sistemáticas de estruturas de vizinhança. Sua metodologia consiste em, a partir de uma solução obtida pelo método CS para o CSAHLP, escolher uma estrutura de vizinhança e, a partir desta estrutura de vizinhança, gerar um vizinho desta solução. Se esse movimento gerar uma melhora na solução corrente esse movimento é aceito e retorna-se à primeira estrutura de vizinhança. Caso contrário, seleciona-se uma nova estrutura de vizinhança. O algoritmo do método VND é apresentando pela Figura 4.9.

1. **procedimento** Busca Local
2.     Seja  $s^0$  uma solução inicial e  $r$  o número de estruturas de vizinhança
3.      $s \leftarrow s^0$  { Solução corrente }
4.      $k \leftarrow 1$ ; { Tipo de estrutura de vizinhança }
5.     **enquanto** (Critério de parada não satisfeito) **faça**
6.         Gere um vizinho qualquer  $s' \in N(s)$
7.         **Se**  $(f(s') < f(s))$  **então**
8.              $s \leftarrow s'$ ;
9.              $k = 1$ ;
10.         senão
11.              $k \leftarrow k + 1$
12.     **fim-se**
13.     **fim-enquanto**
14.     **retorne**  $s$
15. **fim-procedimento**

Figura 4.9 – Algoritmo do método VND

Uma possível solução utilizada pelo método de busca local (BL) para o CMAHLP corresponde a um vetor formado por  $n^2$  posições, e cada uma delas corresponde a um par (origem, destino) que está associado à sua rota (*hub*-origem, *hub*-destino).

A Figura 4.10 apresenta uma possível solução para o problema, em que o número de clientes da rede  $n = 5$  e a quantidade de concentradores foi definido como  $p = 3$ . Os índices das posições do vetor  $V$  representados por  $g$  variam de 0 a 24 para representar as possíveis combinações de pares origem-destino e cada posição  $v[g]$  pode armazenar um valor representado por  $h$ , que pode variar de 0 a 8, para representar todas as possíveis rotas passando por um par de concentradores. Analisando, por exemplo, a posição  $g = 3$  do vetor  $V$ , em que seu valor associado é  $h = v[g] = 7$ , isso significa que alguma carga que sai do par origem-destino  $(i, j) = (1, 4)$  deve ser enviado via concentradores  $(k, l) = (3, 2)$ , pela seguintes relações:

- $(i(g), j(g)) = ((g \text{ div } n)+1, (g \% n)+1)$ , representando o par origem-destino;
- $(k(h), l(h)) = ((h \text{ div } p)+1, (h \% p)+1)$  para o arco de concentradores associado ao par origem destino.

Em que  $\%$  representa o resto da divisão e *div* a divisão inteira. Vale ressaltar que na tabela de possíveis concentradores, apresentada na Figura 4.10, os índices  $k$  e  $l$  representam o índice do vetor de concentradores escolhidos. Esse vetor armazena quais os pontos de 1 a  $n$  foram escolhidos para serem fixados como concentradores. Por exemplo, uma carga que sai do cliente 1 para o 4, passando pelo par  $(k, l) = (3, 2)$  na realidade irá passar pelos concentradores  $(5, 4)$  que são os valores reais correspondentes aos índices 2 e 3 da tabela de concentradores escolhidos conforme apresentado na Figura 4.10.

O método BL obtém os concentradores da solução inicial obtida pelo CS e em seguida associa cada posição do vetor  $V$  à rota que passa pelo par de concentradores mais próximos, conforme o algoritmo apresentado na Figura

4.11. Para calcular o custo  $C_{ijkm}$  basta somar os custos de coleta, transferência e distribuição multiplicadas pela distância correspondente ( $C_{ijkm} = \lambda d_{ik} + \alpha d_{km} + \delta d_{mj}$ ).

