



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

sid.inpe.br/mtc-m21b/2014/04.28.19.18-TDI

## **META-HEURÍSTICAS HÍBRIDAS PARA O PROBLEMA PROBABILÍSTICO DE LOCALIZAÇÃO DE MÁXIMA COBERTURA**

Ligia Corrêa de Souza

Dissertação de Mestrado do Curso de Pós-Graduação em Computação Aplicada, orientada pelos Drs. Luiz Antonio Nogueira Lorena, e Marcos Antonio Pereira, aprovada em 10 de abril de 2014.

URL do documento original:

<<http://urlib.net/8JMKD3MGP5W34M/3G7U6HB>>

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. Antonio Fernando Bertachini de Almeida Prado - Coordenação Engenharia e Tecnologia Espacial (ETE)

Dr<sup>a</sup> Inez Staciarini Batista - Coordenação Ciências Espaciais e Atmosféricas (CEA)

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

Dr. Germano de Souza Kienbaum - Centro de Tecnologias Especiais (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:**

Marciana Leite Ribeiro - 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/2014/04.28.19.18-TDI

## **META-HEURÍSTICAS HÍBRIDAS PARA O PROBLEMA PROBABILÍSTICO DE LOCALIZAÇÃO DE MÁXIMA COBERTURA**

Ligia Corrêa de Souza

Dissertação de Mestrado do Curso de Pós-Graduação em Computação Aplicada, orientada pelos Drs. Luiz Antonio Nogueira Lorena, e Marcos Antonio Pereira, aprovada em 10 de abril de 2014.

URL do documento original:

<<http://urlib.net/8JMKD3MGP5W34M/3G7U6HB>>

INPE  
São José dos Campos  
2014

Dados Internacionais de Catalogação na Publicação (CIP)

---

Souza, Ligia Corrêa de.

So89m Meta-heurísticas híbridas para o problema probabilístico de localização de máxima cobertura / Ligia Corrêa de Souza. – São José dos Campos : INPE, 2014.

xxiv + 53 p. ; (sid.inpe.br/mtc-m21b/2014/04.28.19.18-TDI)

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

Orientadores : Drs. Luiz Antonio Nogueira Lorena, e Marcos Antonio Pereira.

1. Localização de facilidades. 2. Problema probabilístico de localização-alocação de máxima cobertura. 3. Meta-heurística. 4. Adaptive large neighborhood search. I.Título.

CDU 519.142

---



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 **Mestre** em

**Computação Aplicada**

Dr. Solon Venâncio de Carvalho



Presidente / INPE / SJC Campos - SP

Dr. Luiz Antonio Nogueira Lorena



Orientador(a) / INPE / São José dos Campos - SP

Dr. Marcos Antonio Pereira



Orientador(a) / UNESP / Guaratinguetá - SP

Dr. Antônio Augusto Chaves



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

Dr. Luiz Leduino de Salles Neto



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

*Este trabalho foi aprovado por:*

( ) maioria simples

unanimidade

Aluno (a): **Ligia Corrêa de Souza**

**São José dos Campos, 10 de Abril de 2014**



*“Os bons e os maus resultados dos nossos ditos e obras vão-se distribuindo, supõe-se que de uma maneira bastante uniforme e equilibrada, por todos os dias do futuro, incluindo aqueles, infindáveis, em que já cá não estaremos para poder comprová-lo, para congratularmo-nos ou para pedir perdão, aliás, há quem diga que é isto a imortalidade de que tanto se fala”.*

*José Saramago*





*A minha família.*



## **AGRADECIMENTOS**

Agradeço a minha família, em especial, a meus pais Claudete e Zacarias, pelo apoio e ensinamentos; minha tia Marinéia, por me acolher, cuidar e ser meu ombro amigo durante os últimos anos do mestrado, além de ser um exemplo de como podemos realmente mudar a vida das pessoas com a educação; a minha doce irmã-amiga Carol, que sempre esteve ao meu lado, mesmo que distante, e sempre me apoiou em todas as minhas decisões.

A meus orientadores, Luiz Antonio Nogueira Lorena, que muito contribuiu durante esse longo aprendizado, e Marcos Antonio Pereira, que com sua paciência inimaginável de sempre, acreditou em mim e me ajudou em todas as dificuldades que surgiram.

A minha amiga Camila, que durante nossos telefonemas à distância, aturou meus desabafos, torceu por mim e me incentivou; e aos amigos que fiz no INPE, em especial, Marina, André, Sabrina, Pettras, Diego, Daniel, Rudinei e Marlon, amigos que me proporcionaram momentos únicos e tornaram o caminho mais leve.

A meu amigo e companheiro Sóstenes, pelas conversas francas e apoio sólido, que me mantiveram inteira nos momentos mais difíceis.

À CAPES pelo auxílio financeiro.

A meus alunos da UNITAU do ano de 2013 que me proporcionaram a experiência enriquecedora de ser professora e um vasto aprendizado, quando achei que minha missão seria apenas ensinar.

A todos os meus professores que me fizeram crer, indubitavelmente, na educação.



## RESUMO

Problemas de localização de facilidades tratam de decisões relativas ao atendimento da demanda por produtos ou serviços para um indivíduo ou uma população, considerando aspectos logísticos. O Problema Probabilístico de Localização Alocação de Máxima Cobertura é um problema de localização de facilidades, que visa localizar  $p$  centros, a fim de maximizar o atendimento à demanda para uma dada distância de cobertura, considerando aspectos de qualidade de serviço. Nesta dissertação é estudada a utilização das meta-heurísticas Large Neighborhood Search e Adaptive Large Neighborhood Search para resolver este problema. Os resultados obtidos com a utilização deste método são comparados com os resultados obtidos pelo software CPLEX e com os resultados da literatura, estabelecendo as medidas de desempenho, tais como tempos de execução e qualidade da solução, da abordagem utilizada.

**PALAVRAS-CHAVE:** localização de facilidades, problema probabilístico de localização-alocação de máxima cobertura, meta-heurística, Adaptive Large Neighborhood Search.



# HYBRID METHODS FOR THE PROBABILISTIC MAXIMAL COVERING LOCATION ALLOCATION PROBLEM

## ABSTRACT

Facility location problems deal with decisions concerning the satisfaction of the demand for products or services of an individual or a population, considering logistical aspects. The Probabilistic Maximal Covering Location Allocation Problem is a facility location problem that aims to locate  $p$  centers in order to maximize the attended demand for a given covering distance, considering service quality aspects. The present dissertation investigates the use of the metaheuristics Large Neighborhood Search and Adaptive Large Neighborhood Search to solve this problem. The results obtained with the use of these methods are compared with results obtained by CPLEX software and the literature results, establishing performance measures, such as computing times and solution quality, of this approach.

**KEYWORDS:** facility location, probabilistic maximal covering location allocation problem, metaheuristic, Adaptive Large Neighborhood Search.





## LISTA DE FIGURAS

Figura 2.1 - Ilustração de um problema de localização de máxima cobertura. ...	8
Figura 2.2 - Diagrama de uma fila com um servidor e fonte infinita. ....	12
Figura 2.3 - Diagrama de transição de filas de Poisson. ....	13
Figura 4.1- Representação de um problema de maximização .....	21
Figura 5.1 - Representação de uma solução.....	29
Figura 5.2 - Solução inicial. ....	31
Figura 5.3 - Solução destruída. ....	31
Figura 5.4 - Solução reparada.....	32
Figura 5.5 - Soluções relativas ao CPLEX da rede de 30 pontos.....	38
Figura 5.6 - Tempo computacional das soluções da rede de 30 pontos. ....	38
Figura 5.7 - Soluções relativas ao CPLEX da rede de 324 pontos.....	41
Figura 5.8 - Tempo computacional das soluções da rede de 324 pontos. ....	42
Figura 5.9 - Soluções relativas ao CPLEX da rede de 818 pontos.....	44
Figura 5.10 - Tempo computacional das soluções da rede de 818 pontos. ....	45
Figura 5.11 - Rede de 1177 pontos.....	45
Figura 5.12 - Soluções relativas ao CPLEX da rede de 1177 pontos.....	47
Figura 5.13 - Tempo computacional das soluções da rede de 1177 pontos. ...	48



## LISTA DE TABELAS

Tabela 5.1 - Resultados para a rede de 30 pontos. ....	37
Tabela 5.2 - Resultados para a rede de 324 pontos. ....	40
Tabela 5.3 - Resultados para a rede de 818 pontos. ....	43
Tabela 5.4 - Resultados para a rede de 1177 pontos. ....	46



## LISTA DE SÍMBOLOS

$S$	Distância de serviço ou raio de cobertura
$v$	Valor da função objetivo
$I$	Conjunto dos clientes da rede
$N_i$	Conjunto das facilidades candidatas que estão dentro da distância de serviço do cliente $i$
$a_i$	Demanda do cliente $i$
$x_{ij}$	Variável de alocação
$y_j$	Variável de localização
$f_i$	Taxa de chegada dos clientes a uma facilidade
$\mu$	Taxa média de atendimento
$b$	Número máximo de clientes na fila
$\tau$	Tempo máximo de espera na fila
$\varphi$	Probabilidade mínima de encontrar no máximo $b$ clientes na fila ou esperar $\tau$ minutos
$p$	Número de facilidades
$\omega_j$	Taxa de chegada atribuída a facilidade $j$
$k$	Número de estados no sistema de filas
$\rho_k$	Probabilidade de ter $k$ clientes no sistema de filas
$x$	Solução corrente
$N(x)$	Vizinhança da solução $x$
$x^b$	Melhor solução
$x^t$	Solução temporária
$c(x)$	Valor da função objetivo da solução $x$
$\rho^-$	Conjunto dos pesos das funções de destruição
$\rho^+$	Conjunto dos pesos das funções de reparo
$d$	Função de destruição
$r$	Função de reparo
$\Omega^-$	Conjunto dos métodos de destruição
$\Omega^+$	Conjunto dos métodos de reparo

$P_j^-$	Probabilidade de escolher o método de destruição $j$
$\psi$	Pontuação máxima para os métodos de destruição e reparo
$\delta_i$	Parâmetro $i$ para os pesos
$\gamma$	Função distribuição de probabilidade
$n$	Sequência de iterações de tamanho $n$
$T_0$	Temperatura inicial
$\alpha$	Fator de resfriamento
$\beta$	Parâmetro de decaimento

## LISTA DE ABREVIATURAS E SIGLAS

ALNS	<i>Adaptative Large Neighborhood Search</i>
CS	<i>Clustering Search</i>
FIFO	<i>First in – First out</i>
GC	Geração de Colunas
LNS	<i>Large Neighborhood Search</i>
MCLP	<i>Maximal Covering Location Problem</i>
MH	Meta-heurística
PCP	<i>p-Center Problem</i>
QM-CLAM	<i>Queueing Maximal Covering Location-Allocation Model</i>
SA	<i>Simulated Annealing</i>
SCLP	<i>Set Covering Location Problem</i>





## SUMÁRIO

	<u>Pág.</u>
<b>1 INTRODUÇÃO</b> .....	1
<b>2 PROBLEMA PROBABILÍSTICO DE LOCALIZAÇÃO-ALOCAÇÃO DE MÁXIMA COBERTURA</b> .....	5
2.1 Descrição do problema.....	8
2.2 Formulação Matemática.....	10
2.2.1 Condições de estocacidade.....	12
<b>3 REVISÃO BIBLIOGRÁFICA</b> .....	17
3.1 Modelos congestionados.....	17
3.2 Abordagens anteriores para o QM-CLAM.....	18
3.2.1 Heurística de Marianov e Serra (1998).....	18
3.2.2 Abordagem de Corrêa e Lorena (2006).....	18
3.2.3 Abordagem de Corrêa et al. (2007).....	19
3.2.4 Abordagem de Corrêa et al. (2009).....	20
<b>4 META-HEURÍSTICAS</b> .....	21
4.1 LNS.....	22
4.1.1 Funções de destruição e reparo do LNS.....	23
4.2 ALNS.....	24
4.2.1 Funções de destruição e reparo do ALNS.....	26
<b>5 APLICAÇÃO DO LNS E DO ALNS AO QM-CLAM</b> .....	29
5.1 Implementação.....	29
5.1.1 Solução de localização.....	29
5.1.2 Solução inicial.....	29
5.1.3 Solução de alocação.....	30
5.1.4 Funções de destruição e reparo do LNS.....	30
5.1.5 Funções de destruição e reparo do ALNS.....	32
5.1.5.1 Funções de destruição.....	32
5.1.5.2 Funções de reparação.....	33
5.1.6 Aceitação da solução.....	33
5.2 Resultados.....	34
<b>6 CONCLUSÃO</b> .....	49
6.1 Trabalhos futuros.....	49
<b>REFERÊNCIAS BIBLIOGRÁFICAS</b> .....	51



## 1 INTRODUÇÃO

Localização de facilidades é uma das atividades da Logística que envolve decisões sobre o número de centros, de fábricas, depósitos, escolas, postos de saúde, delegacias etc e locais de instalação dos mesmos, visando o atendimento de certa demanda de produtos ou serviços (DASKIN, 1995).

Dentre os critérios utilizados para decidir a melhor solução de localização estão: proximidade aos clientes, custos de instalação e de distribuição, disponibilidade e qualidade da infraestrutura, mão de obra e fornecedores. Tais fatores afetam diretamente a localização, tornando-se fatores restritivos que devem ser considerados no planejamento.

Os critérios para a seleção apropriada dos locais para a instalação das facilidades envolvem questões mais elaboradas do que a minimização de custos ou distâncias. Outros aspectos, inclusive questões mercadológicas, como incentivos fiscais, leis ambientais, estabilidade política e vantagens competitivas, podem interferir no processo de decisão. Logo, para cada problema de localização de facilidades, devem ser considerados os aspectos que melhor representem as diversas dimensões do problema.

Em algumas aplicações, o número de clientes e a demanda associada a eles alocada à alguma facilidade pode impactar na qualidade de serviço. Dependendo da natureza do serviço, pode haver clientes à espera para ser atendidos. O Problema Probabilístico de Localização-Alocação de Máxima Cobertura (do inglês *Queueing Maximal Covering Location-Allocation Model – QM-CLAM*) considera a qualidade mínima de atendimento dos clientes localizados dentro do raio de cobertura de uma dada facilidade como um critério relevante. A aplicação deste problema abrange os âmbitos público e privado, como localização de postos de saúde (CORRÊA et al., 2007) e sistemas congestionados (MARIANOV; SERRA, 1998).

Para resolver o QM-CLAM, técnicas baseadas em Relaxação Lagrangeana combinadas a métodos de Otimização por Subgradientes e o Algoritmo Genético Construtivo foram empregadas (CORRÊA; LORENA, 2006). Ribeiro e

Lorena (2007) propuseram a resolução deste problema por meio da Relaxação Lagrangeana com *clusters*, conhecida como *LagClus*. A *LagClus* utiliza uma representação do problema por meio de um grafo de conflitos. Em seguida, algumas arestas de conexão desse grafo são relaxadas no sentido lagrangeano, e os subgrafos resultantes correspondem a subproblemas que podem então ser resolvidos independentemente. Métodos como o *Clustering Search* (CS) (CORRÊA et al., 2007) e técnicas como Geração de Colunas (CORRÊA et al., 2009) encontraram soluções boas e ótimas, respectivamente, para instâncias propostas na literatura.

