

**MARCOS MIURA**

**MODELAGEM HEURÍSTICA NO PROBLEMA DE DISTRIBUIÇÃO  
DE CARGAS FRACIONADAS DE CIMENTO**

Dissertação apresentada à Escola  
Politécnica da Universidade de São  
Paulo para obtenção do título de  
Mestre em Engenharia.

SÃO PAULO

2008

**MARCOS MIURA**

**MODELAGEM HEURÍSTICA NO PROBLEMA DE DISTRIBUIÇÃO  
DE CARGAS FRACIONADAS DE CIMENTO**

Dissertação apresentada à Escola  
Politécnica da Universidade de São  
Paulo para obtenção do título de  
Mestre em Engenharia.

Área de Concentração:  
Engenharia de Sistemas Logísticos

Orientador:  
Prof. Dr. Cláudio Barbieri da Cunha

SÃO PAULO

2008

Este exemplar foi revisado e alterado em relação à versão original, sob responsabilidade única do autor e com anuênciâa de seu orientador.

São Paulo, 08 de outubro de 2008.

Assinatura do autor

Assinatura do orientador

## FICHA CATALOGRÁFICA

**Miura, Marcos**

**Modelagem heurística no problema de distribuição de cargas fracionadas de cimento / M. Miura. -- São Paulo, 2008.**

**84 p.**

**Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Transportes. Sistemas Logísticos.**

**1. Transportes 2. Roteirização 3. Heurística 4. Algoritmos genéticos I. Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Transportes II.t.**

## **AGRADECIMENTOS**

Ao Prof. Cláudio Barbieri da Cunha pela orientação e contribuição no desenvolvimento deste trabalho.

Aos meus pais e toda minha família pelo apoio.

À minha futura esposa Daniela.

A todos os meus amigos.

## **RESUMO**

Esta dissertação trata do problema do agrupamento de cargas fracionadas na distribuição de cimento ensacado partindo de um depósito central. O problema consiste em definir quais entregas de cimento serão carregadas juntas em um determinado veículo, de modo a aproveitar ao máximo sua capacidade e ao mesmo tempo reduzir o custo com o frete pago aos transportadores que farão sua distribuição.

Em especial, o método de resolução proposto pode ser dividido em três fases. Na primeira fase, as entregas pertencentes a um mesmo cliente são agrupadas prioritariamente. Na segunda fase, são agrupadas as entregas de clientes dentro de uma mesma cidade. Neste caso, uma simplificação necessária é considerar que todas as entregas de uma mesma cidade estão localizadas em um único ponto. Com isso, a distância entre os clientes se torna irrelevante e é proposto um método baseado em um algoritmo genético para resolução de problemas de “*bin-packing*” (*BPP*). Para a terceira fase, é considerado o agrupamento para pontos de entrega pertencentes a cidades diferentes, onde as distâncias rodoviárias são consideradas. Nesta etapa, é proposta uma variação do método anterior, incorporando ao modelo algumas heurísticas para resolução de problemas de roteirização de veículos, como o algoritmo de Clarke & Wright e o algoritmo do Vizinho Mais Próximo.

## ABSTRACT

This thesis deals with the problem of merging less-than-truckload deliveries in bagged cement distribution from a central depot. The problem consists in defining which cement deliveries shall be loaded in each given vehicle, in order to maximize the vehicle full capacity as well as reduce carriers' freights.

Particularly, the solution method can be divided hierarchically in three stages. In the first stage, the deliveries from the same client are merged with priority. In the second stage, the deliveries from the same city are merged. In this case, a necessary assumption is to consider the deliveries from the same city as located in a single destination point. Consequently, the distances among deliveries can be assumed as irrelevant and a heuristic method is proposed, which relies on a genetic algorithm for the “*bin-packing*” problem (*BPP*). In the third stage, merging of different delivery points that are apart from each other is considered. For this step, a variation of the previous method is proposed, incorporating some heuristics to solve the vehicle routing problem, like the Clarke & Wright’s savings algorithm and the Nearest Neighbor algorithm.