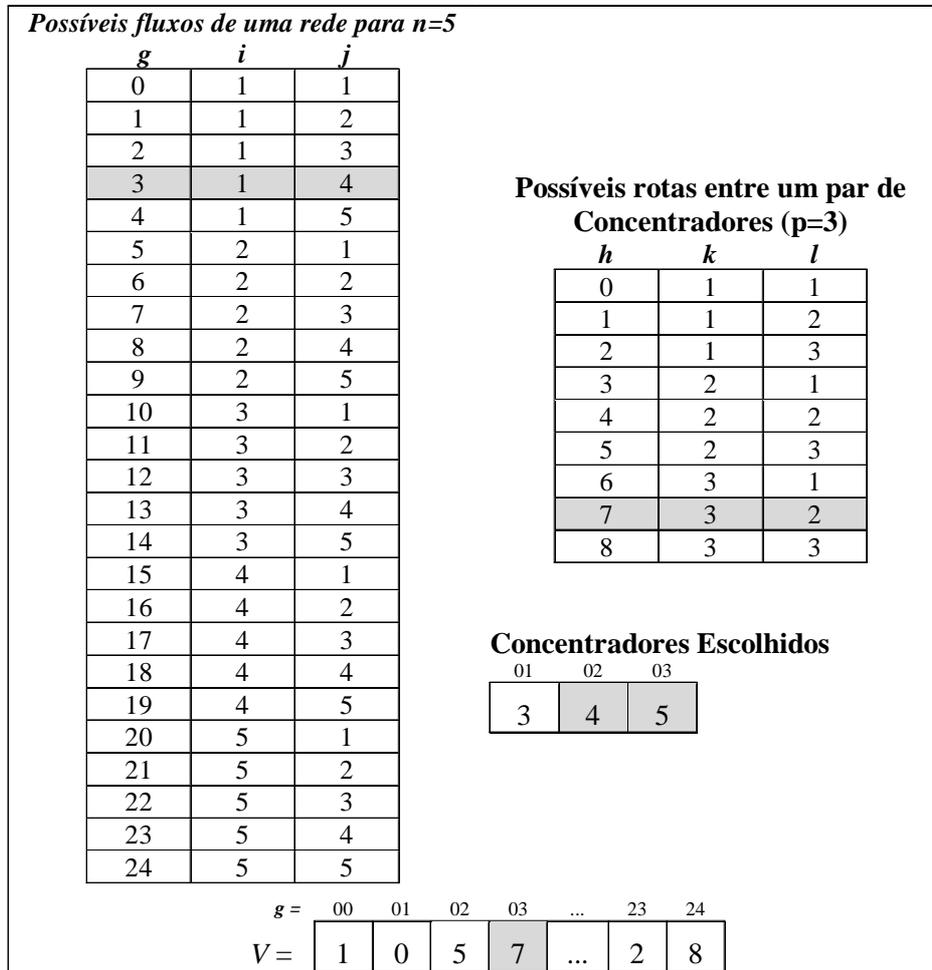


Figura 4.10 – Representação de uma solução CMAHLP

Após a obtenção do vetor  $V$ , com suas alocações às rotas dos concentradores mais próximos, este passa por uma função de avaliação responsável por pontuar a solução encontrada. Esta função obtém os valores da solução de acordo com a função-objetivo. A única exceção acontece quando uma solução gerada ultrapassa a restrição de capacidade de fluxo do concentrador. Neste caso, a função de avaliação atribui uma penalidade para que esta solução seja descartada no processo de busca do método.

```

1. algoritmo menorCusto( $i, j, p$ )
2.   menor  $\leftarrow \infty$ ;
3.   para  $k \leftarrow 0$  até  $p-1$  faça
4.     para  $l \leftarrow 0$  até  $p-1$  faça
5.       se ( $C_{ijkl} < \text{menor}$ ) então
6.         menor  $\leftarrow C_{ijkl}$ ;
7.         indK  $\leftarrow k$ ;
8.         indL  $\leftarrow l$ ;
9.       fim-se
10.    fim-para
11.  fim-para
12.  retorne (indK*p+indL);
13. fim-algoritmo

```

Figura 4.11 – Algoritmo para obtenção do menor custo  $C_{ijkl}$

Para o desenvolvimento do método de busca local foram definidas 4 vizinhanças:

- I. *Reallocate Hub*: escolhe um nó não-*hub* A, e nó *hub* B para trocar de status, fazendo com que A torne-se um concentrador e B um nó não-*hub*;
- II. *Add Hub*: esse procedimento adiciona um elemento não-*hub* ao conjunto de concentradores da solução;
- III. *Delete Hub*: escolhe um *hub* do conjunto de concentradores da solução para se transformar em elemento não-*hub*.
- IV. *Swap Route*: este movimento escolhe duas rotas do vetor  $V$  de forma aleatória e faz a troca entre elas.

Após os movimentos I, II ou III, todos os elementos do vetor  $V$  são realocados à rota de concentradores mais próxima.

O método LB desenvolvido tem como solução de referência inicial o resultado do CS com o método de busca local e as ramificações locais do LB usam a variável binária  $H$  do modelo (2.19-2.24) proposto por Ebery *et al.* (2000).

## 5 RESULTADOS COMPUTACIONAIS

A verificação da eficiência dos métodos implementados baseou-se no conjunto de dados AP (Ernst e Krishnamoorthy, 1996). O conjunto de dados AP corresponde a um conjunto de exemplares pequenos, de 10 a 50 nós, e um conjunto de exemplares grandes, de 100 a 200 nós. Os valores dos custos de coleta, transferência e distribuição são respectivamente 3.0, 0.75 e 2.0. Os fluxos não são simétricos, ou seja,  $w_{ij} \neq w_{ji}$ , e além disso, um cliente (*spoke*) pode enviar uma correspondência a si próprio ( $w_{ii} \neq 0$ ).

Os exemplares consideram dois tipos de custos fixos: apertado (T, “*Tight*”) e frouxo (L, “*Loose*”) e dois tipos de capacidades: apertado (T) e frouxo (L). Assim, para cada tamanho do problema, existem quatro exemplares: LL, LT, TL e TT. Nas tabelas de resultados computacionais a identificação do exemplar usa a notação nFC, onde n é o tamanho do problema, F é o tipo de custo fixo e C é o tipo de capacidade.

Neste trabalho apresentam-se os resultados para exemplares de tamanho  $n = 10, 20, 25, 40$  e  $50$  do conjunto de dados AP. Problemas com custos fixos do tipo T têm custos fixos altos para os nós com grandes fluxos. Isso torna difícil para o modelo escolher esses nós de alto volume como *hubs* (que de outra forma seriam candidatos naturais). Assim, estes problemas são mais difíceis de resolver. Problemas com os custos fixos do tipo L não apresentam esta tendência. Os testes foram executados em um computador com processador Intel Core i3 1.5 GHz, com 4 GB de memória RAM, sob o sistema operacional Linux.

Inicialmente, são apresentados os resultados dos testes realizados com as heurísticas candidatas como geradoras da solução de referência para o método LB. A Tabela 5.1 apresenta os resultados obtidos por estas heurísticas. Nesta tabela são apresentadas as seguintes informações:

- **Inst** - Identificação do exemplar;

- **MSC** - Valor da melhor solução conhecida para o problema (Ernst e Krishnamoorthy, 1999);
- **Gap** - Desvio percentual da melhor solução encontrada pela heurística (SH) em relação à melhor solução conhecida, ou seja:  $\text{Gap} = 100\% * (\text{SH} - \text{MSC}) / \text{MSC}$ ;
- **TE** - Tempo de execução total do algoritmo (em segundos), até que o critério de parada seja alcançado.

Os parâmetros e critérios de parada utilizados para as metaheurísticas apresentadas na Tabela 5.1 são:

- VNS: 1000 execuções sem melhora;
- *Simulated Annealing* (SA): Temperatura inicial ( $T_0 = 100000$ ), taxa de resfriamento ( $\alpha = 0,95$ );
- Algoritmo Genético (AG): Tamanho da população ( $\text{POP} = 20$ ), número de gerações = 1000, porcentagem de *crossover* = 80%, porcentagem de mutação = 20%, porcentagem de elite = 20%;
- *Clustering Search com Simulated Annealing* (CSSA): número de *clusters* = 20, raio do *cluster* = 5; índice de ineficácia = 30; método de busca local = 10000 execuções sem melhora, mantidos os parâmetros utilizados no método SA.

A Tabela 5.1 apresenta os resultados obtidos pelas seguintes heurísticas, consideradas no Capítulo 4: VNS (*Variable Neighborhood Search*), AG (Algoritmo Genético), SA (*Simulated Annealing*), CSSA (*Clustering Search* usando o algoritmo SA como gerador de soluções). Esta tabela apresenta também os resultados das heurísticas RD-EK (*Random Descent*) e SA-EK (*Simulated Annealing*) propostas por Ernst e Krishnamoorthy (1999).