Mesmo com o emprego de *softwares* comerciais avançados, os tempos computacionais para a obtenção de soluções ótimas aumentam em proporção direta com o tamanho das instâncias, o que justifica a pesquisa em métodos baseados em heurísticas. Algumas destas heurísticas exploram o espaço de soluções do problema, na vizinhança de alguma solução conhecida, em busca de uma solução de melhor qualidade. Algumas implementações permitem, temporariamente, a adoção de soluções de pior qualidade como um mecanismo para expandir o espaço de busca e escapar de ótimos locais.

Este trabalho explora a formulação matemática do QM-CLAM, considerando separadamente as decisões de localização e alocação, para permitir o emprego de uma heurística híbrida que combina um método exato com uma heurística de busca em vizinhança. Foram consideradas as heurísticas *Large Neighborhood Search* (LNS) e *Adaptative Large Neighborhood Search* (ALNS). A heurística LNS foi proposta por Shaw (1998) para problemas de roteamento de veículos tendo sido empregada na resolução destes problemas desde então. A heurística ALNS é uma adaptação da LNS, desenvolvida por Ropke e Pisinger (2006b) e foi aplicada com êxito ao problema de roteamento de veículos (ROPKE; PISINGER, 2006a; HEMMELMAYR, 2012).

Neste trabalho, as meta-heurísticas LNS e ALNS serão utilizadas para determinar a solução de localização, a partir da qual define-se um modelo matemático para calcular a solução de alocação por um método exato (CPLEX) de acordo com a configuração de localização proposta pela heurística. O algoritmo proposto coordena o emprego desses dois métodos num processo

iterativo até que algum critério de parada seja satisfeito (tempo de execução, qualidade da solução etc).

O presente trabalho está organizado da seguinte forma. No capítulo 2 apresenta-se um breve estudo sobre o modelo QM-CLAM. No capítulo 3 apresentam-se abordagens anteriores sobre o problema. No capítulo 4 faz-se uma breve explicação sobre os métodos LNS e ALNS utilizados para resolver o problema proposto. No capítulo 5, comparam-se os resultados dos estudos computacionais realizados com os apresentados na literatura, visando avaliar a qualidade das soluções e os tempos computacionais obtidos com o emprego da técnica proposta. E por fim, no capítulo 6 apresentam-se as conclusões e propostas de trabalhos futuros.



## **2 PROBLEMA PROBABILÍSTICO DE LOCALIZAÇÃO-ALOCAÇÃO DE MÁXIMA COBERTURA**

A localização de facilidades é uma questão essencial no planejamento estratégico de empresas privadas e públicas e está relacionada à variação da população em uma região, capital de investimento e estimativa de clientes que podem ser servidos (PRADO, 2007).

A tomada de decisão de localização visa a escolha de locais para a instalação de uma ou mais facilidades, com o propósito de fornecer produto e/ou serviços a um conjunto de centros consumidores. Desse modo, o problema de localização de facilidades envolve decisões que afetam empreendimentos novos ou existentes.

Pode-se definir duas classes de problemas de localização. O modelo clássico para representar a primeira classe é o do problema de p-medianas, que visa determinar a configuração de custo mínimo de instalação de facilidades e de atendimento da demanda de cada cliente em uma rede conectada por um número finito de caminhos. Podem ocorrer casos em que existam mais de uma configuração de custo mínimo (PEREIRA, 2005). Problemas de minimização de custos caracterizam problemas do setor privado onde se busca a redução do custo total de transporte e de infraestrutura. Neste caso, as facilidades são, em geral, fábricas, depósitos, lojas de franquias, postos de combustível etc (LORENA et al., 2001). Podem ser considerados custos fixos de instalação de facilidades.

A segunda classe de problemas enfoca a distância máxima entre qualquer cliente e a facilidade designada para atendimento, denominada distância de cobertura ou de serviço ou de atendimento. Tais problemas são denominados problemas de cobertura, cujo objetivo é determinar o número mínimo de centros necessários ao atendimento de todos os pontos de demanda, para uma dada distância de cobertura, sem considerar o valor da demanda de cada ponto (PEREIRA, 2005).

Como no setor público visa-se basicamente a maximização de um benefício para a sociedade, já que o principal propósito deve ser satisfazer a maior quantidade de pessoas possível com o que se tem disponível de recurso econômico, os problemas de cobertura se ajustam bem a estas aplicações. A localização de escolas, postos de saúde, corpo de bombeiros, ambulâncias, viaturas de polícia, pontos de ônibus, podem ser citados como exemplos de aplicações em setores públicos (LORENA et al., 2001).

Esta classe de problemas considera que toda a demanda deva ser atendida, o que nem sempre é possível. Uma alternativa a este problema considera que o número de facilidades a serem instaladas não é suficiente para o atendimento de toda a demanda existente, para uma dada distância de cobertura. O Problema de Localização de Máxima Cobertura (do inglês *Maximal Covering Location Problem* – MCLP) considera uma rede formada por arcos, um conjunto de pontos de demanda (clientes) e um conjunto de locais candidatos para a instalação de centros de atendimento (facilidades). A cada facilidade está associada uma distância de cobertura  $S$  de forma que apenas os clientes situados a uma distância menor que  $S$  de alguma facilidade serão atendidos. Como o número de facilidades a serem instalados pode não ser suficiente para atender todos os clientes, o problema busca determinar os locais de instalação das facilidades que atendam a maior parte da demanda existente. Modelos de cobertura e de máxima cobertura são frequentemente utilizados por órgãos públicos para a localização de serviços, sobretudo os emergenciais.

O MCLP foi proposto por Church e ReVelle (1974) como uma alternativa para os problemas de localização de cobertura (do inglês *Set Covering Location Problem* – SCLP), que busca determinar o número mínimo de facilidades que atendam toda a demanda considerando que a distância entre um ponto de demanda e a facilidade mais próxima não ultrapasse a distância crítica; e o problema dos  $p$ -centros (do inglês *p-Center Problem* – PCP) cujo objetivo é localizar  $p$  facilidades de maneira que a distância máxima entre qualquer ponto de demanda até a facilidade mais próxima seja mínima.

Nota-se que tanto no SCLP quanto no PCP exige-se que todos os pontos de demanda sejam cobertos, mas nem sempre existem recursos suficientes



disponíveis, o que pode tornar inviável a busca por uma solução. Por sua vez, o MCLP relaxa a restrição de atendimento de todos os clientes, mas procura maximizar o atendimento da demanda dos clientes, dentro de uma distância ou tempo de serviço, com um número fixo de facilidades, respeitando as restrições de recursos (ARAKAKI, 2002; PEREIRA, 2005).

No modelo proposto por Church e ReVelle (1974) para o MCLP, as decisões de atendimento são representadas pelas variáveis  $y_i$ , com  $y_i = 1$  se o cliente é coberto por alguma facilidade e  $y_i = 0$ , caso contrário. A decisão de localização é representada pelas variáveis binárias  $x_j$ , com  $x_j = 1$ , se a facilidade é instalada no ponto  $j$ , e  $x_j = 0$ , caso contrário. O parâmetro  $a_i$  define a população total ou a demanda do ponto  $i$ ;  $p$  é o número de facilidades a serem localizadas;  $I$  é o conjunto dos pontos de demanda;  $J$  é o conjunto dos locais candidatos para instalação de facilidades e  $N_i$  é o subconjunto de locais candidatos  $j$  que podem atender o cliente  $i$ , de acordo com uma dada distância de atendimento  $S$ .

Com os parâmetros assim definidos, o modelo segue como:

$$v(MCLP) = \text{Max} \left\{ \sum_{i \in I} a_i y_i \right\} \quad 2.1$$

*Sujeito a:*

$$\sum_{j \in N_i} x_j \geq y_i, \quad \forall i \in I \quad 2.2$$

$$\sum_{j \in J} x_j = p \quad 2.3$$

$$y_i \text{ e } x_j \in \{0,1\}, \quad \forall i \in I \text{ e } \forall j \in J \quad 2.4$$

A função objetivo (2.1) maximiza a demanda alocada aos centros. As restrições (2.2) definem que somente é possível alocar um ponto de demanda  $i$  a um centro  $j$  se houver um centro instalado em  $j$ . A restrição de cardinalidade (2.3)

define a quantidade de centros a serem abertos. E as restrições (2.4) definem a natureza binária das variáveis de decisão.

A Figura 2.1 ilustra uma solução para um problema de localização de máxima cobertura com três facilidades a serem instaladas, para um dado raio de cobertura.

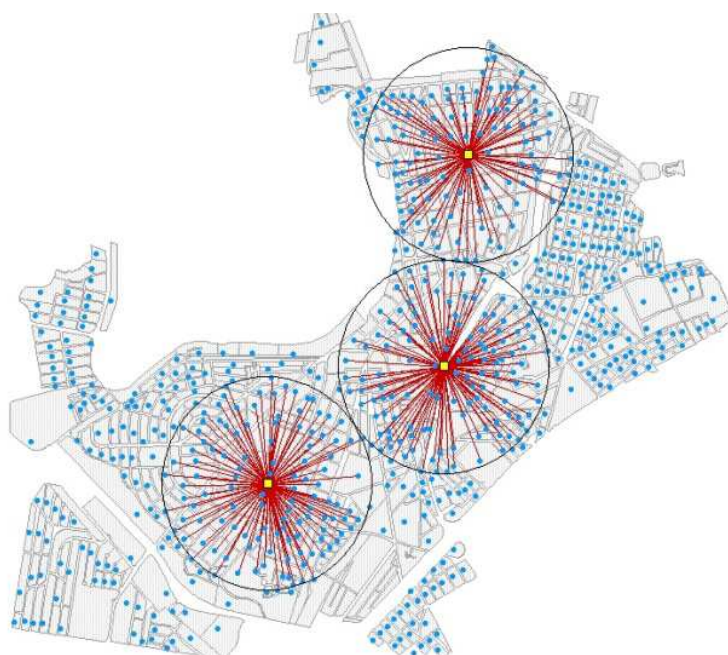


Figura 2.1 - Ilustração da solução de um problema de localização de máxima cobertura.

Fonte: Pereira e Lorena (2001).

## 2.1. Descrição do QM-CLAM

Como mencionado anteriormente, o MCLP considera que nem toda a demanda seja atendida, em vista da quantidade de facilidades e da distância de atendimento estabelecidas. No entanto, essa abordagem não considera alguns detalhes do atendimento (como a espera causada por filas) que, em alguns casos, podem incorrer em custos adicionais indiretos. Usualmente, associa-se a formação de filas a um excesso de demanda de um serviço em relação a capacidade de atendimento (MOREIRA, 2007).

Outros problemas podem ser modelados como extensões dessa formulação, de modo a representar com mais fidelidade a realidade, pois para os sistemas

que apresentam comportamento de chegada aleatória de clientes, a qualidade do serviço não está relacionada apenas com a cobertura dos clientes.

Sendo assim, as facilidades devem ser localizadas de tal maneira que os clientes cheguem dentro de um tempo aceitável e também que, uma vez no centro de atendimento, o tempo de espera em fila não seja maior que um limite máximo e/ou que o comprimento da mesma não seja maior que um valor máximo, garantindo assim uma qualidade mínima no atendimento (MARIANOV; SERRA, 1998).

O congestionamento no atendimento pode ocorrer não só porque a capacidade do centro seja insuficiente, mas também devido à variabilidade no intervalo de chegadas de clientes e do tempo de atendimento nos centros. Os modelos tradicionais que tratam desse problema adicionam uma restrição de capacidade que força a demanda por serviço, normalmente constante e menor do que a máxima capacidade do centro, considerando que o número de solicitações de serviço é constante no tempo. Porém, tratando o problema de forma determinística, podem-se ter servidores ociosos ou sobrecarregados (MOREIRA, 2007).

Considerando essa aleatoriedade nos processos de chegada e de atendimento, Marianov e Serra (1998) propuseram modelos que definem uma qualidade mínima no serviço tratando a estocasticidade do problema em restrições adicionais de capacidade.

Esses autores definiram então o Problema Probabilístico de Localização-Alocação de Máxima Cobertura (QM-CLAM), fazendo as considerações de estocasticidade da demanda, que busca localizar uma dada quantidade de facilidades com um ou vários servidores, de modo que a população, a uma dada distância de um centro de atendimento, seja servida adequadamente, isto é, que o usuário não fique na fila por um período maior que um dado tempo limite ou que, ao chegar ao centro, não encontre um número de outros clientes acima do previsto, dada uma probabilidade mínima de que isso ocorra. No modelo estudado neste trabalho, considera-se um servidor por centro, ou seja, só existe um posto de atendimento ao cliente em cada facilidade, responsável

por seu atendimento integral.

## 2.2. Formulação Matemática

O modelo MCLP tradicional de Church e ReVelle (1974) não pode ser usado para tratar as restrições de capacidade, pois não comporta as variáveis de localização e alocação. Sem elas não seria possível computar as solicitações de serviços que chegam a um centro, para, conseqüentemente, poder determinar a ocorrência de um congestionamento.

É possível derivar novos modelos do QM-CLAM utilizando os trabalhos de Cornuéjols e Thizy (1982) e Murray e Gerrard (1997). Segundo Pontin et al. (2010), a formulação original de Marianov e Serra (1998) é a mais adequada para representar o problema aqui tratado.

No caso geral, considera-se que o conjunto de locais candidatos para a instalação das facilidades é distinto do conjunto de clientes. Neste caso, têm-se dois conjuntos de pontos: um para representar os locais potenciais para a instalação das facilidades e outro para representar os clientes. Neste trabalho, será assumido que cada ponto que representa um cliente é também um local candidato para a instalação de uma facilidade.

As alocações são representadas pela matriz binária  $x_{ij}$ , sendo  $I$  o conjunto dos pontos de demanda e  $N_i$  o conjunto dos pontos candidatos para instalação das facilidades que estão dentro da distância de atendimento do cliente  $i$ , com  $x_{ij} = 1$ , se o ponto de demanda  $i$  é alocado à facilidade  $j$ , e  $x_{ij} = 0$ , caso contrário. A decisão de localização é representada pelas variáveis binárias  $y_j$ , com  $y_j = 1$  se o local  $j$  é selecionado para a instalação de um centro e  $y_j = 0$ , caso contrário. O parâmetro  $a_i$  define a demanda total do ponto ou cliente  $i$ ;  $b$  é o número máximo de clientes na fila com probabilidade de, no mínimo,  $\varphi$ ;  $\tau$  é o tempo máximo (em minutos) de espera na fila com probabilidade de, no mínimo,  $\varphi$ ;  $f_i$  é a taxa de chegada do cliente  $i$ , conforme um processo de Poisson;  $\mu$  é a taxa média de atendimento sendo que o tempo médio de atendimento segue uma distribuição exponencial;  $p$  é o número de facilidades a serem localizadas.