Tabela 5.1 – Resultados das heurísticas candidatas

Inst	MSC	VNS		AG		SA		CSSA		RD-EK		SA-EK	
		Gap	TE										
10LL	224250,05	0,77	0,03	0,00	0,07	3,20	0,02	0,77	0,02	0,00	0,03	0,00	0,19
10LT	250992,26	1,30	0,03	0,00	0,12	5,25	0,02	3,63	0,02	0,00	0,03	0,00	0,21
10TL	263399,94	5,39	0,00	0,00	0,04	0,00	0,02	0,43	0,01	0,00	0,03	0,00	0,15
10TT	263399,94	0,00	0,03	4,65	0,07	3,50	0,02	0,43	0,02	0,00	0,03	0,00	0,17
20LL	234690,96	0,00	0,42	2,03	0,09	11,81	0,06	0,00	0,06	0,00	0,34	0,00	0,65
20LT	253517,40	0,00	0,55	6,74	0,12	0,00	0,06	0,00	0,12	0,00	0,26	0,00	0,63
20TL	271128,18	2,60	0,03	0,00	0,11	2,60	0,01	2,60	0,03	0,00	0,28	0,00	0,53
20TT	296035,40	0,85	0,59	0,85	0,13	9,13	0,06	3,83	0,06	0,00	0,25	0,00	0,66
25LL	238977,95	0,00	0,93	22,63	0,07	2,60	0,05	2,60	0,09	0,00	0,78	0,00	1,05
25LT	276372,50	1,31	1,69	5,77	0,09	4,56	0,11	1,31	0,10	0,00	0,63	0,00	1,07
25TL	310317,64	0,00	1,01	0,00	0,04	26,00	0,02	0,00	0,09	0,00	0,77	0,00	1,12
25TT	348369,15	0,00	1,53	16,77	0,05	1,51	0,12	1,06	0,10	0,00	0,63	0,00	1,15
40LL	241955,71	0,00	8,46	8,83	0,06	0,00	0,23	0,07	0,28	0,00	3,79	0,00	2,66
40LT	272218,32	0,00	12,63	23,85	0,04	5,32	0,32	7,31	0,32	0,00	4,07	0,00	2,78
40TL	298919,01	0,00	6,75	2,22	0,03	0,00	0,24	0,14	0,28	0,00	4,37	0,00	2,92
40TT	354874,10	2,40	14,62	17,89	0,08	2,44	0,36	2,44	0,45	0,00	4,17	0,00	3,13
50LL	238520,59	0,00	20,96	7,14	0,10	0,00	0,36	0,00	0,37	0,00	11,41	0,00	4,30
50LT	272897,49	0,00	43,81	45,36	0,12	0,91	0,51	2,38	0,68	0,00	11,43	0,00	4,88
50TL	319015,77	0,00	17,02	6,25	0,05	2,00	0,35	0,76	0,46	0,00	11,45	0,00	5,63
50TT	417440,99	1,64	40,17	16,04	0,19	1,77	0,91	1,65	1,62	0,45	11,32	0,71	5,91
	<b>Média</b>	<b>0,81</b>	<b>8,56</b>	<b>9,35</b>	<b>0,08</b>	<b>4,13</b>	<b>0,19</b>	<b>1,57</b>	<b>0,27</b>	<b>0,02</b>	<b>3,30</b>	<b>0,04</b>	<b>1,99</b>

Pelos resultados mostrados na Tabela 5.1 verifica-se que o método CSSA é capaz de gerar boas soluções, em um tempo de execução reduzido. Assim, neste trabalho, o método CSSA foi escolhido para trabalhar com os métodos propostos, tanto para gerar soluções de referência iniciais, como para obter limitantes superiores para as soluções dos problemas considerados.

Dado que o problema com alocação única é uma versão mais restrita do problema com alocação múltipla, a mesma estratégia (CSSA) será utilizada para a solução do CMAHLP. Isso é possível porque uma solução para o problema com alocação simples é também uma solução do problema com alocação múltipla. No entanto, para melhorar a qualidade desta solução para problemas com alocação múltipla, acrescentou-se ao método CSSA uma estratégia de busca local, para explorar os detalhes da solução gerada para o problema com alocação única considerando a representação de soluções para o problema com alocação múltipla apresentado na Seção 4.3. Este novo método foi denominado CSSA+BL.

### 5.1. Resultados para o Problema de Localização de Concentradores Capacitado com Alocação Única (CSAHLP)

Para o problema de localização de concentradores capacitados com alocação única (CSAHLP), foram desenvolvidos os seguintes métodos de solução:

- CPLEX-CS: Método *branch-and-cut* utilizado pelo *solver* CPLEX, que utiliza o método CSSA foi utilizado para gerar um única solução inicial de referência para o início da busca do método.
- LB-CS: Método *Local Branching* em que o processo de busca faz uso do CSSA como método gerador de limitantes superiores, assim como no método CPLEX-CS.
- LB-ALEAT: Consiste do método LB que utiliza um procedimento gerador de solução inicial aleatória.

Nas tabelas de resultados mostradas a seguir, utiliza-se o símbolo “\*\*” para indicar que o método terminou a busca pela solução de modo prematuro, devido a estouro de memória.

A Tabela 5.2 mostra os resultados obtidos com os métodos implementados. Essa tabela inclui também os resultados obtidos pelo *solver* CPLEX e pelos métodos N-EK (método CSAHLP-N) e PN-EK (método CSAHLP-PN), propostos por Ernst e Krishnamoorthy (1999). Para estes testes foram utilizados:  $k = 15$  nas restrições *local branching* e critério de parada do CS igual a 100 iterações sem melhora.

Tabela 5.2 – Resultados obtidos para o PLCC com alocação única

Inst	MSC	CPLEX		CPLEX-CS		LB-CS		LB-ALEAT		N-EK		PN-EK	
		Gap	TE	Gap	TE	Gap	TE	Gap	TE	Gap	TE	Gap	TE
10LL	224250,05	0,00	0,07	0,00	0,17	0,00	0,06	0,00	0,08	0,57	0,27	0,57	0,27
10LT	250992,26	0,00	0,16	0,00	0,14	0,00	0,09	0,00	0,14	1,17	0,22	0,37	0,14
10TL	263399,94	0,00	0,42	0,00	0,14	0,00	0,06	0,00	0,09	0,88	0,29	0,88	0,30
10TT	263399,94	0,00	0,10	0,00	0,08	0,00	0,06	0,00	0,08	0,88	0,25	0,88	0,22
20LL	234690,96	0,00	0,37	0,00	0,30	0,00	0,45	0,00	0,56	0,09	2,49	0,09	2,18
20LT	253517,40	0,00	1,94	0,00	0,82	0,00	0,46	0,00	1,37	2,53	5,81	2,53	5,75
20TL	271128,18	0,00	0,41	0,00	0,35	0,00	0,44	0,00	0,63	0,19	2,42	0,19	2,39
20TT	296035,40	0,00	2,61	0,00	1,02	0,00	1,54	0,00	4,71	2,38	6,68	1,61	5,68
25LL	238977,95	0,00	1,39	0,00	1,67	0,00	1,80	0,00	2,33	0,37	13,34	0,37	14,22
25LT	276372,50	0,00	5,93	0,00	2,94	0,00	2,72	0,00	9,99	2,39	31,38	1,84	25,77
25TL	310317,64	0,00	0,98	0,00	0,88	0,00	0,94	0,00	1,10	0,31	7,16	0,31	6,78
25TT	348369,15	0,00	4,13	0,00	4,01	0,00	0,75	0,00	6,13	4,33	28,89	4,33	17,50
40LL	241955,71	0,00	8,45	0,00	4,62	0,00	5,87	0,00	45,92	0,65	67,80	0,65	61,81
40LT	272218,32	0,00	14,46	0,00	7,59	0,00	10,00	0,00	122,88	1,62	91,29	1,62	68,08
40TL	298919,01	0,00	5,00	0,00	4,61	0,00	8,69	0,00	41,35	0,07	62,80	0,07	72,56
40TT	354874,10	0,00	58,98	0,00	55,85	0,00	110,34	0,00	158,69	2,96	304,73	2,96	225,00
50LL	238520,59	0,00	9,60	0,00	9,44	0,00	12,85	0,00	59,19	0,27	164,59	0,27	179,56
50LT	272897,49	0,00	91,95	0,00	82,89	0,00	75,44	0,00	984,01	1,83	808,18	1,82	664,49
50TL	319015,77	0,00	72,49	0,00	58,51	0,00	38,00	3,77	344,88	2,68	448,70	2,68	427,33
50TT	417440,99	0,15	24878,64*	0,00	18126,13	0,00	971,85	0,00	3392,73	5,27	*	5,27	*
	<b>Média</b>	<b>0,01</b>	<b>1257,90</b>	<b>0,00</b>	<b>918,11</b>	<b>0,00</b>	<b>62,12</b>	<b>0,19</b>	<b>258,84</b>	<b>1,57</b>	<b>107,75</b>	<b>1,47</b>	<b>93,69</b>

Pelos resultados mostrados na Tabela 5.2 pode-se observar que os métodos CPLEX, N-EK e PN-EK não conseguiram chegar ao fim para o exemplar 50TT, devido ao estouro de memória. Os resultados desta tabela mostram que o método LB-CS obtém a melhor solução conhecida para todos os exemplares ( $gap = 0,00$ ). Observa-se também que o método LB-ALEAT obtém a melhor solução para quase todos os exemplares (excetuando-se o exemplar 50TL), mas a um custo computacional maior, demonstrando que uma boa solução inicial de referência é importante para o desempenho do método. Os resultados desta tabela mostram que, mesmo para o método CPLEX, uma boa solução inicial é importante, pois para quase todos os casos (18 em 20), o tempo computacional do método CPLEX-CS é menor do que o do método CPLEX. Também é possível observar o bom desempenho do método CPLEX-CS (também obtém  $gap = 0,00$  para todos os exemplares), embora, em geral, requeira um tempo computacional maior do que o método LB-CS (10 em 20

casos, com média de cerca de 918 segundos, contra cerca de 62 segundos para o método LB-CS).

A Tabela 5.3 apresenta mais detalhes sobre os resultados obtidos pelos métodos CPLEX-CS e LB-CS. A coluna **Ref** mostra o valor da solução inicial de referência (obtida com o método CSSA), **GapI** corresponde ao *gap* inicial, em relação à melhor solução conhecida, **Cuts** corresponde ao número de cortes utilizados, e os demais campos repetem os resultados já apresentados na Tabela 5.2.

Tabela 5.3 – Detalhes dos resultados obtidos pelos métodos CPLEX-CS e LB-CS

Inst	MSC	Ref	LB-CS				CPLEX-CS			
			GapI	Cuts	Gap	TE	GapI	Cuts	Gap	TE
10LL	224250,05	225979,17	0,77	0	0,00	0,06	0,77	26	0,00	0,17
10LT	250992,26	260108,70	3,63	30	0,00	0,09	3,63	75	0,00	0,14
10TL	263399,94	264543,96	0,43	0	0,00	0,06	0,43	45	0,00	0,14
10TT	263399,94	264543,96	0,43	0	0,00	0,06	0,43	38	0,00	0,08
20LL	234690,96	234690,96	0,00	0	0,00	0,45	0,00	0	0,00	0,30
20LT	253517,40	253517,40	0,00	0	0,00	0,46	0,00	154	0,00	0,82
20TL	271128,18	278167,72	2,60	5	0,00	0,44	2,60	5	0,00	0,35
20TT	296035,40	307371,90	3,83	178	0,00	1,54	3,83	124	0,00	1,02
25LL	238977,95	245182,91	2,60	132	0,00	1,80	2,60	132	0,00	1,67
25LT	276372,50	279990,93	1,31	276	0,00	2,72	1,31	275	0,00	2,94
25TL	310317,64	310317,65	0,00	0	0,00	0,94	0,00	14	0,00	0,88
25TT	348369,15	352069,91	1,06	157	0,00	0,75	1,06	197	0,00	4,01
40LL	241955,71	242114,04	0,07	86	0,00	5,87	0,07	144	0,00	4,62
40LT	272218,32	292104,90	7,31	291	0,00	10,00	7,31	301	0,00	7,59
40TL	298919,01	299339,78	0,14	56	0,00	8,69	0,14	23	0,00	4,61
40TT	354874,10	363542,94	2,44	359	0,00	110,34	2,44	400	0,00	55,85
50LL	238520,59	238520,59	0,00	0	0,00	12,85	0,00	42	0,00	9,44
50LT	272897,49	279400,68	2,38	506	0,00	75,44	2,38	610	0,00	82,89
50TL	319015,77	321449,00	0,76	20	0,00	38,00	0,76	89	0,00	58,51
50TT	417440,99	424337,65	1,65	1181	0,00	971,85	1,65	1719	0,00	18126,13
		<b>Média</b>	<b>1,57</b>	<b>163,85</b>	<b>0,00</b>	<b>62,12</b>	<b>1,5705</b>	<b>220,65</b>	<b>0,00</b>	<b>918,108</b>

Pelos resultados mostrados na Tabela 5.3 pode-se verificar que, partindo-se da mesma solução inicial de referência, o método LB-CS é mais eficiente do que o CPLEX-CS, exigindo menos cortes para obter soluções de boa qualidade para o problema de localização de concentradores com alocação única.