Com os parâmetros assim definidos, o modelo segue como:

$$v(QM - CLAM) = \text{Max} \left\{ \sum_{i \in I} \sum_{j \in N_i} a_i x_{ij} \right\} \quad 2.5$$

Sujeito a:

$$\sum_{j \in N_i} x_{ij} \leq 1, \quad \forall i \in I \quad 2.6$$

$$\sum_j y_j = p \quad 2.7$$

$$x_{ij} \leq y_j, \quad \forall i \in I \text{ e } j \in N_i \quad 2.8$$

$$\sum_{i \in I} f_i x_{ij} \leq \mu^{b+2} \sqrt{1 - \varphi}, \quad \forall j \quad 2.9$$

ou

$$\sum_{i \in I} f_i x_{ij} \leq \mu + \frac{1}{\tau} \ln(1 - \varphi), \quad \forall j \quad 2.10$$

$$y_j \text{ e } x_{ij} \in \{0,1\}, \quad \forall i \in I \text{ e } \forall j \quad 2.11$$

A função objetivo (2.5) maximiza a demanda alocada aos centros. As restrições (2.6) impõem que cada ponto de demanda deve ser alocado a, no máximo, uma facilidade. A restrição de cardinalidade (2.7) define a quantidade de centros a serem abertos. As restrições (2.8) definem que somente é possível alocar um ponto de demanda  $i$  a um centro instalado em  $j$ . As restrições (2.9) e (2.10) são as restrições de congestionamento. Dependendo do problema, apenas uma delas será utilizada. As restrições (2.9) forçam para que haja no máximo  $b$  clientes na fila com, no mínimo, a probabilidade  $\varphi$ . As restrições (2.10) determinam que o tempo gasto no centro  $j$  seja, no máximo  $\tau$  minutos, com, no mínimo, a probabilidade  $\varphi$ . E, por fim, as restrições (2.11) definem a natureza binária das variáveis de decisão.

### 2.2.1. Condições de estocasticidade

Ao descrever as restrições (2.9) e (2.10), Marianov e Serra (1998) consideraram o modelo de fila M/M/1/∞/FIFO, mostrada na Figura 2.2, o que significa que os intervalos entre chegadas são exponencialmente distribuídos, os tempos de atendimento são independentes e também exponencialmente distribuídos, com apenas um servidor, com capacidade de clientes infinita e a disciplina de fila é do tipo “o primeiro a chegar é o primeiro a ser atendido” (*first in - first out – FIFO*).

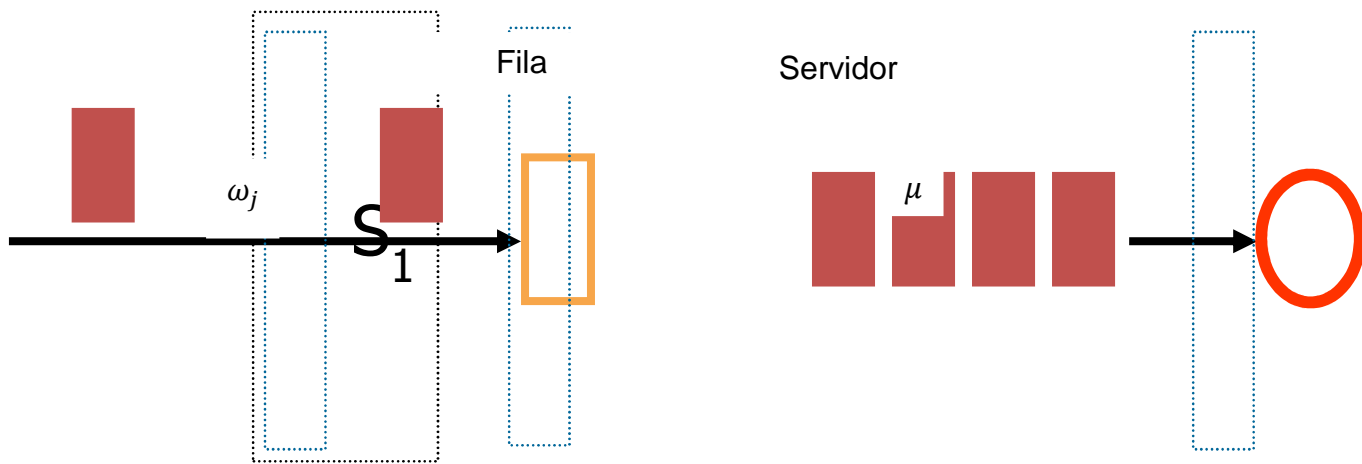


Figura 2.2 - Diagrama de uma fila com um servidor e fonte infinita.

Para as restrições (2.9) e (2.10), considera-se que as solicitações de serviços de cada nó de demanda  $i$  acontecem de acordo com um processo de Poisson com taxa  $f_i$ . Usuários podem vir de vários pontos de demanda, por isso, o processo de chegadas a uma facilidade  $j$  uma superposição de processos de Poisson com taxa total:

$$\omega_j = \sum_{i \in I} f_i x_{ij} \quad 2.12$$

Isso significa que se a variável  $x_{ij}$  for 1, o cliente  $i$  será alocado ao centro  $j$ , e a sua taxa correspondente será incluída no cálculo de  $\omega_j$ . Assume-se também que o tempo de serviço tem uma distribuição exponencial, com taxa  $\mu$ , e, então, média  $\frac{1}{\mu}$ .

Se o estado  $k$  for definido como sendo  $k$  clientes no sistema (em atendimento ou na fila), o diagrama de transição de estado é o mostrado na Figura 2.3. Neste diagrama, o estado  $k = 0$  significa sistema em espera;  $k = 1$ , significa um cliente sendo atendido; para  $k = 2$ , tem-se um cliente sendo atendido e um na fila, e assim por diante.

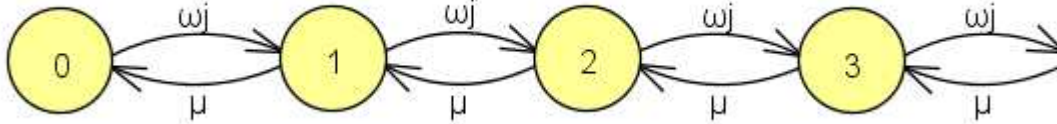


Figura 2.3 - Diagrama de transição de filas de Poisson.

Deseja-se que a probabilidade de um cliente não encontrar mais do que  $b$  pessoas na fila seja, no mínimo,  $\varphi$ . Se  $p_k$  representar a probabilidade de estar no estado  $k$ , então esse requisito pode ser escrito como:

$$p_0 + p_1 + p_2 + p_3 + \dots + p_{b+1} \geq \varphi \quad 2.13$$

As probabilidades  $p_k$  obedecem a uma distribuição geométrica (TAHA, 2008), sendo que  $\rho_j = \frac{\omega_j}{\mu}$  e

$$p_k = (1 - \rho_j)\rho_j^k, \quad \rho_j < 1 \quad 2.14$$

Há a necessidade de que a condição  $\omega_j \leq \mu$  ocorra para manter um equilíbrio no processo de atendimento dos centros. Se  $\omega_j > \mu$ , a série geométrica não convergirá e as probabilidades  $p_k$  do estado de equilíbrio não existirão.

Logo, de 2.13 e 2.14 tem-se:

$$(1 - \rho_j) + (1 - \rho_j)\rho_j + (1 - \rho_j)\rho_j^2 + \dots + (1 - \rho_j)\rho_j^{b+1} \geq \varphi, \quad \text{ou} \quad 2.15$$

$$(1 - \rho_j) \sum_{k=0}^{b+1} \rho_j^k \geq \varphi$$

Da soma dos elementos de uma progressão geométrica finita, tem-se:

$$(1 - \rho_j) \frac{(1 - \rho_j^{b+2})}{(1 - \rho_j)} \geq \varphi \quad 2.16$$

$$1 - \rho_j^{b+2} \geq \varphi \quad 2.17$$

$$\rho_j^{b+2} \leq 1 - \varphi \quad 2.18$$

$$\rho_j \leq \sqrt[b+2]{1 - \varphi} \quad 2.19$$

Como  $\rho_j = \frac{\omega_j}{\mu}$ , pode-se escrever:

$$\omega_j \leq \mu \sqrt[b+2]{1 - \varphi} \quad 2.20$$

Das equações 2.12 e 2.20, tem-se  $\sum_{i \in I} f_i x_{ij} \leq \mu \sqrt[b+2]{1 - \varphi}$ ,  $\forall j$ , que é a equação (2.9).

Se escolhermos a restrição 2.10, temos:

$$P(\text{tempo de espera na facilidade } j \leq \tau) \geq \varphi$$

Para as filas consideradas (M/M/1/ $\infty$ /FIFO) o tempo de espera de cada cliente é exponencialmente distribuído com taxa  $\mu_j - \omega_j$ . Então, podemos usar a função distribuição de probabilidade do tempo de espera numa fila M/M/1,  $\gamma$ , como segue:

$$f_\gamma(\gamma_j) = (\mu_j - \omega_j) e^{-(\mu_j - \omega_j)\gamma_j} \quad 2.21$$

Sendo que a distribuição acumulada corresponde a:

$$P(\gamma_j \leq \tau) = F_\gamma(\tau) = 1 - e^{-(\mu_j - \omega_j)\tau} \quad 2.22$$

A probabilidade na equação 2.22 é definida como maior ou igual a  $\varphi$

$$1 - e^{-(\mu_j - \omega_j)\tau} \geq \varphi, \quad \forall j \quad 2.23$$



$$e^{-(\mu_j - \omega_j)\tau} \leq 1 - \varphi, \quad \forall j \quad 2.24$$

$$-(\mu_j - \omega_j)\tau \leq \ln(1 - \varphi), \quad \forall j \quad 2.25$$

$$\omega_j \leq \mu_j + \frac{1}{\tau} \ln(1 - \varphi), \quad \forall j \quad 2.26$$

Como em 2.12 temos  $\omega_j = \sum_{i \in I} f_i x_{ij}$ , então podemos escrever 2.26 como:

$$\sum_{i \in I} f_i x_{ij} \leq \mu_j + \frac{1}{\tau} \ln(1 - \varphi), \quad \forall j \quad 2.27$$

E como adotamos  $\mu_1 = \mu_2 = \dots = \mu_j = \mu, \quad \forall j$ , então temos a equação 2.10 do modelo apresentado.

Devemos garantir, entretanto, que os parâmetros  $\mu, \tau$  e  $\varphi$  devem respeitar a seguinte condição:

$$\mu > \left| \frac{1}{\tau} \ln(1 - \varphi) \right| \quad 2.28$$

Para a busca do valor ótimo do problema descrito pela formulação (2.5)-(2.11) utilizou-se o *software* CPLEX. Sistemas comerciais de otimização podem resolver algumas instâncias desse problema, mas devido à natureza combinatorial do QM-CLAM, os tempos de processamento necessários para a obtenção de soluções para o problema podem ser elevados mesmo para instâncias de pequeno porte, o que justifica a pesquisa de métodos alternativos para resolver este problema.



### **3 REVISÃO BIBLIOGRÁFICA**

#### **3.1 Modelos congestionados**

Em problemas reais, a distância ou o tempo de deslocamento entre os pontos de demanda e as facilidades constituem fatores importantes para estabelecer o nível de serviço oferecido aos clientes. Na localização de facilidades como hospitais, serviços de atendimento de emergência ou estações do corpo de bombeiros, por exemplo, as chegadas dos clientes aos centros de atendimento normalmente constituem processos aleatórios, ocasionando o surgimento de filas. Da Teoria de Filas, um sistema básico de filas é definido por um processo de chegada, um processo de atendimento, um determinado número de servidores (atendentes), se existe ou não limitação de capacidade e pela disciplina de atendimento (TAHA, 2007).

Vários modelos na literatura são baseados em sistemas de filas. Em geral, porém, estes modelos são orientados para os sistemas de emergência, em que os servidores se deslocam para o local da emergência, em oposição a sistemas em que os servidores são fixos, como centros de cuidados de saúde, como em Berman et al. (1985). Muitos outros trabalhos, considerando servidor único, foram realizados como em Berman et al. (1985) e Berman et al. (1987).

Em geral, um grande esforço computacional é necessário se estes modelos forem usados para a localização dos servidores. Todos os modelos citados anteriormente são não-lineares cujo objetivo é minimizar o tempo de resposta esperado de servidores que viajam para o local de uma emergência. Por isso, estes modelos usam aproximações lineares para modelar o sistema.

Dessa forma, modelos derivados do MCLP são mais utilizados para esse tipo de localização com congestionamento, pois são geralmente lineares e podem ser resolvidos num tempo computacional razoável.

Uma variante do MCLP, normalmente chamado de localização em sistemas congestionados, leva em conta a natureza estocástica da demanda e tempos de resposta. Berman et al. (1985) estão entre os primeiros a incorporar esses sistemas em modelos de localização. Berman e Mandowsky (1986)

apresentam um modelo de localização-alocação para distritalização de uma região com congestionamento e Berman et al. (1987) consideraram o problema de localização de um mecanismo único com sistema de filas  $M / G / k$ .

Marianov e Reville (1994) consideraram o *Queuing Probabilistic Location Set Covering Problem* com o sistema de filas  $M/G/s/s$  e Silva e Serra (2008) apresentaram o MCLP com um sistema de filas  $M/M/1$  e com diferentes níveis de prioridade. O modelo utilizado neste trabalho é o QM-CLAM com sistemas de filas  $M/M/1$  e um servidor, formulado por Marianov e Serra (1998). Além desses, pode-se encontrar uma revisão completa dos modelos com demandas estocásticas e congestionamento no atendimento em Berman e Krass (2004). E mais recentemente Moghadas e Kakhki (2011) estudaram o QM-CLAM com modelos de filas  $M/G/1$ .

## **3.2 Abordagens anteriores para o QM-CLAM**

### **3.2.1 Heurística de Marianov e Serra (1998)**

O QM-CLAM, com o modelo de filas  $M/M/1$  que será estudado neste trabalho, foi proposto por Marianov e Serra (1998), que apresentaram vários modelos considerando o congestionamento do serviço de um problema de máxima cobertura, com um ou mais servidores por centro, e resolveram o problema utilizando uma heurística e o CPLEX. A heurística inicia alocando todos os pontos de demanda a todos os pontos candidatos a facilidade que podem atendê-los de acordo com a distância de cobertura, sendo que os pontos de demandas podem ser alocados a mais de uma facilidade. A partir disso, calcula-se a capacidade de cada facilidade candidata e o valor da função objetivo visando desalocar as facilidades que não melhoram o valor da função objetivo, tendo em vista que todas as restrições do problema devem ser respeitadas.

### **3.2.2 Abordagem de Corrêa e Lorena (2006)**

Para resolver o QM-CLAM, Corrêa e Lorena (2006) utilizaram uma abordagem baseada na aplicação da Relaxação Lagrangeana com o algoritmo de Otimização por Subgradientes e o Algoritmo Genético Construtivo. Esse

algoritmo genético trabalha com uma população de soluções formada por esquemas e estruturas com os operadores tradicionais de um algoritmo genético. As estruturas referem-se a qualquer cadeia contendo ou não símbolo #; os esquemas fazem referência explícita a uma estrutura que contém o símbolo #, isto é, uma população de soluções candidatas parciais ou incompletas, que servirão de base para a construção de uma população com soluções melhores completas, ao longo do processo evolutivo. O símbolo # é chamado de curinga e representa um ponto ainda não definido para o problema. As soluções são representadas por uma cadeia que apresenta três tipos de informações:

- 1 : vértice que representa uma facilidade;
- 2 : vértice que representa um ponto de demanda que é atribuído a uma facilidade; e
- # : vértice curinga, que se tornará uma facilidade ou um ponto de demanda a ser atribuído a uma facilidade durante o processo evolutivo.

O AGC trabalha com uma população de tamanho variável, com a população inicial gerada aleatoriamente. A cada geração do processo evolutivo, novos indivíduos são criados, um para cada cruzamento entre dois indivíduos existentes, privilegiando os mais bem adaptados. Esses novos indivíduos podem sofrer mutação, dentro de uma probabilidade definida *a priori*.