Por fim, procurou-se executar os métodos CPLEX-CS e LB-CS para exemplares maiores. Para isto, foram gerados exemplares com 100 e 200 nós, com base no conjunto de dados AP. Deve-se observar que, apesar do nome apresentado na Tabela 5.4 ser o mesmo que o apresentado por Ebery *et al.* (2000), não se tratam dos mesmos exemplares. A solução inicial de referência Ref apresentada na Tabela 5.4 foi obtida por meio do CSSA e o Gap apresentado corresponde ao desvio percentual da solução encontrada em relação ao limitante inferior de relaxação de Programação Linear obtido pelo CPLEX, um vez que a solução ótima não é conhecida. Na Tabela 5.4 o símbolo “-” indica que o método não foi capaz de obter resultado devido a falta de memória.

Tabela 5.4 – Resultados dos métodos LB-CS e CPLEX-CS para exemplares maiores

Inst	Ref	LB-CS			CPLEX-CS		
		Sol	Gap	TE	Sol	Gap	TE
100LL	244310,40	243254,16	0,00	1987,86	243254,16	0,00	3814,45
100LT	269295,61	263102,54	1,24	2522,18	-	-	-
100TL	359953,06	359146,48	0,30	1470,30	359146,48	0,30	1251,04
100TT	548131,76	488479,47	0,11	3194,27	-	-	-

Os resultados da Tabela 5.4 mostram que o método CPLEX-CS foi capaz de resolver apenas o exemplar 100LL. Para todos os demais exemplares, o método CPLEX-CS não foi capaz de chegar a uma solução devido ao estouro de memória. O método LB-CS foi capaz de resolver os exemplares de 100 nós e resolve o exemplar 100LL em aproximadamente metade do tempo gasto pelo método CPLEX-CS.

## 5.2. Resultados para o Problema de Localização de Concentradores Capacitado com Alocação Múltipla (CMAHLP)

Como comentado anteriormente, para a geração de soluções de referência para o problema com alocação múltipla foi utilizada a heurística CSSA+BL, que corresponde à heurística CSSA com o acréscimo de uma estratégia de busca

local (BL) para a melhora da solução obtida para o problema com alocação única. Neste caso, o valor de  $k$  usado nas restrições *local branching* foi fixado em 5 e o critério de parada foi fixado em 1000 iterações.

O parâmetros do CSSA+BL são os mesmos obtidos pelo CSSA, com a adição do critério de parada para o método de Busca Local: 1000 iterações.

Tabela 5.5 – Resultados obtidos com a heurística CSSA

Inst	MSC	CSSA+BL		E-R		E-U	
		Gap	TE	Gap	TE	Gap	TE
10LL	221032,73	4,71	0,04	0,00	0,05	0,00	0,61
10LT	246495,04	0,92	0,06	0,00	0,04	0,00	0,62
10TL	257558,08	1,68	0,03	0,00	0,06	0,00	0,59
10TT	257558,08	2,71	0,04	0,00	0,04	0,00	0,53
20LL	230385,45	1,87	0,15	0,00	0,30	0,00	4,24
20LT	246433,75	2,87	0,19	0,00	0,22	0,00	3,77
20TL	266877,48	0,00	0,09	1,72	0,37	0,00	4,57
20TT	287574,76	3,82	0,22	1,13	0,45	0,00	4,73
25LL	233636,70	2,29	0,33	0,18	0,55	0,00	8,51
25LT	263760,55	4,78	0,33	0,00	0,46	0,00	8,26
25TL	305982,08	1,42	0,20	0,71	1,47	0,00	10,85
25TT	333221,74	5,73	0,29	4,37	1,23	0,00	9,98
40LL	238313,46	1,53	0,65	0,00	2,01	0,00	37,71
40LT	261547,13	0,00	0,88	0,18	2,21	0,00	36,46
40TL	297204,12	0,00	0,68	0,66	4,65	0,00	47,73
40TT	339965,86	6,89	2,13	1,27	9,66	0,00	52,83
50LL	234399,92	0,00	0,96	0,00	6,29	0,00	92,83
50LT	264999,71	3,92	2,05	0,22	5,76	0,00	88,72
50TL	314845,55	0,00	1,33	1,45	16,92	0,00	119,93
50TT	391287,29	8,51	2,24	5,84	20,15	1,36	120,59
	<b>Média</b>	<b>2,68</b>	<b>0,64</b>	<b>0,89</b>	<b>3,64</b>	<b>0,07</b>	<b>32,70</b>

A Tabela 5.5 mostra as soluções obtidas pela heurística CSSA-BL e por dois métodos apresentados por Ebery *et al.* (2000): E-R, uma heurística que utiliza um procedimento de realocação de roteamento, e E-U, que utiliza um procedimento denominado *multicommodity flow procedure*. Nesta tabela, a coluna MSC corresponde às melhores soluções da literatura, reportadas por Ebery *et al.* (2000), Gap corresponde ao desvio percentual da solução obtida em relação à melhor solução conhecida e TE corresponde ao tempo de execução (em segundos). Os resultados da Tabela 5.5 mostram que os métodos E-R e E-U são melhores em termos de qualidade da solução. No

entanto, o método CSSA-BL obtém soluções razoáveis em um tempo computacional bem menor, uma vez que a solução inicial tem como objetivo fornecer uma solução boa, viável para acelerar o processo de busca da técnica LB.

A Tabela 5.6 apresenta os resultados obtidos para o PLCC com alocação múltipla pelos métodos CPLEX, CPLEX-CS, LB-CS e LB-ALEAT. Nesta tabela, para o método CMAHLP-N o gap utilizado apresentado por Ebery *et al.* (2000) corresponde ao gap em relação ao limite de relaxação linear obtido pelo CPLEX.