O AGC aplicado ao problema obteve soluções muito boas para a rede de 324 pontos em comparação ao CPLEX e a Relaxação Lagrangeana com o algoritmo de Otimização por Subgradientes utilizados em Corrêa e Lorena (2006).

### **3.2.3 Abordagem de Corrêa et al. (2007)**

Corrêa et al. (2007) usaram a heurística híbrida CS, que consiste na detecção de regiões promissoras de busca, baseada em agrupamentos de soluções. No CS um processo de agrupamento iterativo é executado simultaneamente com uma metaheurística, identificando grupos de soluções que merecem especial interesse. As regiões destes grupos de soluções devem ser exploradas depois de detectadas, por meio de heurísticas de busca local específicas. Corrêa et al.

(2007) resolveram instâncias até 818 vértices e obtiveram resultados competitivos.

#### **3.2.4 Abordagem de Corrêa et al. (2009)**

Corrêa et al. (2009) propuseram a técnica de Geração de Colunas e várias soluções ótimas foram encontradas para instâncias propostas na literatura.

O presente estudo utilizará o trabalho de Corrêa et al. (2009) para a comparação dos resultados, visto que são os melhores encontrados na literatura até o momento.

## 4 META-HEURÍSTICAS

Tradicionalmente utilizam-se métodos baseados em gradiente para resolver problemas de otimização, mas um dos grandes fatores de preocupação é que o algoritmo convirja para um ótimo local. Entende-se por ótimo local o ponto em que qualquer pequeno movimento realizado no mesmo só piora o valor da função objetivo (HILLIER; LIEBERMAN, 2010).

A Figura 4.1 representa um problema de maximização e seus ótimos locais e global, sendo este último o de maior valor para a função objetivo dentre os demais pontos ótimos locais.

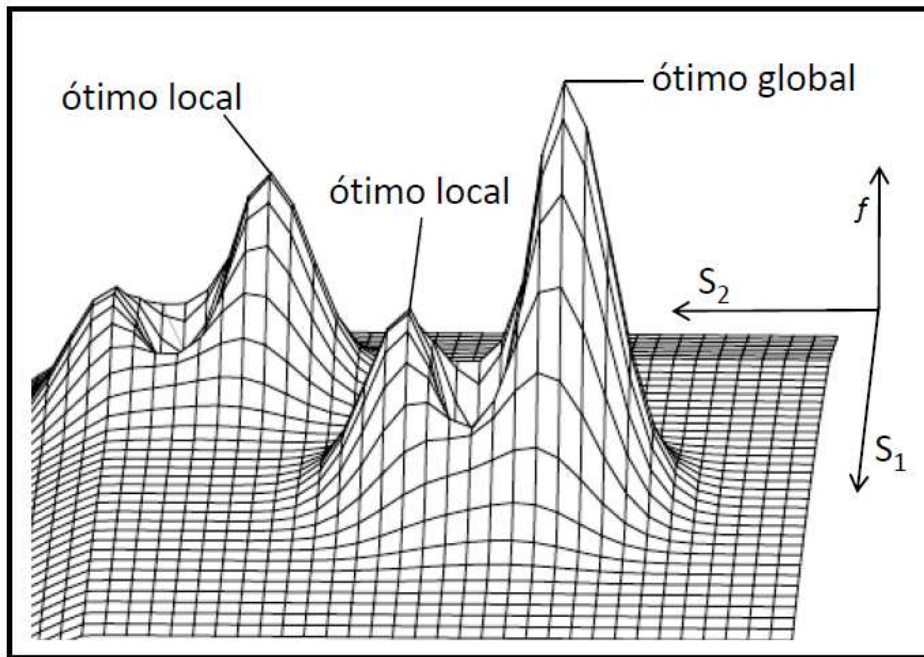


Figura 4.1- Representação de um problema de maximização

Fonte: Chaves (2009)

Um procedimento de melhoria local bem elaborado geralmente é bem sucedido na convergência para um ótimo local. Um método baseado em gradiente é suscetível a essa convergência prematura para um ótimo local. Além disso, muitos problemas têm espaços de busca muito complexos, muitas vezes em função de seus inúmeros ótimos locais, o que dificulta ainda mais o processo de busca. Em problemas combinatoriais de grande porte, o conjunto de

soluções viáveis pode ser muito grande, o que torna inviável a análise individual de todas elas, uma vez que o tempo computacional exigido fica impraticável.

Os algoritmos heurísticos foram desenvolvidos com o propósito de resolver problemas com elevado nível de complexidade, como problemas com as características citadas anteriormente, em tempo computacional razoável, apesar de não conseguir definir se a solução encontrada é ótima ou quão próxima está da otimalidade (GOLDBARG, 2000).

Meta-heurísticas utilizam conhecimento histórico dos resultados adquiridos pelo método para orientar na busca do ótimo global. São procedimentos que orquestram a interação entre métodos de melhoria local e estratégias de níveis mais altos que tem mecanismos para escapar de ótimos locais, assim realizando uma busca consistente no espaço de soluções, destinados a encontrar uma boa solução, eventualmente a ótima (HILLIER; LIEBERMAN, 2010).

#### **4.1 LNS**

A meta-heurística *Large Neighborhood Search* (LNS) foi proposta por Shaw (1998). A ideia principal do LNS é que a vizinhança permite que a meta-heurística navegue no espaço de solução facilmente, mesmo que a instância possua restrições que tornam o problema difícil de ser resolvido.

No LNS, a vizinhança é definida implicitamente por um método de destruição e um método de reparação. O primeiro método destrói parte da solução atual, tornando-a inviável, e contém um elemento aleatório de tal modo que diferentes partes da solução são destruídas em cada chamada do método, enquanto um método de reparação reconstrói a solução para obter uma nova solução viável. A vizinhança  $N(x)$  de uma solução  $x$  é então definida como o conjunto de soluções que podem ser alcançadas pela primeira ao aplicar o método de destruição e, em seguida, o método de reparo.

No Algoritmo 1, mostra-se o pseudo-código do LNS. A variável  $x^b$  é a melhor solução observada durante a busca,  $x$  é a solução atual e  $x^t$  é uma solução



temporária que pode ser descartada ou promovida para ser a solução atual. A função  $d(\cdot)$  é o método de destruição que retorna a solução  $x$  parcialmente destruída enquanto  $r(\cdot)$  é o método de reparo que devolve uma solução viável construída a partir da solução destruída.

---

**Algoritmo 1: LNS**

---

```
1: Entrada: uma solução  $x$  viável
2:  $x^b = x$ ;
3: Repita
4:    $x^t = r(d(x))$ 
5:   Se aceitar  $x^t$  então
6:      $x = x^t$ 
7:   Fim se
8:   Se  $c(x^t) > c(x^b)$  então
9:      $x^b = x^t$ 
10:  Fim se
11: Até que o critério de parada seja atingido
12: Retorne  $x^b$ 
```

---

Na linha 2 uma solução aleatória é gerada e armazenada na variável  $x^b$ . Na linha 4 aplicam-se os métodos de destruição e reparo gerando a solução  $x^t$ . Na linha 5, a nova solução é avaliada, e uma heurística determina se esta solução deve ser a nova solução corrente (linha 6) ou se deve ser rejeitada. A função de aceitação pode ser implementada de diferentes maneiras. A linha 8 verifica se a nova solução é melhor do que a melhor solução conhecida e então atualiza-se a melhor solução na linha 9, se necessário, sendo que  $c(x)$  denota o valor objetivo da solução  $x$ . O processo das linhas (4 – 10) é repetido até que um determinado critério de parada seja alcançado. Na linha 12 a melhor solução encontrada é retornada.

#### 4.1.1 Funções de destruição e reparo do LNS

De acordo com Pisinger e Ropke (2009), a escolha mais importante da aplicação do método de destruir é o grau de destruição: se apenas uma pequena parte da solução é destruída, a heurística pode ter dificuldade para explorar o espaço de busca porque grande parte da vizinhança é perdida. Por outro lado, se uma parte muito grande da solução é destruída, perde-se a essência da vizinhança. E dependendo da maneira que a solução for reparada,

pode ser demorado ou pode haver produção de soluções de baixa qualidade. Shaw (1998) propôs o aumento gradual do grau de destruição, enquanto Ropke e Pisinger (2006b) escolheram o grau de destruição aleatoriamente em cada iteração.

O método de destruição também deve ser escolhido de tal modo que todo o espaço de busca possa ser alcançado, ou pelo menos a parte interessante do espaço de busca, onde o ótimo global deverá ser encontrado, possibilitando, então, a destruição de qualquer parte da solução (PISINGER; ROPKE, 2009).

O método de reparação deve construir uma boa solução, se possível a melhor, a partir da solução parcial. Esse método ideal será mais lento do que uma heurística, mas pode potencialmente levar a soluções de alta qualidade em poucas iterações. No entanto, de um ponto de vista de diversificação, uma operação de reparação que encontra sempre a melhor solução possível pode não ser atraente, pois pode ser difícil sair de ótimo locais, a menos que grande parte da solução seja destruída a cada iteração.

Para diversificar a busca, nos métodos de destruição pode-se selecionar aleatoriamente as partes da solução que devem ser destruídas, e para intensificar pode-se tentar remover as variáveis que tenham um grande custo. Outro modo de destruir é utilizar a informação histórica (PISINGER; ROPKE, 2007), por exemplo contar quantas vezes uma determinada variável para um valor específico leva a uma má solução. Pode-se então tentar remover as variáveis que estão atualmente atribuídas a um valor impróprio, com base nessa informação.

Os métodos de reparação podem ser baseados em algoritmos de aproximação ou algoritmos exatos, sendo que estes últimos podem ser modificados a fim de obter tempos computacionais mais reduzidos.

## **4.2 ALNS**

A meta-heurística *Adaptive Large Neighborhood Search* (ALNS) é uma extensão da meta-heurística LNS, e foi desenvolvida por Ropke e Pisinger (2006b). Esta meta-heurística foi utilizada para resolver uma gama de

problemas relacionados, principalmente, a roteamento de veículos (ROPKE; PISINGER, 2006a; HEMMELMAYR, 2012). Até recentemente, apenas o trabalho de Hemmelmayr (2012) utilizava o ALNS aplicado a um problema de localização.

No ALNS dispõe-se de múltiplos métodos de destruição e reparo, que realizam uma alternância entre uma solução inviável e uma solução viável. Cada método tem um peso, que pode ser ajustado dinamicamente, para controlar sua frequência de utilização durante o processo de solução. Desse modo, tem-se também múltiplas vizinhanças. A escolha dos métodos de destruição ou reparo é baseado em mérito, ou seja, as funções que apresentam melhor desempenho tendem a ser escolhidas com mais frequência.

No Algoritmo 2, mostra-se o pseudo-código do ALNS. A variável  $x^b$  é a melhor solução observada durante a busca,  $x$  é a solução atual e  $x^t$  é uma solução temporária que pode ser descartada ou promovida para ser a solução atual. A função  $d(\cdot)$  é o método de destruição que retorna a solução  $x$  parcialmente destruída enquanto  $r(\cdot)$  é o método de reparo que devolve uma solução viável construída a partir da solução destruída. Os conjuntos dos métodos de destruição e reparo são denotados por  $\Omega^-$  e  $\Omega^+$ , respectivamente. Já  $\rho^-$  e  $\rho^+$  armazenam os pesos dos métodos de destruição e reparo, respectivamente. Inicialmente, todos os métodos tem o mesmo peso, significando que todos tem a mesma chance de serem escolhidos.

---

#### Algoritmo 2: ALNS

---

- 1: Entrada: uma solução  $x$  viável
- 2:  $x^b = x$ ;  $\rho^- = (1, \dots, 1)$ ;  $\rho^+ = (1, \dots, 1)$ ;
- 3: Repita
- 4:     Selecione um método de destruição  $d \in \Omega^-$  e um de reparo  $r \in \Omega^+$  usando  $\rho^-$  e  $\rho^+$
- 5:      $x^t = r(d(x))$
- 6:     Se aceitar  $x^t$  então
- 7:      $x = x^t$
- 8:     Fim se
- 9:     Se  $c(x^t) < c(x^b)$  então
- 10:      $x^b = x^t$
- 11:     Fim se
- 12:     Atualize  $\rho^-$  e  $\rho^+$  após  $k$  iterações
- 13: Até que o critério de parada seja atingido

Na linha 2 uma solução aleatória é gerada e armazenada na variável  $x^b$ . Na linha 4 escolhe-se os métodos de destruição e reparação, para que sejam aplicados na linha 5 gerando a solução  $x^t$ . Na linha 6, a nova solução é avaliada, e uma heurística determina se esta solução deve ser a nova solução corrente (linha 7) ou se deve ser rejeitada. A função de aceitação pode ser implementada de diferentes maneiras. A linha 9 verifica se a nova solução é melhor do que a melhor solução conhecida e então atualiza-se a melhor solução na linha 10, se necessário, sendo que  $c(x)$  denota o valor objetivo da solução  $x$ . Na linha 12 os pesos são atualizados após  $n$  iterações, e o processo das linhas (4 – 12) é repetido até que um determinado critério de parada seja alcançado. Na linha 14 a melhor solução encontrada é retornada.

#### 4.2.1 Funções de destruição e reparo do ALNS

O ALNS, como dito anteriormente, tem múltiplos métodos de destruição e reparo, porém somente um método de destruição e um de reparação pode ser aplicado a cada iteração do método. Há, então, a necessidade de escolher, a cada iteração, o método a ser utilizado.

A seleção dos métodos de destruição e reparo pode ser feita utilizando, por exemplo, a roda da roleta, como sugere Ropke e Pisinger (2006b), calculado do seguinte modo:

$$P_j^- = \frac{\rho_j^-}{\sum_{k=1}^{|\Omega^-|} \rho_k^-} \quad 4.1$$

sendo  $P_j^-$  a probabilidade de se escolher o método de destruição  $j$ . As probabilidades para os métodos de reparo são determinados do mesmo modo.

Outros métodos, como o do torneio, também podem ser utilizados para calcular as probabilidades dos métodos de destruição e reparo.

Os pesos são ajustados dinamicamente, com base no desempenho de cada método. Quando uma iteração do ALNS é concluída, uma pontuação  $\psi$  para os métodos de destruição e reparo é calculada usando a fórmula:

$$\psi = \begin{cases} \delta_1 & \text{se a nova solução é melhor solução já encontrada} \\ \delta_2 & \text{se a nova solução é melhor do que a solução corrente} \\ \delta_3 & \text{se a nova solução é aceita} \\ \delta_4 & \text{se a nova solução é rejeitada} \end{cases} \quad 4.2$$

sendo  $\delta_1, \delta_2, \delta_3$  e  $\delta_4$  parâmetros que, normalmente  $\delta_1 \geq \delta_2 \geq \delta_3 \geq \delta_4 \geq 0$ .

Os pesos são atualizados, após uma sequência fixa  $k$  de iterações, seguindo a fórmula abaixo, sendo que  $a$  e  $b$  são, respectivamente, os índices dos métodos de destruição e reparo utilizados na última iteração e  $\beta \in [0,1]$  é o parâmetro de decaimento que controla como os pesos são sensíveis a alterações no desempenho dos métodos.

$$\rho_a^- = \beta \rho_a^- + (1 - \beta)\psi, \quad \rho_b^+ = \beta \rho_b^+ + (1 - \beta)\psi \quad 4.3$$

O ALNS é propenso a favorecer métodos de reparo complexos, que mais frequentemente atingem soluções de alta qualidade, em comparação com métodos mais simples. Isso é bom se os métodos de reparo simples e complexos têm um tempo computacional muito parecido, mas pode não ser o caso. Se alguns métodos são significativamente mais lentos do que outros, pode-se normalizar a pontuação  $\psi$  de um método com uma medida do tempo de consumo de ação da heurística correspondente (ROPKE; PISINGER, 2006b).