Tabela 5.6 – Resultados obtidos para o PLCC com alocação múltipla

Inst	CPLEX		CPLEX-CS		CMAHLP-N		LB-CS		LB-ALEAT	
	Gap	TE	Gap	TE	Gap	TE	Gap	TE	Gap	TE
10LL	0,00	3,47	0,00	0,59	2,72	1,62	0,00	0,41	0,00	0,45
10LT	0,00	3,72	0,00	0,75	3,47	2,17	0,00	0,71	0,00	0,76
10TL	0,00	1,75	0,00	0,46	4,05	3,30	0,00	0,39	0,00	0,44
10TT	0,00	1,44	0,00	0,42	2,97	1,40	0,00	0,34	0,00	0,35
20LL	0,00	43,65	0,00	22,98	1,14	70,48	0,00	5,15	0,00	27,57
20LT	0,00	27,33	0,00	13,26	2,60	146,63	0,00	17,95	0,00	13,34
20TL	0,00	22,06	0,00	10,90	4,10	97,38	0,00	5,16	0,00	18,83
20TT	0,00	16,16	0,00	6,06	2,89	107,92	0,00	5,15	0,00	8,77
25LL	0,00	104,84	0,00	160,10	1,31	457,96	0,00	5,36	0,00	148,42
25LT	0,00	232,16	0,00	175,79	2,89	805,75	0,00	188,94	0,00	194,04
25TL	0,00	93,02	0,00	28,03	3,29	798,70	0,00	5,36	0,00	38,13
25TT	0,00	38,20	0,00	27,11	4,33	866,99	0,00	31,96	0,00	22,22
40LL	0,00	2976,18	0,00	2911,17	2,00	11687,75	0,00	707,83	0,00	3756,83
40LT	0,00	1242,79	0,00	1128,48	2,82	39722,20	0,00	7,79	0,00	843,62
40TL	0,00	167,92	0,00	161,93	1,84	8966,49	0,00	7,67	0,00	182,28
40TT	0,00	131,19	0,00	61,59	2,43	5278,76	0,00	66,52	0,00	74,58
50LL	0,00	18004,74	0,00	17707,96	1,43	60640,03	0,00	12,56	0,00	22601,79
50LT	0,00	11090,23	0,00	7219,66	4,23	*	0,00	7240,89	0,00	7818,52
50TL	0,00	985,36	0,00	657,53	3,57	79762,58	0,00	12,63	0,00	672,02
50TT	0,00	435,83	0,00	491,62	2,68	68887,91	0,00	500,57	0,00	483,05
<b>Média</b>	<b>0,00</b>	<b>1781,10</b>	<b>0,00</b>	<b>1539,32</b>	<b>2,84</b>	<b>14647,69</b>	<b>0,00</b>	<b>441,17</b>	<b>0,00</b>	<b>1845,30</b>

Pelos resultados da Tabela 5.6, observa-se que também para o caso de alocação múltipla, o método LB-CS obtém, em geral, os melhores resultados. Observa-se também, pela comparação dos resultados obtidos pelos métodos LB-CS e LB-ALEAT, que uma boa solução inicial é importante para que o método obtenha uma boa solução em menor tempo computacional. A mesma

observação pode ser feita em relação aos métodos CPLEX e CPLEX-CS, embora, neste caso, o efeito não ter sido tão significativo.

Na Tabela 5.7 são apresentados os resultados obtidos para o PLCC com alocação múltipla pelo método LB-CS, considerando os mesmos exemplares da Tabela 5.4. Observa-se que o método CPLEX-CS não foi capaz de resolver estes exemplares devido a problemas de estouro de memória. Para a Tabela 5.7, o Gap apresentado também corresponde ao desvio percentual da solução encontrada em relação ao limitante inferior de relaxação de Programação Linear obtido pelo CPLEX.

Tabela 5.7 – Resultados para exemplares maiores do PLCC com alocação múltipla

Inst	Ref	LB-CS		
		Sol	Gap	TE
100LL	244319,42	238480,45	128,38	613,19
100LT	266206,35	252851,83	54,53	800,51
100TL	350710,40	350709,52	42,24	739,94
100TT	526839,56	477443,12	46,73	702,45

A Tabela 5.7 mostra que o método LB-CS foi capaz de melhorar as soluções iniciais de referência para exemplares de tamanho 100. Deve-se observar, no entanto que, embora o método LB-CS tenha conseguido melhorar a solução inicial de referência para o problema com alocação múltipla, também não conseguiu chegar ao final da busca devido a problemas com estouro de memória.

Os resultados dos testes computacionais apresentados neste capítulo mostram que a estratégia de solução proposta neste trabalho, o método LB-CS, é efetivo tanto para resolver o PLCC com alocação única quanto o PLCC com alocação múltipla, superando o *solver* CPLEX mesmo quando este otimizador é utilizado a partir de uma boa solução inicial, como no método CPLEX-CS.

## 6 CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS

O problema de localização de concentradores capacitado 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.

O problema, no entanto, é conhecido ser da classe NP-difícil (GAREY e JOHNSON, 1979) e algoritmos exatos para determinar a solução ótima do problema constituem um grande desafio, pois precisam utilizar métodos que exigem grande esforço computacional e podem ser impraticáveis para exemplares do problema de grandes dimensões.

Baseado nos estudos e testes realizado sobre a técnica LB é possível identificar que uma das vantagens da utilização desse método é que, com a inserção de cortes LB, a busca por soluções ocorre de maneira mais rápida, por meio da exploração de vizinhanças menores de uma determinada solução. O método B&B também pode iniciar a busca com soluções iniciais de referência, porém sem o uso de cortes LB a busca por soluções pode levar muito mais tempo.

Sabe-se que um método exato gasta a maior parte do seu tempo de processamento tentando provar que a solução encontrada é ótima. Os testes realizados com o CPLEX neste trabalho mostram o grande tempo de execução gasto por este *solver* para a solução dos problemas menores (já que para os problemas maiores o CPLEX nem foi capaz de encontrar uma solução). Mostrou-se também, com o método CPLEX-CS, que bons limitantes superiores auxiliam o CPLEX na solução de problemas de localização de concentradores. No entanto, ainda assim, bons limitantes inferiores são necessários para acelerar o processo de busca e auxiliar na prova de otimalidade das soluções.

Existem muitos trabalhos que apresentam heurísticas e meta-heurísticas para o PLC com alocação única, no entanto, somente foi encontrado um trabalho que relaciona duas heurísticas para o problema com alocação múltipla. Assim, inicialmente, a determinação de uma boa heurística para o CMAHLP foi bem trabalhosa. A representação da solução utilizada, proposta na Seção 4.3, é capaz de trabalhar com detalhes de uma rede com alocação múltipla, no entanto, esses detalhes comprometeram muito a qualidade final das soluções, devido ao tamanho da tabela de possíveis rotas, que cresce conforme o tamanho de  $n$  e  $p$ . Para contornar este problema, a estratégia CSSA do problema com alocação única foi utilizada associada a um método de busca local para resolver o CMAHLP. Com essa ideia foi desenvolvida uma heurística denominada CSSA+BL para o PLCC com alocação múltipla, capaz de fornecer limitantes superiores razoáveis para o problema. Portanto, a representação de soluções proposta na Seção 4.3 foi utilizada como estratégia para melhorar a solução obtida pelo CSSA, com a finalidade de encontrar uma solução boa para o CMAHLP.

Dentre os métodos desenvolvidos, tanto para o CSAHLP quanto para o CMAHLP, o método LB-CS é, na maioria dos casos, mais rápido e fornece soluções de melhor qualidade. Esta conclusão comprova, portanto, o objetivo deste trabalho em demonstrar que o método *Local Branching* com cortes genéricos é efetivo para problemas de localização de concentradores.