Mesmo considerando vizinhanças de grande porte e a diversidade das mesmas, o algoritmo ALNS explora grande parte do espaço de soluções de forma estruturada. O algoritmo resultante torna-se muito robusto, sendo capaz de se adaptar às várias características individuais, e raramente se detém em ótimos locais. A calibração do algoritmo ALNS é bem limitada devido ao ajuste automático da influência de cada método utilizado.



## 5 APLICAÇÃO DO LNS E DO ALNS AO QM-CLAM

Até recentemente, as metaheurísticas LNS e ALNS têm sido mais amplamente aplicadas nas áreas de roteamento de veículos. Neste trabalho é proposto resolver o QM-CLAM utilizando tais meta-heurísticas por se tratar de uma abordagem diferente da encontrada na literatura.

### 5.1. Implementação

Foram implementadas as meta-heurísticas LNS e ALNS para a resolução do problema. As seções seguintes apresentam uma breve descrição destas implementações.

#### 5.1.1. Solução de localização

A solução de localização do problema é apresentada na forma de um vetor binário, sendo que o tamanho do vetor corresponde ao número de locais candidatos para a instalação das facilidades. O elemento  $i$  do vetor  $y$  é 1 quando o local candidato  $i$  é selecionado para a instalação de uma facilidade, e 0 caso contrário. Por exemplo, supondo que o número de facilidades candidatas seja dez e que o número de facilidades a serem instaladas seja três, uma possível solução de localização seria:

$$y: \begin{array}{|c|c|c|c|c|c|c|c|c|c|} \hline 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ \hline \end{array}$$

Figura 5.1 - Representação de uma solução de localização.

E as facilidades seriam instaladas nos locais candidatos: 3, 5 e 10.

#### 5.1.2. Solução inicial

Uma solução inicial aleatória é gerada para inicializar o algoritmo. Para esta solução, seleciona-se ao acaso  $p$  posições do vetor solução, inicializado com o valor 0 em todas as posições, para serem iguais a 1. Desse modo, temos uma solução inicial que respeita a restrição de cardinalidade e garante sua viabilidade.

### 5.1.3. Solução de alocação

Após a geração da solução de localização, as variáveis de alocação são determinadas, usando um método exato, como solução do seguinte problema:

$$v(\text{subproblema}) = \text{Max} \left\{ \sum_{i \in I} \sum_{j \in N_i} a_i x_{ij} \right\} \quad 5.1$$

Sujeito a:

$$\sum_{j \in N_i} x_{ij} \leq 1, \quad \forall i \in I \quad 5.2$$

$$x_{ij} \leq Y_j, \quad \forall i \in I \text{ e } j \in N_i \quad 5.3$$

$$\sum_{i \in I} f_i x_{ij} \leq \mu^{b+2} \sqrt{1 - \varphi}, \quad \forall j \quad 5.4$$

ou

$$\sum_{i \in I} f_i x_{ij} \leq \mu + \frac{1}{\tau} \ln(1 - \varphi), \quad \forall j \quad 5.5$$

$$x_{ij} \in \{0,1\}, \quad \forall i \in I \text{ e } \forall j \quad 5.6$$

Neste modelo,  $Y_j$  são constantes que correspondem à solução de localização gerada após a reparação e a restrição (2.7) é considerada implicitamente.

### 5.1.4. Funções de destruição e reparo do LNS

Como mencionado anteriormente, o método de destruição transforma uma solução viável em uma solução inviável. Pode-se tornar uma solução inviável violando uma restrição do problema.

De acordo com Ropke e Pisinger (2006b), a destruição da solução deve ser feita considerando o problema estudado. Na implementação da meta-heurística LNS foi utilizada uma função de destruição aleatória com porcentagem fixa.



Esta função destrói 50% das facilidades instaladas, pois a quantidade de facilidades utilizadas nos testes computacionais iniciais é pequena – há instâncias com apenas duas facilidades – e, portanto, faz-se necessário destruir metade ou mais do vetor solução para que o mesmo mude em todos os casos testados, pelo menos algumas vezes.

Escolhemos aleatoriamente, entre as posições que têm valor igual a 1, uma posição do vetor solução, e o substituímos por 0, tornando assim a solução inviável, pois desrespeita a restrição de cardinalidade que define a quantidade de facilidades.

Consideramos por exemplo, um vetor de 10 posições com  $p = 4$ . Destruiremos, portanto, 2 posições desse vetor. As soluções inicial e após a destruição são mostradas nas Figura 5.2 e Figura 5.3, respectivamente.

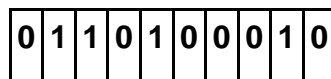


Figura 5.2 - Solução inicial.

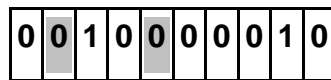


Figura 5.3 - Solução destruída.

O método de reparo recupera a viabilidade da solução anteriormente destruída e, assim como para o método de destruição, tem-se a liberdade de escolhê-lo. Aconselha-se escolher um método que trabalhe bem com as particularidades do problema estudado. Para o LNS, implementou-se uma função de reparo aleatória.

Nesta função, escolhemos de forma aleatória os elementos que tem o valor 0, até que a solução respeite as restrições do problema, ou seja, até que hajam novamente  $p$  centros.

Considerando o exemplo da Figura 5.2, um vetor de 10 posições com  $p = 4$ , sendo que as posições 2 e 5 foram modificadas, uma possível solução reparada seria como segue na Figura 5.4.

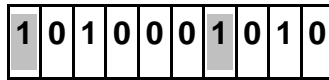


Figura 5.4 - Solução reparada.

### 5.1.5. Funções de destruição e reparo do ALNS

Como o ALNS tem múltiplas funções de destruição e reparo, utilizamos o método da roleta descrito na equação (4.1) para a escolha das funções. Quando uma iteração do ALNS é concluída, uma pontuação  $\psi$  para os métodos de destruição e reparação é calculada usando a fórmula (4.2). Na implementação em questão utilizou-se  $\delta_1 = 8$ ,  $\delta_2 = 4$ ,  $\delta_3 = 1$  e  $\delta_4 = 0$ .

Os pesos são atualizados, após uma sequência de  $k = 20$  iterações, seguindo a fórmula descrita em (4.3) com parâmetro de decaimento  $\beta = 0,35$ , denotando que cada atualização de determinado peso de dada função é influenciada por 35% do peso anterior e 65% dos pontos atribuídos a ele.

Atendendo a exigência de múltiplas funções de destruição e reparo, utilizamos, além das funções usadas no LNS, as funções descritas a seguir.

#### 5.1.5.1 Funções de destruição

a) Função de destruição aleatória, com porcentagem não-fixa

Destrói aleatoriamente  $d < p$  facilidades, sendo o fator de destruição  $d$  é escolhido aleatoriamente a cada execução da função.

b) Função que destrói aleatoriamente uma das duas facilidades mais próximas

Identifica quais são as facilidades com distância entre elas menor do que o raio de cobertura definido para a rede e destrói uma delas aleatoriamente para cada par de facilidades, a fim de evitar sobreposições. Se não houver facilidades nesta condição, remove-se uma de duas facilidades mais próximas.

c) Função que destrói a facilidade com menor demanda associada a ela

Identifica a demanda total atendida por cada facilidade e elimina a facilidade

com a menor demanda associada. Caso seja aplicada à solução inicial que ainda não possui a solução de alocação, elimina-se a facilidade com menor demanda, cujo ponto de demanda é associado à facilidade somente se estiver dentro do raio de cobertura de mesma.

#### **5.1.5.2 Funções de reparação**

- a) Função que repara inserindo as facilidades que estiverem mais distantes de alguma facilidade mais próxima já instalada

A fim de cobrir melhor o espaço de soluções atendendo clientes ainda não cobertos, identificam-se as distâncias entre os potenciais pontos de instalação e as facilidades existentes mais próximas a eles. Insere-se um a um na solução, em ordem decrescente da distância aferida até a facilidade mais próxima, até que o número de facilidades exigido seja atingido.

- b) Função de reparo que insere as facilidades que tem clientes com maiores demandas

Esta função pretende atender clientes ainda não cobertos pelas facilidades já instaladas, visando somente os maiores pontos de demanda dentro do raio de cobertura da facilidade.

- c) Função de reparo que insere as facilidades que tem o maior número de clientes

Esta função pretende atender o maior número de clientes possível dentro do raio de serviço da facilidade, inserindo as facilidades que o possuem, em ordem decrescente de quantidade.

#### **5.1.6. Aceitação da solução**

Em Shaw (1998), a solução reparada era aceita se melhorasse o valor da função objetivo. Em trabalhos posteriores, como Ropke e Pisinger (2006b), usaram um critério de aceitação baseado no *Simulated Annealing* (SA).

Com este critério de aceitação, se a solução corrente melhora o valor da

função objetivo, então ela é sempre aceita; caso a solução piore tal valor, então ela é aceita com probabilidade  $e^{-\frac{c(x^t)-c(x)}{T}}$ , sendo  $c(x^t)$  o valor da função objetivo da solução temporária e  $c(x)$  o valor da função objetivo da solução atual. A temperatura  $T$  é inicializada em  $T_0 > 0$  e diminui gradualmente realizando a atualização  $T_{novo} = \alpha T_{corrente}$  a cada iteração utilizando o parâmetro  $\alpha$ , sendo  $0 < \alpha < 1$ . A ideia é que  $T$  seja relativamente alta inicialmente, permitindo que soluções de piora sejam mais facilmente aceitas no início da busca. Neste trabalho, adotamos os parâmetros  $T_0 = 100000$  e  $\alpha = 0.95$ .

Com o decorrer da busca, a probabilidade de aceitar soluções de piora diminui. Isso faz com que, no início, haja uma busca maior no espaço de soluções, possibilitando assim fugir de ótimos locais. Esta ideia garante que todas as soluções têm certa probabilidade de serem aceitas, conforme a etapa em que se encontra o algoritmo.

## 5.2. Resultados

O LNS e o ALNS foram testados na rede de 30 pontos fornecida em Marianov e Serra (1998), nas redes de 324 pontos e de 818 pontos obtidas de uma base de dados geográficos da cidade de São José dos Campos-SP, além da rede de 1177 pontos desenvolvida neste trabalho, acrescidas da população fictícia em cada ponto de demanda, e que estão disponíveis em <http://www.lac.inpe.br/~lorena/instancias.html>. Cada ponto de demanda é um potencial centro de atendimento e as distâncias são euclidianas. Para a rede de 30 pontos foram considerados: raio de cobertura ( $S$ ) igual a 1,5 milhas, tempo médio de atendimento igual a 20 minutos, taxa de chamada diária igual a 0,015 vezes a demanda do ponto para as restrições do comprimento de fila ( $f_i$ ) e 0,006 vezes a demanda do ponto para as restrições de tempo de espera, todos definidos em Marianov e Serra (1998). Para as redes de 324, 818 e 1177 pontos considerou-se raio de cobertura igual a 250 m, 750 m e 950 m, respectivamente. Para estas últimas três redes, fixou-se o tempo médio de atendimento igual a 15 minutos e taxa de chamada igual a 0,01 vezes a demanda do ponto.

O valor de  $\mu$  pode ser calculado da seguinte forma: como a taxa de chegada é diária, temos que a taxa de atendimento também deve ser e, portanto, devemos saber quantos clientes podem ser atendidos por dia em cada facilidade, ou seja, temos que  $\mu = \frac{24*60}{20} = 72$ , para a rede de 30 pontos e  $\mu = \frac{24*60}{15} = 96$  para as de 324, 818 e 1177 pontos.

Vários problemas foram gerados, variando-se os parâmetros  $p, b, \tau$  e  $\varphi$ . Os resultados dessa abordagem foram comparados aos obtidos pelo uso do *software* comercial CPLEX 12.5 [ILOG, 2012] e aos resultados de Corrêa, Lorena e Ribeiro (2009), que utilizaram Geração de Colunas (GC). Os valores apresentados no artigo de Marianov e Serra (1998) foram obtidos por meio de uma heurística e pelo uso do CPLEX versão 3.0. A versão 12.5 do CPLEX mostrou melhores resultados para a mesma formulação matemática que os apresentados naquele artigo, para as duas soluções (heurística e CPLEX 3.0). Portanto, apenas os resultados obtidos com a versão 12.5 foram considerados para comparar com os resultados obtidos com o LNS e com o ALNS para a rede de 30 pontos.

Os nomes dos problemas foram codificados do seguinte modo: quantidade de facilidades para instalar, quantidade de pessoas na fila, tipo de restrição ('0' para restrições de quantidade de pessoas na fila ou '1' para restrições de tempo de atendimento) e probabilidade mínima de encontrar  $b$  pessoas na fila ou probabilidade mínima de esperar  $\tau$  minutos para ser atendido.

Os problemas foram resolvidos em um computador Intel Core i5 2,4GHz com 4GB RAM. Os programas para chamada das funções do CPLEX, para o LNS e ALNS foram codificados em C. Os resultados são apresentados nas tabelas 5.1, 5.2, 5.3 e 5.4, contendo as seguintes colunas: o número do problema; os parâmetros utilizados; a solução do CPLEX; a solução do GC encontrada em Corrêa et al. (2009); a solução do LNS e a solução do ALNS. Na coluna referente ao CPLEX são fornecidos: a solução inteira viável e o *gap* obtidos do CPLEX, com o tempo limitado em 7800 segundos para a rede de 1177 e em 3600 segundos (1 hora) para as demais redes. O valor de *Gap Cplex* igual a zero define que o valor ótimo foi obtido. Esse valor é calculado pelo CPLEX

como  $\frac{100*(\text{limite superior}-\text{limite inferior})}{\text{limite inferior}}$ , e fornece, a diferença percentual entre esses limitantes.

Nas colunas do LNS e ALNS, são apresentados a melhor solução, a média das soluções (com 10 execuções independentes) com o tempo computacional limitado em 3600 segundos ou quando  $T \leq 0.001$  ou quando o algoritmo atingir 100 iterações de piora na função objetivo, sendo que o tempo médio das 10 execuções é apresentado, assim como o *Gap*. Os *Gaps* LNS e ALNS são calculados por  $\frac{100*(\text{melhor solução do CPLEX} - \text{melhor solução do LNS/ALNS})}{\text{melhor solução do LNS/ALNS}}$ , e fornecem, em porcentagem, os *gaps* entre os valores da solução. Na última coluna, encontra-se a solução do GC, o *gap* calculado por  $\frac{100*(\text{melhor solução do CPLEX} - \text{melhor solução do GC})}{\text{melhor solução do GC}}$  e o tempo computacional relatado. Se o *gap* for igual a zero, significa que o método encontrou a mesma solução que o CPLEX e se for inferior a zero, a solução obtida pelo respectivo método é melhor que a solução do CPLEX.

A última linha de cada tabela apresenta os valores médios das execuções, o que permite avaliar o desempenho global e a consistência das soluções obtidas por cada método.

A Tabela 5.1 apresenta os resultados computacionais para a rede de 30 pontos.

Tabela 5.1 - Resultados para a rede de 30 pontos.