Como mencionado anteriormente a determinação de bons limitantes é muito importante para o melhor desempenho do método. Neste trabalho não foram considerados os limitantes inferiores. Com o uso dos limitantes superiores e inferiores a estratégia de busca por soluções poderia se tornar ainda melhor. Assim, sugere-se como continuação do trabalho, o estudo e inserção de limitantes inferiores na estratégia LB, para os problemas estudados.

O problema CMAHLP é um problema pouco estudado. Neste trabalho, foi possível identificar que a estratégia CSSA desenvolvida para o CSAHLP, seria

capaz de gerar soluções iniciais razoáveis para o CMAHLP, pois o uso direto da representação de soluções para o problema com alocação múltipla mostrou-se muito ineficaz para a busca de soluções. No entanto, os resultados mostraram que mesmo utilizando uma solução inicial apenas razoável, o método LB-CS foi capaz de obter soluções de melhor qualidade para os problemas apresentados. Mas este trabalho mostrou também que uma boa solução inicial influencia positivamente no resultado final do método. Assim, sugere-se um estudo mais aprofundado no sentido de desenvolver uma heurística de melhor qualidade para a geração de soluções iniciais para os problemas com alocação múltipla.

Outra possibilidade de continuação deste trabalho consiste em explorar as diversas formas de realizar as ramificações da árvore de enumeração, por exemplo, testar o que ocorre quando são exploradas vizinhanças diferentes, alterando-se o valor de  $k$  nas restrições de ramificação local de forma dinâmica, conforme o nível da árvore. Outra possibilidade de continuação deste trabalho é o desenvolvimento do método LB como método exato. Neste caso, será necessário um estudo sobre a melhor forma do método explorar todas as ramificações da árvore de enumeração *Local Branching*.



## REFERÊNCIAS BIBLIOGRÁFICAS

ABDINNOUR-HELM, S. A hybrid heuristic for the uncapacitated hub location problem. **European Journal of Operational Research**, v. 106, p. 489-499, 1998.

ALUMUR, S.; KARA, B. Y. Network hub location problems: the state of the art. **European Journal of Operational Research**, v. 190, n. 1, p. 1-21, Oct., 2008.

ARAÚJO, S.A.; ARENALES, M.N.; CLARK, A.R. Joint rolling-horizon scheduling of materials processing and lot-sizing with sequence-dependent setups. **Journal of Heuristics**, v. 13, p. 337-358, 2007.

AYKIN, T. Lagrangian relaxation based approaches to capacitated hub-and-spoke network design problem. **European Journal of Operational Research**, v. 79, n. 33, p. 501-523, 1994.

AYKIN, T. Networking policies for hub-and-spoke systems with application to the air transportation system. **Transportation Science**, v. 29, n. 3, p. 201-221, 1995a.

CAMPBELL, J.F. Integer programming formulations of discrete hub location problems. **European Journal of Operational Research**, v. 72, p. 387-405, 1994.

CARELLO, G.; DELLACROCE, F.; GHIRARDI, M.; TADEI, R. Solving the hub location problem in telecommunication network design: a local search approach. **Networks**, v. 44, n. 2, p. 94-105, 2004.

CHAVES, A.A. **Uma meta-heurística híbrida com busca por agrupamentos aplicada à problemas de otimização combinatória**. 2009. 197 p. (INPE-15685-TDI/1459). Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, 2009.

Disponível em: <<http://urlib.net/8JMKD3MGP8W/34NDEC8>>. Acesso em: 23 jul. 2014.

CHEN, J.F. A hybrid heuristic for the uncapacitated hub location problem. **Omega, The International Journal of Management Science**, v. 35, p. 211-220, 2007.

CHEN, J.F. A note on solution of the uncapacitated single allocation p-hub median problem. **Journal of the Chinese Institute of Industrial Engineers**, v. 25, n. 1, p. 11-17, 2008.

CUNHA, C.B.; SILVA, M.R. 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, 2007.

DELLA CROCE, F.; SALASSA, F. A variable neighborhood search based matheuristic for nurse rostering problems. **Annals of Operations Research**, Springer, v. 218, n. 1, p. 185-199, 2014.

EBERY, J. Solving a large Single allocation p-hub problems with two or three hubs. **European Journal of Operational Research**, v. 128, n. 2, p. 447-458, 2001.

EBERY, J; KRISHNAMOORTHY, M; ERNST, A; BOLAND, N. The Capacitated multiple allocation hub location problem: Formulations and algorithms. **European Journal of Operational Research**, v. 120, p. 614–631, 2000.

ERNST, A.T., JIANG, H., KRISHNAMOORTHY, M. **Reformulations and computational results for uncapacitated single and multiple allocation hub covering problems**. Clayton South Australia: Csiro Mathematical and Information Sciences, 2005. Unpublished Report.

ERNST, A.T.; KRISHINAMOORTHY, M. Efficient algorithms for the uncapacitated single allocation p-hub median problem. **Location Science**, v. 4, n. 3, p. 139-154, 1996.

ERNST, A.T.; KRISHNAMOORTHY, M. An exact solution approach based on shortest-paths for P-Hub median problems. **INFORMS Journal on Computing**, v.10, n. 2, p. 149-162, Feb., 1998.

ERNST, A.T.; KRISHNAMOORTHY, M. Solution algorithms for the capacitated single allocation hub location problem. **Annals of Operations Research**, v. 86, p. 141-159, 1999.

ESCUADERO, L.F.; SALMERON, J. On a fix-and-relax framework for a class of project scheduling problems. **Annals of Operations Research**, v. 140, n. 1, p. 163-188, 2005.

FARAHANI, R. Z.; HEKMATFAR, M; ARABANI, A. B.; NIKBAKHSH, E. Hub location problems: a review of models, classification, solution techniques, and applications. **Computers & Industrial Engineering**, v. 64, n. 4, p. 1096-1109, 2013.

FERREIRA, D.; MORABITO, R.; RANGEL, S. Relax and fix heuristics to solve one-stage one-machine lot-scheduling models for small-scale soft drink plants. **Computers & Operations Research**, v. 37, n. 4, p. 684-691, abr. 2010.

FISCHETTI, M.; LODI, A. Local branching, **Mathematical Programming**, v. 98, n. 1-3, p. 23-47, 2003.

FISCHETTI, M.; POLO, C.; SCANTAMBURLO, M. A Local branching heuristic for mixed-integer programs with 2-level variables, with an application to a telecommunication network design problem. **Networks**, v. 44, n. 2, p. 61-72, 2004.