Problema	Parâmetros	CPLEX			GC			LNS				ALNS			
		Solução	Gap (%)	Tempo (s)	Melhor solução	Gap (%)	Tempo (s)	Melhor solução	Média das soluções	Gap (%)	Tempo (s)	Melhor solução	Média das soluções	Gap (%)	Tempo (s)
1	2_0_0_85	3700	0,240	0,19	3700	0,000	0,53	3700	3700	0,000	1,99	3700	3700	0,000	2,06
2	3_0_0_85	5390	0,130	0,11	5390	0,000	28,03	5390	5365	0,000	5,25	5390	5372	0,000	4,24
3	2_0_1_85	5100	0,000	0,39	5090	0,196	5,55	5100	5073	0,000	1,35	5100	5014	0,000	1,74
4	3_0_1_85	5390	0,190	0,25	5390	0,000	3,77	5390	5390	0,000	1,50	5390	5355	0,000	1,88
5	2_0_2_85	5210	1,360	0,47	5210	0,000	4,77	5210	5101	0,000	1,35	5210	5095	0,000	1,95
6	3_0_2_85	5390	0,160	0,06	5390	0,000	1,88	5390	5371	0,000	1,56	5390	5339	0,000	1,42
7	5_0_0_95	5330	0,380	3600,00	5330	0,000	6,91	5330	5328	0,000	41,30	5330	5315	0,000	77,37
8	6_0_0_95	5410	0,370	3600,00	5410	0,000	42,41	5410	5392	0,000	34,19	5410	5330	0,000	30,65
9	3_0_1_95	5270	0,190	1286,71	5270	0,000	14,75	5270	5236	0,000	5,10	5270	5212	0,000	1,46
10	4_0_1_95	5390	0,190	1,23	5390	0,000	30,88	5390	5390	0,000	3,06	5390	5362	0,000	3,87
11	2_0_2_95	4520	13,240	0,36	4520	0,000	0,70	4520	4520	0,000	2,04	4520	4520	0,000	2,58
12	3_0_2_95	5390	0,120	0,06	5390	0,000	11,45	5390	5357	0,000	2,82	5390	5358	0,000	2,16
13	4_1_48_90	1920	0,000	0,42	1920	0,000	0,44	1920	1920	0,000	9,88	1920	1920	0,000	21,07
14	5_1_48_90	2400	0,000	0,70	2400	0,000	3,08	2400	2400	0,000	23,52	2400	2400	0,000	30,81
15	3_1_49_90	2160	0,090	0,17	2160	0,000	0,36	2160	2160	0,000	3,42	2160	2160	0,000	2,87
16	4_1_49_90	2880	0,070	0,22	2880	0,000	0,45	2880	2880	0,000	6,19	2880	2880	0,000	5,96
17	5_1_50_90	4700	16,380	0,81	4700	0,000	0,83	4700	4700	0,000	75,77	4700	4170	0,000	113,75
18	6_1_50_90	5390	1,480	0,73	5390	0,000	48,44	5390	5390	0,000	44,90	5390	5210	0,000	35,83
19	6_1_41_85	5330	0,190	3600,00	5300	0,566	11,74	5330	5321	0,000	91,69	5330	5100	0,000	122,36
20	7_1_41_85	5410	0,180	3600,00	5350	1,121	41,84	5410	5394	0,000	49,47	5390	5156	0,371	28,59
21	8_1_41_85	5470	0,000	0,09	5470	0,000	1,20	5470	5470	0,000	8,03	5470	5205	0,000	47,34
22	4_1_42_85	4600	0,000	0,31	4600	0,000	0,81	4600	4600	0,000	30,40	4600	4600	0,000	36,06
23	5_1_42_85	5390	1,480	0,86	5390	0,000	34,17	5390	5384	0,000	49,95	5390	5330	0,000	38,57
Média			1,584	682,35		0,082	12,83			0,000	21,51			0,016	26,72

Observamos que o LNS encontrou o melhor valor para todas as instâncias assim como o CPLEX. O ALNS encontrou valor inferior ao CPLEX em um caso. Das 23 instâncias resolvidas pelo GC, em 3 delas não foi encontrado o melhor valor. Para esta rede, os métodos LNS e ALNS obtiveram soluções melhores do que as obtidas pelo GC. Os *gaps* relativos ao CPLEX estão dispostos na Figura 5.5.

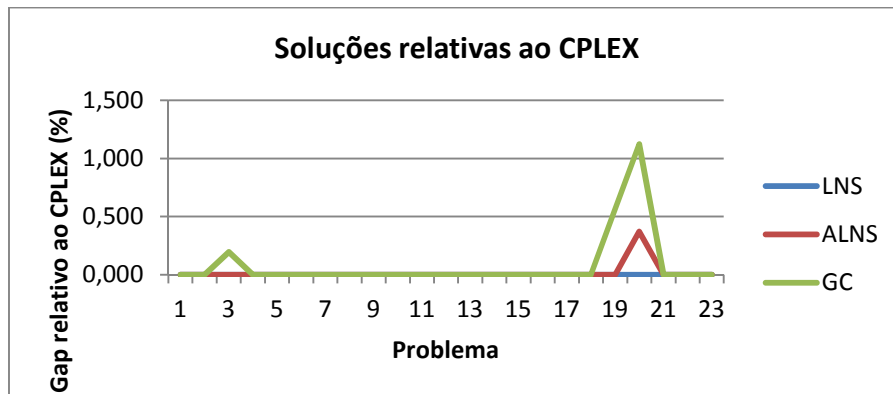


Figura 5.5 - Soluções relativas ao CPLEX da rede de 30 pontos.

A Figura 5.6 mostra os tempos computacionais das soluções da rede de 30 pontos computados pelos métodos.

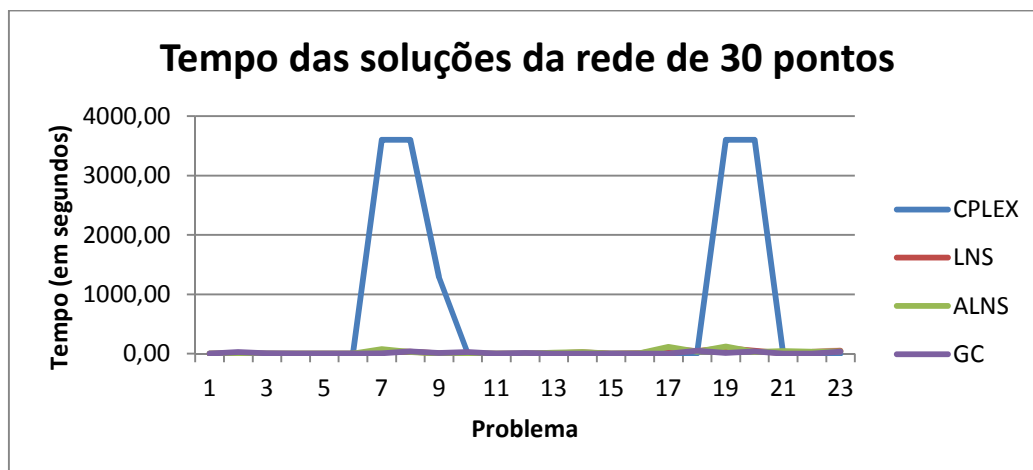


Figura 5.6 - Tempo computacional das soluções da rede de 30 pontos.

Nota-se que o CPLEX apresenta dois picos significativos nas oscilações, ou seja, o CPLEX não provou a otimalidade das soluções encontradas dentro do tempo estipulado, enquanto os restantes dos métodos se mantiveram mais estáveis. As meta-heurísticas LNS e ALNS obtiveram tempos competitivos



ainda que o GC tenha alcançado tempos melhores na maior parte dos casos.

A Tabela 5.2 mostra os resultados obtidos para a rede de 324 pontos.

Tabela 5.2 - Resultados para a rede de 324 pontos.

Problema	Problema	CPLEX			GC			LNS				ALNS			
		Solução	Gap (%)	Tempo (s)	Melhor Solução	Gap (%)	Tempo (s)	Melhor solução	Média das soluções	Gap (%)	Tempo (s)	Melhor solução	Média das soluções	Gap (%)	Tempo (s)
1	10_0_0_85	37177	0,010	85,97	37180	-0,008	275,11	37094	36881	0,224	565,22	36447	33695,8	2,003	206,69
2	10_0_1_85	51000	0,020	3600,00	51000	0,000	507,81	50466	50020	1,058	438,06	49133	45322,3	3,800	133,66
3	10_0_2_85	59739	0,010	96,70	59740	-0,002	604,48	59205	58522	0,902	367,35	57743	55363	3,457	112,28
4	10_0_0_95	21460	0,010	223,17	21460	0,000	65,20	21458	21385	0,009	1561,93	20920	20285,5	2,581	905,03
5	10_0_1_95	35360	0,020	3600,00	35360	0,000	47,33	35359	35260	0,003	712,99	34716	34014	1,855	91,91
6	10_0_2_95	45390	0,010	3600,00	45390	0,000	327,84	45370	45208	0,044	942,98	45043	42255,5	0,770	110,60
7	20_0_0_85	74352	0,010	3600,00	74358	-0,008	2630,44	73460	72480	1,214	1194,28	71071	66245,1	4,617	177,31
8	20_0_1_85	101965	0,050	3600,00	101973	-0,008	10801,33	97385	95602,4	4,703	153,10	95489	90005	6,782	100,58
9	20_0_2_85	119451	0,030	3600,00	119448	0,003	10801,59	113635	110931,5	5,118	81,42	112713	101100,9	5,978	77,84
10	20_0_0_95	42918	0,010	311,17	42920	-0,005	2425,52	42482	42217	1,026	3600,00	41557	40584,7	3,275	315,17
11	20_0_1_95	70719	0,020	3600,00	70720	-0,001	1067,41	70282	69622	0,622	1782,71	70017	67511	1,003	225,06
12	20_0_2_95	90768	0,03	3600,00	90778	-0,011	2369,98	89573	88330	1,334	1561,07	90638	90191,9	0,143	764,70
13	10_1_40_85	27699	0,010	397,66	27700	-0,004	238,39	27599	27391	0,362	1329,69	27530	27110	0,614	46,94
14	10_1_41_85	29360	0,030	3600,00	29360	0,000	145,88	29353	29132	0,024	853,67	28970	28619	1,346	309,86
15	10_1_42_85	30950	0,020	3600,00	30950	0,000	42,91	30936	30819	0,045	1395,98	30948	30550	0,006	279,14
16	10_1_48_90	26920	0,010	115,16	26920	0,000	144,72	26876	26699	0,164	1272,98	26920	26557	0,000	146,63
17	10_1_49_90	28329	0,010	398,07	28330	-0,004	315,70	28256	28084	0,258	1347,13	28328	27815	0,004	279,49
18	10_1_50_90	29680	0,020	3600,00	29680	0,000	81,81	29673	29580	0,024	1107,83	29673	29099,6	0,024	175,52
19	20_1_40_85	55394	0,020	3600,00	55396	-0,004	8457,66	54787	53219	1,108	2321,86	54249	52811,6	2,111	642,27
20	20_1_41_85	58709	0,050	3600,00	58720	-0,019	1377,42	58006	56987,6	1,212	440,11	56775	55942,8	3,406	839,51
21	20_1_42_85	61892	0,030	3600,00	61900	-0,013	735,14	61316	60401,4	0,939	618,74	61672	61419,1	0,357	1212,61
22	20_1_48_90	53837	0,010	3600,00	53838	-0,002	4645,49	52686	52325	2,185	431,37	53400	53106,6	0,818	1488,96
23	20_1_49_90	56653	0,010	3600,00	56654	-0,002	10800,92	55514	54858,9	2,052	470,58	55711	54447,2	1,691	1335,84
24	20_1_50_90	59357	0,020	3600,00	59360	-0,005	1273,39	58695	57997	1,128	582,73	59232	57008	0,211	1070,90
Média			0,019	2617,83		-0,004	2507,65			1,073	1047,24			1,952	460,354

O LNS foi inferior ao CPLEX quanto à qualidade da solução em todas as instâncias testadas e o ALNS alcançou o mesmo valor da solução que o CPLEX somente em um caso em todos os casos testados com as meta-heurísticas. O GC apresentou soluções iguais ou melhores do que o CPLEX em 23 dos 24 casos. Os métodos obtiveram soluções semelhantes com *gaps* baixos ou inferiores a zero em relação ao CPLEX. Os *gaps* relativos ao CPLEX estão dispostos na Figura 5.7.

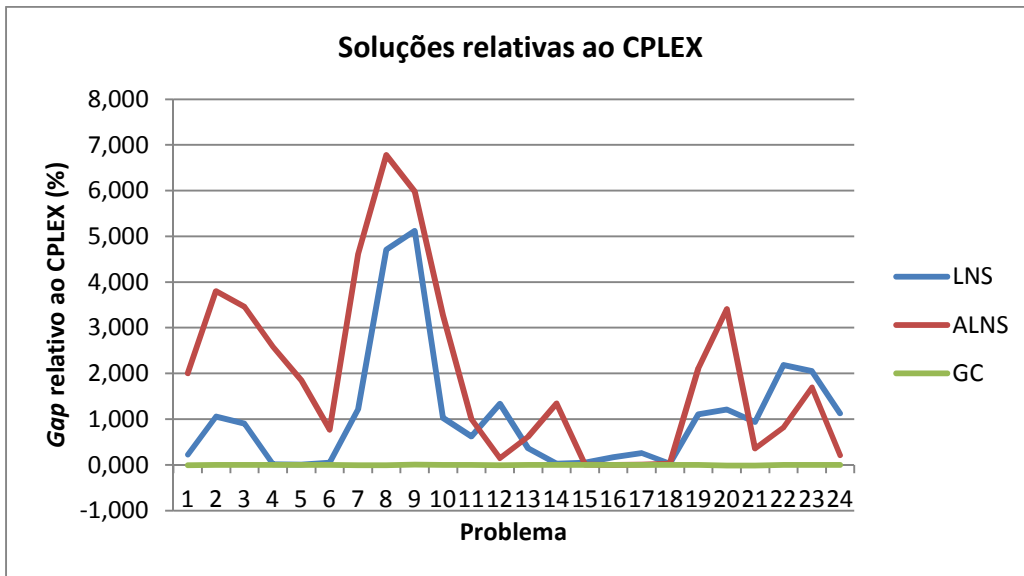


Figura 5.7 - Soluções relativas ao CPLEX da rede de 324 pontos.

Apesar das soluções serem relativamente próximas, podemos notar na Figura 5.8 que há grande diferença em relação ao tempo computacional. O GC obteve a solução em tempos computacionais altos em inúmeros testes, assim como o CPLEX. Para esta rede, o melhor desempenho, em média, de tempo foram obtidos pelos métodos LNS e ALNS.

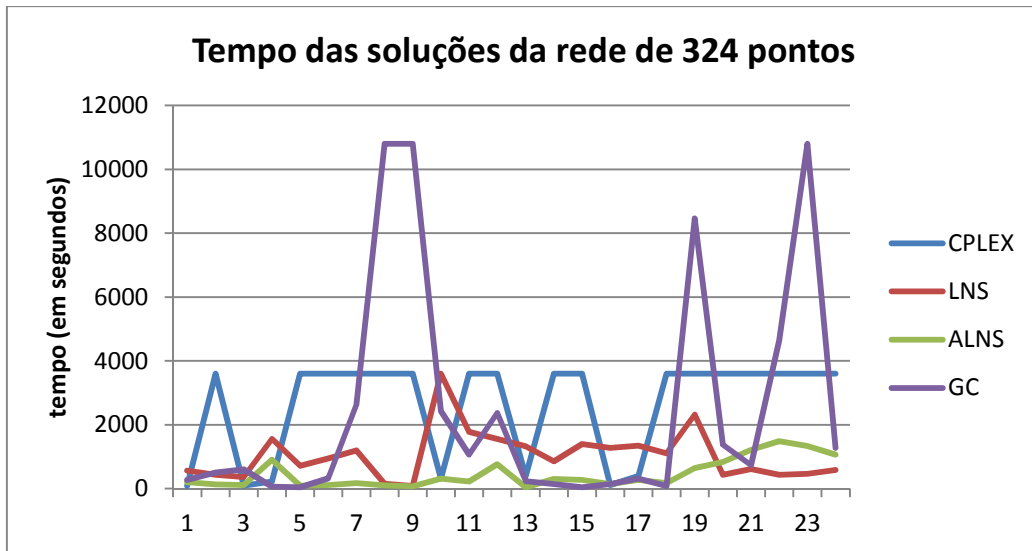


Figura 5.8 - Tempo computacional das soluções da rede de 324 pontos.

Para a rede de 818 pontos, a Tabela 5.3 mostra os resultados.

Tabela 5.3 - Resultados para a rede de 818 pontos.

Problema	Parâmetros	CPLEX			GC			LNS				ALNS			
		Solução	Gap (%)	Tempo (s)	Melhor Solução	Gap (%)	Tempo (s)	Melhor solução	Média das soluções	Gap (%)	Tempo (s)	Melhor solução	Média das soluções	Gap (%)	Tempo (s)
1	10_0_0_95	21460	0,300	3600,00	21460	0,000	163,08	21460	21460	0,000	227,28	21460	21460	0,000	138,94
2	10_0_1_95	35263	0,290	3600,00	35360	-0,274	207,69	35360	35360	-0,274	727,30	35360	35360	-0,274	443,75
3	10_0_2_95	45390	0,010	3600,00	45390	0,000	251,33	45390	45390	0,000	754,53	45390	45390	0,000	401,01
4	20_0_0_85	74350	0,020	3600,00	74360	-0,013	412,74	74358	74333,6	-0,011	1586,22	74350	74283	0,000	1244,54
5	20_0_1_85	96011	462,630	3600,00	102000	-5,872	447,06	102000	101998,2	-5,872	1539,06	101996	101972	-5,868	1464,84
6	20_0_2_85	111425	384,890	3600,00	119480	-6,742	473,52	119478	119473,7	-6,740	1639,93	119478	119471	-6,740	1337,38
7	20_0_0_95	42920	0,030	3600,00	42920	0,000	248,13	42920	42918,6	0,000	921,40	42920	42920	0,000	751,65
8	20_0_1_95	70717	0,020	3600,00	70720	-0,004	390,03	70720	70709,3	-0,004	1546,80	70716	70708	0,001	2212,48
9	20_0_2_95	90780	0,010	3600,00	90780	0,000	460,19	90780	90778	0,000	1494,27	90779	90696	0,001	785,14
10	20_1_48_90	53834	0,020	3600,00	53840	-0,011	343,08	53839	53818,4	-0,009	1880,33	53837	53835	-0,006	1518,79
11	20_1_49_90	56656	0,010	3600,00	56660	-0,007	324,64	56659	56638,1	-0,005	1231,01	56658	56651	-0,004	1912,34
12	20_1_50_90	59351	0,030	3600,00	59360	-0,015	364,84	59358	59352	-0,012	1418,01	59359	59356	-0,013	980,89
13	50_0_0_85	185865	0,020	3600,00	185898	-0,018	3695,94	185061	184686,5	0,434	1587,64	185134	184385	0,395	1521,46
14	50_0_1_85	13975	3800,000	3600,00	255000	-94,520	5072,75	254394	253802,3	-94,507	1732,06	254760	254283	-94,514	2623,07
15	50_0_2_85	296111	82,460	3600,00	298490	-0,797	6004,53	297884	297414	-0,595	1650,01	298356	298117	-0,752	2232,68
16	50_1_48_90	134564	0,040	3600,00	134600	-0,027	2317,55	134267	133871,6	0,221	1852,20	134438	133812	0,094	1231,28
17	50_1_49_90	141633	0,020	3600,00	141650	-0,012	2713,86	141073	140443,5	0,397	1562,06	141267	140936	0,259	2152,18
18	50_1_50_90	148386	0,030	3600,00	148400	-0,009	2444,64	148099	147476,2	0,194	1560,37	146304	146290	1,423	1543,38
19	10_1_48_90	26920	0,010	1976,24	26920	0,000	185,41	26920	26920	0,000	524,94	26920	26920	0,000	501,25
20	10_1_49_90	28330	0,010	2493,61	28330	0,000	196,06	28330	28330	0,000	379,33	28330	28330	0,000	338,24
21	10_1_50_90	29680	0,020	3600,00	29680	0,000	199,28	29680	29680	0,000	414,13	29680	29680	0,000	517,75
22	20_1_40_85	55399	0,020	3600,00	55400	-0,002	293,62	55397	55384,4	0,004	1403,69	55399	55394	0,000	1260,12
23	20_1_41_85	58711	0,050	3600,00	58720	-0,015	372,28	58717	58712	-0,010	1609,76	58718	58691	-0,012	1250,76
24	20_1_42_85	61898	0,020	3600,00	61900	-0,003	380,64	61900	61899	-0,003	1578,40	61900	61900	-0,003	1744,32
Média			197,123	3486,24		-4,514	1165,12			-4,450	1284,20			-4,417	1254,51

As soluções foram próximas exceto as obtidas com o método CPLEX, que obteve um desempenho muito inferior em um caso, como é mostrado na Figura 5.9. Em 19 dos 24 casos, o LNS obteve resultados iguais ou melhores que o CPLEX, o ALNS obteve o mesmo êxito em 18 casos, enquanto o GC alcançou os melhores resultados desta rede.

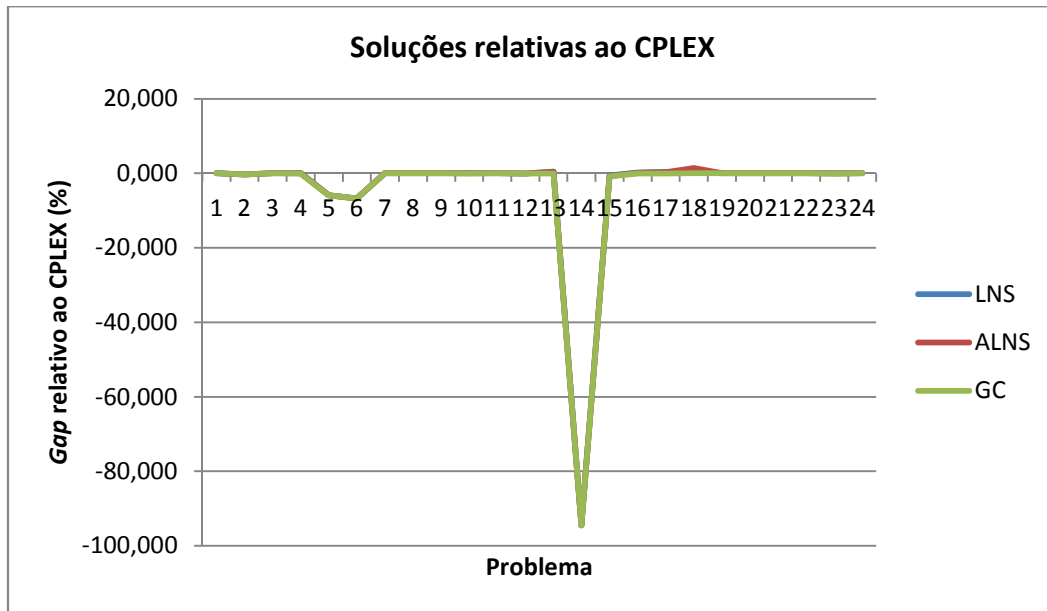


Figura 5.9 - Soluções relativas ao CPLEX da rede de 818 pontos.

Já em relação aos tempos computacionais da rede de 818 pontos, temos o CPLEX obtendo o maior tempo computacional em quase todos os casos (tempo limite estabelecido para o processamento) e o GC sendo inferior em mais da metade dos casos, mas atingindo tempos altos em pelo menos 25% dos casos. Os métodos LNS e ALNS obtiveram tempos inferiores a 2700 segundos, com comportamento mais constante, como pode-se observar na Figura 5.10.

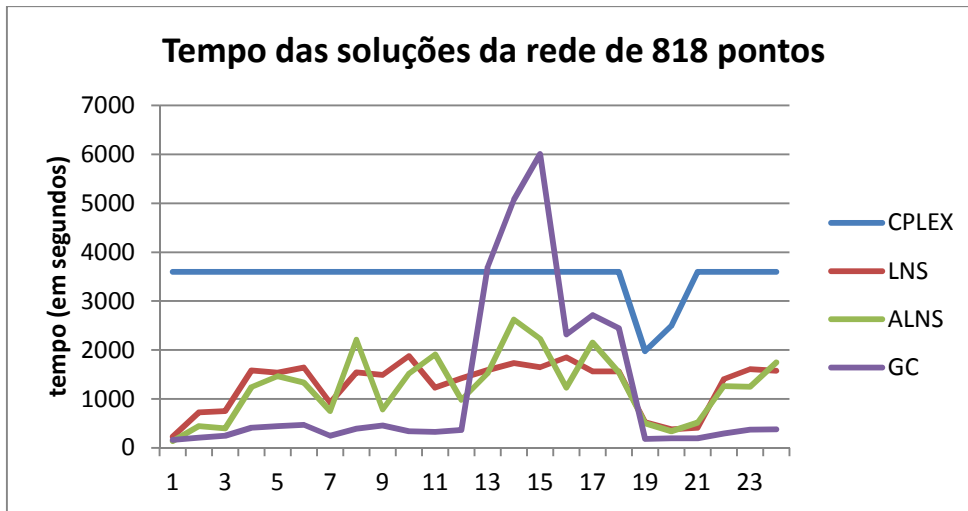


Figura 5.10 - Tempo computacional das soluções da rede de 818 pontos.

Para obter a rede de 1177 pontos, apresentada na Figura 5.11, foram acrescentados 359 pontos à rede já existente de 818 pontos. O objetivo era criar uma rede maior e com restrições que pudessem tornar o problema ainda mais difícil para verificar o desempenho das meta-heurísticas em relação ao CPLEX.

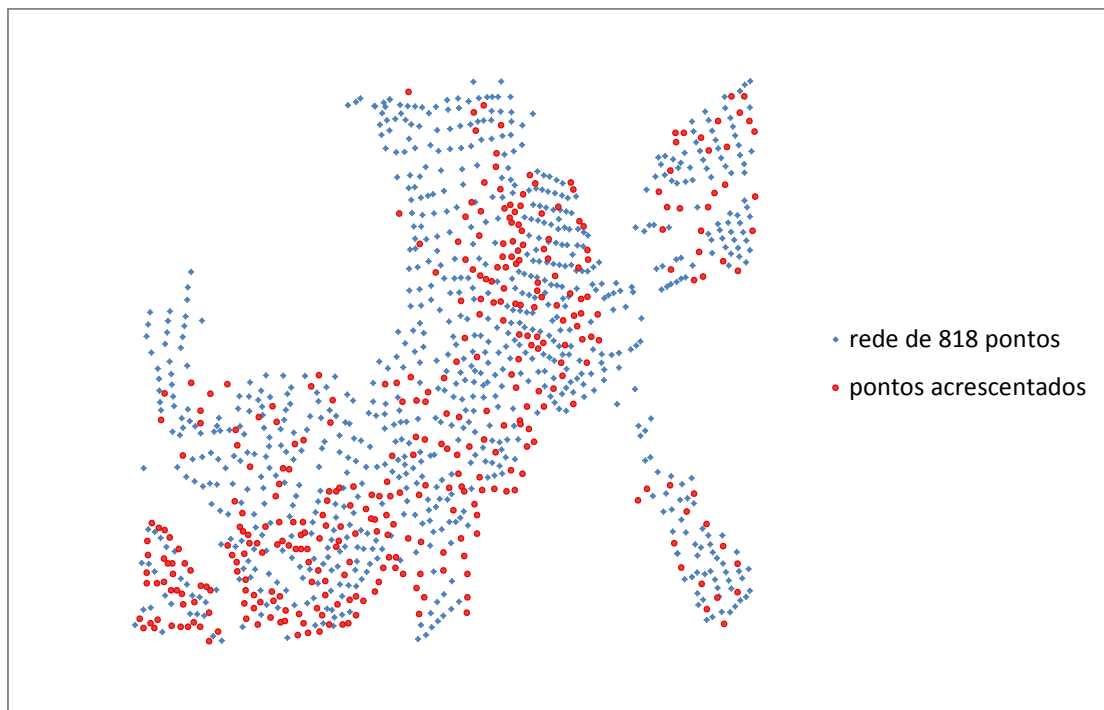


Figura 5.11 - Rede de 1177 pontos.

Os resultados para a rede de 1177 pontos são apresentados na Tabela 5.4.

Tabela 5.4 - Resultados para a rede de 1177 pontos.

Problema	Parâmetros	CPLEX			LNS				ALNS			
		Solução	Gap (%)	Tempo (s)	Melhor solução	Média das soluções	Gap (%)	Tempo (s)	Melhor solução	Média das soluções	Gap (%)	Tempo (s)
1	10_0_0_95	21148	3400,000	7800,00	21460	21460	-1,454	278,98	21460	21460	-1,454	316,21
2	10_0_1_95	34919	2000,000	7800,00	35360	35360	-1,247	461,92	35360	35360	-1,247	410,79
3	10_0_2_95	45300	1500,000	7800,00	45390	45390	-0,198	320,46	45390	45390	-0,198	326,44
4	20_0_0_85	73987	893,420	7800,00	74360	74359,5	-0,502	1104,16	74360	74360	-0,502	1126,64
5	20_0_1_85	99837	636,200	7800,00	102000	102000	-2,121	1054,29	102000	102000	-2,121	918,19
6	20_0_2_85	117315	526,520	7800,00	119480	119479,7	-1,812	1147,35	119480	119479,2	-1,812	1112,38
7	20_0_0_95	42579	1600,000	7800,00	42920	42920	-0,795	969,56	42920	42920	-0,795	817,61
8	20_0_1_95	69398	959,110	7800,00	70720	70718,7	-1,869	1143,80	70718	70714,6	-1,867	977,42
9	20_0_2_95	90222	714,660	7800,00	90780	90779,7	-0,615	1081,05	90780	90780	-0,615	952,08
10	20_1_48_90	53421	1300,000	7800,00	53840	53839,5	-0,778	1081,09	53840	53840	-0,778	968,58
11	20_1_49_90	56338	1200,000	7800,00	56660	56658,7	-0,568	1019,18	56659	56655,4	-0,567	1030,69
12	20_1_50_90	58468	1200,000	7800,00	59360	59359,3	-1,503	967,71	59330	59140	-1,453	954,77
13	50_0_0_85	182622	302,470	7800,00	185604	185375,7	-1,607	1383,59	185446	185106,9	-1,523	1602,07
14	50_0_1_85	218949	235,690	7800,00	254941	254634,4	-14,118	1263,21	254617	253916,1	-14,008	1022,36
15	50_0_2_85	224659	227,160	7800,00	298554	298380,1	-24,751	1386,39	298133	297747	-24,645	1233,76
16	50_1_48_90	129818	466,180	7800,00	134578	134557,6	-3,537	1282,44	134580	134579,3	-3,538	1466,03
17	50_1_49_90	139068	428,520	7800,00	141498	141357,6	-1,717	1432,09	141397	140873,2	-1,647	1545,12
18	50_1_50_90	145314	405,800	7800,00	148388	148207,4	-2,072	1399,09	148244	148099,1	-1,976	1301,71
19	10_1_48_90	26615	2700,000	7800,00	26920	26920	-1,133	360,93	26920	26920	-1,133	185,29
20	10_1_49_90	28145	2500,000	7800,00	28330	28330	-0,653	366,23	28330	28330	-0,653	298,12
21	10_1_50_90	29497	2400,000	7800,00	29680	29680	-0,617	352,61	29680	29680	-0,617	428,09
22	70_1_40_85	189152	288,580	7800,00	193703	193465,4	-2,349	1345,05	193239	193043,3	-2,115	1350,89
23	70_1_41_85	190192	286,450	7800,00	204980	204616	-7,214	1282,55	204967	204894,3	-7,208	1124,40
24	70_1_42_85	193836	279,190	7800,00	216352	216062,8	-10,407	1421,00	216285	216272,2	-10,379	1291,20
Média			1102,08	7800,00			-3,485	996,03			-3,452	948,37



Podemos observar que o LNS e o ALNS obtiveram as melhores soluções em 100% das instâncias, em tempos computacionais significativamente inferiores.