GAREY, M.R.; JOHNSON, D.S. **Computers and Intractability**: a guide to the theory of NP-completeness. San Francisco: W.H. Freeman, 1979.

GLOVER, F. Tabu Search and adaptive memory programming: advances, applications and challenges. In: BARR, R.S.; HELGASON, R.V.; KENNINGTON, J.L. (eds). **Interfaces in Computer Science and Operational Research**. Kluwer, 1996. p. 1-75.

GOMORY, R. E. Outline of an algorithm for integer solutions to linear programs. **Bulletin of the American Mathematical Society**, v. 64, n. 5, p. 275-278, 1958.

HAMACHER, H. W.; T. MEYER. **Hub cover and hub center problems**. Technical Report, Technische Universität Kaiserslautern. Report in *Wirtschaftsmathematik* n. 98/2006.

HAMACHER, H.W.; LABBÉ, M.; NICKEL, S.; SONNEBORN, T. Adapting polyhedral properties from facility to hub location problems. **Annals of Operations Research**, v. 145, p. 104-116, 2004.

HANSEN, P.; MLADENOVIC, N.; UROSEVIC, D. Variable neighborhood search and local branching. **Computers and Operations Research**, v. 33, p. 3034-3045, 2006.

HOLLAND, J. **Adaptation in natural and artificial systems**. Ann Arbor, Michigan: University of Michigan Press, 1975.

IBM ILOG Inc. **Solver CPLEX**. 2013. Disponível em: <http://www.ilog.com/products/cplex/>. Acesso em: 23 jul. 2014.

KAWAMURA, M.S.; RANCONI, D.P. Aplicação da heurística relax-and-fix no problema de dimensionamento e sequenciamento de lotes de produção em máquinas distintas em paralelo. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 42., 2010, Bento Gonçalves, RS. **Anais...** Bento Gonçalves: SOBRAPO/UFSM, 2010.

- KIRKPATRICK, S.; GELATT JUNIOR, C.D.; VECCHI, M.P. Optimization by simulated annealing. **Science**, New York, v.220, p.671-680, 1983.
- KRATICA, J.; STANIMIROVIC, Z.; TOSIC, D.; FILIPOVIC, V. Two genetic algorithms for solving the uncapacitated single allocation p-hub median problem. **European Journal of Operational Research**, v. 182, p. 15-28, 2007.
- LANG, J.C.; SHEN, Z-J.M. Fix-and-optimize heuristics for capacitated lot-sizing with sequence-dependent setups and substitutions. **European Journal of Operational Research**, v. 214, p. 595-605, 2011.
- LEGATO, P.; TRUNFIO, R. A local branching-based algorithm for the quay crane scheduling problem under unidirectional schedules. **4OR: A Quarterly Journal of Operations Research**, v. 12, p. 123-156, 2014.
- METROPOLIS, N; ROSENBLUTH, A, W; ROSENBLUTH, M, N; TELLER, A, H; TELLER, E. Equation of state calculations by fast computing machines. **Journal of Chemical Physics**, v. 21, p. 1087-1092, 1953.
- MLADENOVIC, N. E HANSEN, P. Variable neighborhood search. **Computers and Operations Research**, v. 24, n. 11, p. 1097-1100, 1997.
- MORAES, L.C.C.; SANTOS, M.O. Heurísticas relax-and-fix para o problema de dimensionamento de lotes com janelas de tempo de produção. In: CONGRESSO LATINO-IBEROAMERICANO DE INVESTIGACIÓN OPERATIVA, 16. / SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 44, 2012, Rio de Janeiro, RJ. **Anais...** Rio de Janeiro, 2012.
- OLIVEIRA, A. C. M.; LORENA, L. A. N. Detecting promising areas by evolutionary clustering search. **Lecture Notes in Artificial Intelligence**, v. 3171, p. 385-394, 2004.
- OLIVEIRA, A. C. M.; LORENA, L. A. N. Hybrid evolutionary algorithms and clustering search. In: Grosan, C.; Abraham, A.; Ishibuchi, H. (eds.). **Hybrid**

**Evolutionary systems** - studies in computational intelligence. Springer, v. 75, p. 81-102, 2007.

O'KELLY, M. A quadratic integer program for the location of interacting hub facilities. **European Journal of Operational Research**, v. 32, p. 393-404, 1987.

O'KELLY, M. Hub facility location with fixed costs. **Papers in Regional Science**, v. 71, n. 3, p. 293–306, July 1992. DOI: 10.1007/BF01434269.

PADBERG, M.; RINALDI, G. Optimization of a 532-city symmetric traveling salesman problem by branch and cut. **Operations Research Letters**, v. 6, n. 1, p. 1-7, 1987.

PADBERG, M.; RINALDI, G. A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems. **SIAM review**, v. 33, n. 1, p. 60-100, 1991.

PEDROSO, J.P.; KUBO, M. Hybrid tabu search for lot sizing problems. In: BLESA, M.; BLUM, C.; ROLI, A.; SAMPELS, M. (eds.). **Lecture Notes in Computer Science**, Springer, Berlin/Heidelberg, v. 3636, p. 66-77, 2005.

SAHLING, F.; BUSCHKÜHL, L.; TEMPELMEIER, H.; HELBER, S. Solving a multi-level capacitated lot sizing problem with multi-period setup carry-over via a fix-and-optimize heuristic. **Computers & Operational Research**, v. 36, p. 2546-2553, 2009.

SASAKI, M.; FUKUSHIMA, M. On the hub-and-spoke model with arc capacity constraints. **Journal of the Operational Research Society of Japan**, v. 46, n. 4, p. 409-428, 2002.

SILVA, M.R. **Uma contribuição ao problema de localização de terminais de consolidação no transporte de carga parcelada**. Dissertação (Mestrado em

Engenharia de transportes) - Escola Politécnica da Universidade de São Paulo, 2004.

TOLEDO, C.F.M.; OLIVEIRA, R.R.R.; FRANÇA, P.M. A hybrid heuristic approach to solve the multilevel capacitated lot sizing problem. In: IEEE CONGRESS ON EVOLUTIONARY COMPUTATION (CEC 2011), 2011, New Orleans, LA, USA. **Proceedings...** New Orleans, LA: IEEE, 2011. p. 1194-1201. ISBN 978-1-4244-7834-7.

TOPCUOGLU, H.; CORUT, F.; ERMIS, M.; YLMAZ, G. Solving the uncapacitated hub location problem using genetic algorithms. **Computers and Operations Research**, v. 32, p. 967-984, 2005.