Na Figura 5.12 apresentam-se as soluções obtidas pelos métodos LNS e ALNS relativas ao CPLEX para a rede de 1177. As meta-heurísticas LNS e ALNS obtiveram desempenhos semelhantes e ambas as abordagens foram superiores ao CPLEX, pois alcançaram soluções melhores em 100% dos casos analisados. Além de resultados inferiores, o CPLEX obteve *gap* superior a 227% para todos os casos testados, sendo que o maior *gap* obtido alcançou a ordem de milhares.

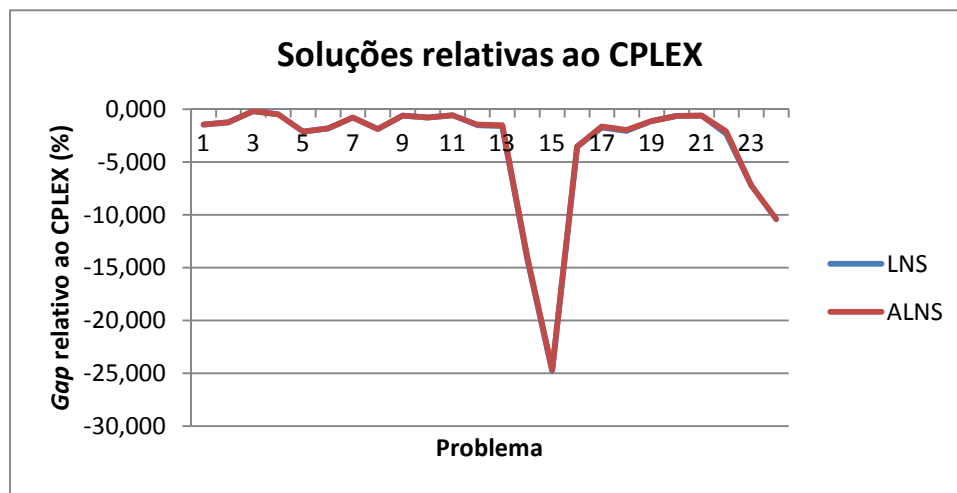


Figura 5.12 - Soluções relativas ao CPLEX da rede de 1177 pontos.

E como se pode ver na Figura 5.13, o LNS e o ALNS foram superiores não só em qualidade da solução como também em tempo computacional, visto que o CPLEX atingiu o máximo limitado para o seu processamento, muito superior ao tempo obtido pelos outros métodos e não atingiu o ótimo ou a melhor solução encontrada para nenhuma instância tratada.

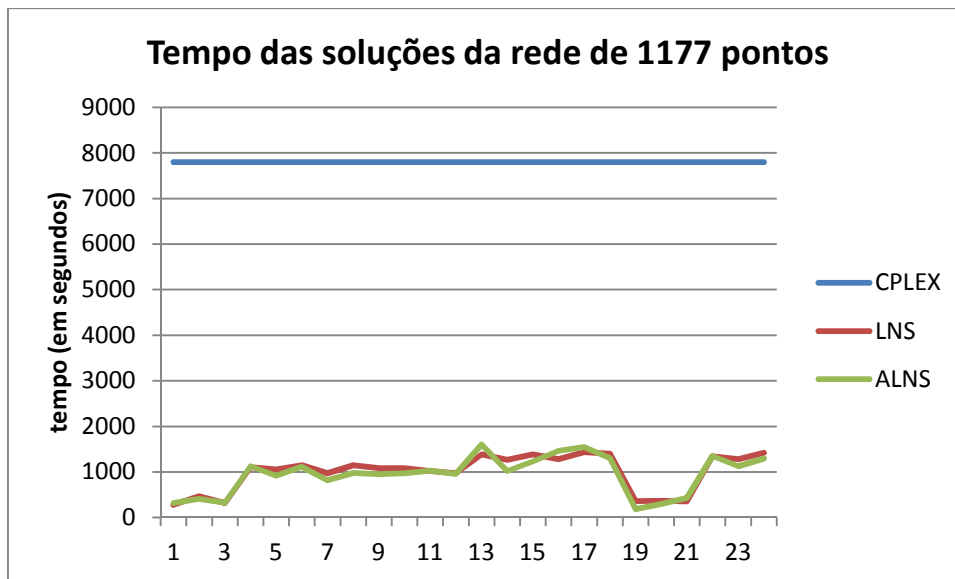


Figura 5.13 - Tempo computacional das soluções da rede de 1177 pontos.

Para as redes maiores, nota-se que o ALNS obtém, em média, *gaps* inferiores ao LNS em tempos computacionais menores. De modo geral, os métodos tem desempenho semelhante para todas as redes.

## 6 CONCLUSÃO

Este trabalho apresentou um método híbrido para a resolução do Problema Probabilístico de Localização-Alocação de Máxima Cobertura utilizando as meta-heurísticas *Large Neighborhood Search* e *Adaptive Large Neighborhood Search*. O método híbrido buscou explorar a formulação matemática do problema, considerando que as decisões de localização e alocação sejam tratadas de forma hierárquica e independente. Funções de destruição e reparo da solução de localização permitiram explorar o espaço de soluções de forma sistemática, permitindo considerar soluções de qualidade inferior, num algoritmo baseado em *Simulated Annealing*, visando escapar de ótimos locais. A solução de localização obtida é então utilizada para determinar os valores ótimos das decisões de alocação, utilizando o *software* comercial de otimização CPLEX, num processo iterativo de acordo com parâmetros definidos.

Os testes computacionais realizados com instâncias da literatura e dados reais demonstraram que, para grande parte das instâncias, o LNS e o ALNS foram capazes de encontrar soluções ótimas ou de boa qualidade em tempo reduzido, o que valida a aplicação deste método ao QM-CLAM.

Observa-se que, quanto maior é a instância, maior é a dificuldade para a obtenção de soluções por métodos exatos. Nestes casos, as meta-heurísticas apresentam desempenho combinando rapidez e qualidade da solução.

Os resultados obtidos com tais abordagens permitem concluir a consistência dos métodos, sendo competitivos para a resolução deste problema, em tempos computacionais razoáveis com convergência rápida.

### 6.1 Trabalhos futuros

O estudo iniciado neste trabalho contém aspectos que podem ser estudados futuramente, de modo a complementar o presente trabalho, através de uma investigação mais aprofundada nos seguintes itens:

- Considerar instâncias maiores deste modelo a fim de confirmar a superioridade das metaheurísticas em relação ao método exato como foi verificado na maior instância testada.

- Aplicar a abordagem ao problema com  $m$  centros de serviços e demais problemas de localização de facilidades.
- Estudar a influência de iniciar o algoritmo com uma boa solução conhecida.
- Investigar métodos que possam auxiliar na confirmação de que a solução encontrada pela meta-heurística é ótima e/ou suficientemente boa – ou seja, encontrar um limitante superior para os resultados.
- Implementar novas funções de destruição e reparo que explorem melhor as características do problema.
- Desenvolver heurísticas para o cálculo das soluções de alocação, em substituição ao método exato.
- Desenvolver métodos de busca local para melhorar a qualidade das soluções.

## REFERÊNCIAS BIBLIOGRÁFICAS

- ARAKAKI, R. G. I. **Heurística de localização-alocação para problemas de localização de facilidades**. 2002. 79 p. (INPE-9572-TDI/837). Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2002. Disponível em: <<http://urlib.net/sid.inpe.br/jeferson/2004/03.08.16.47>>. Acesso em: 27 maio 2014.
- BERMAN, O.; KRASS, D. Facility location problems with stochastic demands and congestion. In: DREZNER, Z.; HAMACHER, H. W. (Ed.). **Facility location: application and theory**. Springer, p.329–371, 2004.
- BERMAN, O.; MANDOWSKY, R. R. Location-allocation on congested networks. **European Journal of Operational Research**, v.26, n.2, p.238–250, 1986.
- BERMAN, O. et al. The stochastic queue p-median location problem. **Transportation Science**, v.21, n.3, p.207-216, 1987.
- BERMAN, et al. Optimal server location on a network operating as an M/G/1 queue. **Operations Research**, v.33, n.4, p.746–771, 1985.
- CHAVES, A. A. **Uma meta-heurística híbrida com busca por agrupamentos aplicada a problemas de otimização combinatória**. Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2009. 199f. Disponível em: <<http://urlib.net/8JMKD3MGP8W/34NDEC8>>. Acesso em: 12 jun. 2014.
- CHURCH, R.; REVELLE, C. The maximal covering location problem. **Papers of the Regional Science Association**, v.32, n.1, p.101 - 118, 1974.
- CORNUÉJOLS, G., THIZY, J. M. Some facets of the simple plant location polytope. **Mathematical Programming**, v.23, n.1, p. 50-74, 1982.
- CORRÊA, F. A. **Relaxações e método de decomposição para alguns problemas de localização de facilidades modelados em grafos**. 2008. 137 p. (INPE-15388-TDI/1405). Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2008. Disponível em: <<http://urlib.net/8JMKD3MGP8W/33NJ938>>. Acesso em: 12 jun. 2014.
- CORRÊA, F. A. et al. Heurística híbrida com detecção de regiões promissoras aplicada ao problema probabilístico de localização-alocação de máxima cobertura. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL (SBPO), 39., 2007, Fortaleza. **Anais...** Fortaleza: UECE/ UFC/ UNIFOR, 2007.
- CORRÊA, F. A.; LORENA, L. A. N. Aplicação da relaxação Lagrangeana e do algoritmo genético construtivo na solução do problema probabilístico de localização-alocação de máxima cobertura. **Revista Gestão & Produção**, v.13, n.2, p. 233-244, 2006.
- CORRÊA, F. A. et al. A decomposition approach for the probabilistic maximal covering location-allocation problem. **Computers & Operations Research**, v.36, n.10, p.2729-2739, 2009.

- DASKIN, M. **Network and discrete location: models, algorithms and applications**. New York: Wiley Interscience, 1995. 500p.
- GOLDBARG, M. C. **Otimização combinatória e programação linear**. Rio de Janeiro: Editora Campus, 2000. 518p.
- HEMMELMAYR, V. C. et al. An adaptive large neighborhood search heuristic for two-echelon vehicle routing problems arising in city logistics. **Computers & Operations Research**, v.39, n.12, p.3215 - 3228, 2012.
- HILLIER, F. S.; LIEBERMAN, G. J. **Introdução à pesquisa operacional**. Porto Alegre: Bookman, 2010, ISBN: 8586804681.
- LORENA, L. A. N. et al. Integração de modelos de localização a sistemas de informações geográficas. **Gestão & Produção**, v.8, n.2, p.180-195, 2001.
- MARIANOV, V.; REVELLE, C. The queuing probabilistic location set covering problem and some extensions. **Socio-Economic Planning Sciences**, v.28, n.3, p.167–178, 1994.
- MARIANOV, V.; SERRA, D. Probabilistic maximal covering location-allocation models for congested systems. **Journal of Regional Science**, v.38, n.3, p. 401-424, 1998.
- MOGHADAS, F. M.; KAKHKI, H. T. Queueing maximal covering location-allocation problem: an extension with M/G/1 queueing systems. **Hindawi Publishing Corporation - Advances in Decision Sciences**. v. 2011, n. ID 605629, 13 p, 2011. Available at: <http://dx.doi.org/10.1155/2011/605629>. Access in: 2014 June 13.
- MOREIRA, D. A. **Pesquisa operacional - curso introdutório**. São Paulo: Thomson Learning, 2007. 356p. ISBN 8522103798.
- MURRAY, A. L.; GERRARD, R. A. Capacitated service and regional constraints in location-allocation modeling. **Location Science**, v.5, n.2, p.113-118, 1997.
- PEREIRA, M. A. **Um método branch-and-price para problemas de localização de p-mediana**. 2005. 94 p. (INPE-14471-TDI/1152). Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2005. Disponível em: <http://urlib.net/sid.inpe.br/jeferson/2005/06.02.11.37>>. Acesso em: 12 jun. 2014.
- PEREIRA, M. A.; LORENA, L. A. N. A heurística Lagrangeana/surrogate aplicada ao problema de localização de máxima cobertura. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL (SBPO), 33., 2001, Campos do Jordão. **Anais...** Campos do Jordão: SBPO, 2001. p.1326-1337.
- PISINGER, D.; ROPKE, S. A general heuristic for vehicle routing problems. **Computers & Operations Research**, v.34, n.8, p.2403–2435, 2007.
- PISINGER, D.; ROPKE, S. Large neighborhood search. In: GENDREAU, M.; POTVIN, J. Y. (Ed.). **Handbook of metaheuristics**. 2.ed. Forthcoming: [s.n], 2009.
- PONTIN, V. M et al. Análise de modelos matemáticos para o problema probabilístico de localização-alocação de máxima cobertura. **Cadernos do IME – Série Estatística**, v.28, p.1-14, 2010.

PRADO, D. F. M. **Busca tabu aplicada ao problema de localização de facilidades com restrições de capacidade e fonte única.** 2007. 101p. Dissertação (Mestrado em Engenharia Elétrica) - Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, 2007.

RIBEIRO, G. M.; LORENA L. A. N. Lagrangean relaxation with clusters and column generation the manufacture's pallet loading problem. **Computers & Operations Research**, v.34, n.9, p.2695-2708, 2007.

ROPKE, S; PISINGER, D. A unified heuristic for a large class of vehicle routing problems with backhauls. **European Journal of Operational Research**, v.171, n.3, p.750-755, 2006(a).

ROPKE, S.; PISINGER, D. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. **Transportation Science**, v.40, n.4, p.455-472, 2006(b)

SHAW, P. Using constraint programming and local search methods to solve vehicle routing problems. In: **CP-98 (Fourth international conference on principles and practice of constraint programming)**. Lecture Notes in Computer Science, v.1520, p.417-431, 1998.

SILVA, F.; SERRA, D. Locating emergency services with different priorities: the priority queuing covering location problem. **Journal of the Operational Research Society**, v.59, n.9, p.1229–1238, 2008.

TAHA, H. A. **Pesquisa operacional.** São Paulo: Pearson Prentice Hall, 2008. 359p. ISBN: 9788576051503.