# SUMÁRIO

<b>Capítulo 1 - Introdução .....</b>	<b>1</b>
1.1 - Apresentação do cenário .....	1
1.2 - Importância.....	3
1.3 - Delineamento do trabalho.....	5
<b>Capítulo 2 - Revisão Bibliográfica.....</b>	<b>6</b>
2.1 - Introdução.....	6
2.2 - O problema de roteirização de veículos .....	6
2.3 - Classificação dos problemas de roteirização de veículos.....	9
2.4 - Problema clássico de roteirização de veículos.....	11
2.4.1 - Heurísticas para resolução do <i>VRP</i> clássico .....	12
2.5 - Variações do problema de roteirização de veículos.....	21
2.5.1 - Problema de roteirização de veículos com frota heterogênea.....	22
2.5.2 - Problema de roteirização de veículos com entregas fracionadas.....	23
2.5.3 - Problema de dimensionamento e roteirização veículos .....	23
2.6 - Trabalhos desenvolvidos no âmbito da EPUSP.....	26
2.7 - Problema de bin-packing.....	30
2.7.1 - Problema da Mochila Binária .....	30
2.7.2 - Heurísticas clássicas para resolução do <i>BPP</i> .....	32
2.7.3 - Trabalhos na literatura sobre <i>BPP</i> .....	36
2.8 - Considerações finais.....	38
<b>Capítulo 3 - Caracterização do Problema .....</b>	<b>39</b>
3.1 - Introdução.....	39
3.2 - Descrição do problema.....	39
3.3 - Formulação matemática .....	42
3.4 - Classificação do problema.....	44
<b>Capítulo 4 - Estratégia de Solução .....</b>	<b>46</b>
4.1 - Introdução.....	46
4.2 - Estratégia baseada em algoritmos genéticos para o problema de agrupamento de cargas.....	48
4.2.1 - Representação .....	51
4.2.2 - População inicial .....	51

4.2.3 - Aptidão .....	52
4.2.4 - Seleção.....	53
4.2.5 - <i>Crossover</i> .....	54
4.2.6 - Mutação .....	55
4.2.7 - Reparo.....	55
4.3 - Fase I: Agrupamento de cargas de um mesmo cliente.....	58
4.4 - Fase II: Agrupamento de cargas de uma mesma cidade .....	61
4.5 - Fases III: Agrupamento de cargas de cidades diferentes .....	63
4.5.1 - População inicial para a Fase III .....	64
4.5.2 - Aptidão para a Fase III .....	64
4.5.3 - Exemplo simplificado.....	65
<b>Capítulo 5 - Experimentos Computacionais .....</b>	<b>67</b>
5.1 - Avaliação das Fases I e II.....	67
5.2 - Comparação com o procedimento atual .....	70
5.3 - Comparação com método exato.....	71
<b>Capítulo 6 - Conclusões e Recomendações.....</b>	<b>73</b>
<b>Referências Bibliográficas .....</b>	<b>75</b>
<b>ANEXO A – O algoritmo do modelo Pégasus.....</b>	<b>82</b>

## LISTA DE FIGURAS

Figura 2.1 - Estratégias para solução de <i>PRV</i> (Fonte: Goldbarg e Luna (2000)).....	14
Figura 2.2 - Três nós percorridos por 2 veículos.....	16
Figura 2.3 - Três nós percorridos por um veículo .....	17
Figura 2.4 - Algoritmo para a versão paralela de Clarke & Wright .....	19
Figura 2.5 - Algoritmo para a versão seqüencial de Clarke & Wright .....	20
Figura 2.6 - Heurística <i>Next-Fit</i> .....	33
Figura 2.7 - Heurística <i>First-Fit</i> .....	34
Figura 2.8 - Heurística <i>Best-Fit</i> .....	35
Figura 3.1 - Exemplo de cálculo de tarifa para agrupamento de entregas de 4 cidades .	41
Figura 3.2 - O problema de agrupamento de cargas de cimento .....	42
Figura 4.1 - As três fases do modelo.....	47
Figura 4.2 - Fluxograma geral do método heurístico proposto .....	48
Figura 2.9 - Esquema geral de funcionamento do Algoritmo Genético (Fonte: Cunha, 2006).....	50
Figura 4.3 - Representação do cromossomo no problema estudado .....	51
Figura 4.4 - Indivíduos da população inicial.....	51
Figura 4.5 - Percentuais de cada indivíduo na seleção .....	54
Figura 4.6 - Crossover.....	54
Figura 4.7 – Mutação .....	55
Figura 4.8 - Algoritmo para procedimento Reparo .....	56
Figura 4.9 - Exemplo da solução após mutação .....	57
Figura 4.10 - Exemplo processo de reparo do cromossomo .....	57
Figura 4.11 - Esquema geral do algoritmo genético para resolução do problema de <i>bin-packing</i> .....	58
Figura 4.12 - Fluxograma da Primeira Fase.....	59
Figura 4.13 - Abordagem para caminhões de diferentes capacidades .....	62
Figura 4.14 - Indivíduos da população inicial da Fase III.....	64

## LISTA DE TABELAS

Tabela 2.1 - Taxonomia para problemas de roteirização de veículos.....	8
Tabela 2.2 - Heurísticas de economias.....	26
Tabela 2.3 - Razões de performance do pior caso para algoritmos de <i>bin-packing</i> .....	36
Tabela 3.1 - Exemplo de dados de entrada para o problema.....	39
Tabela 3.2 - Capacidade máxima dos veículos considerados .....	40
Tabela 3.3 - Taxonomia para o problema proposto.....	44
Tabela 4.1 - Seleção através da roleta.....	53
Tabela 4.2 - Exemplo simplificado da Fase I.....	60
Tabela 4.3 - Resultados após Fase I para exemplo simplificado.....	60
Tabela 4.4 - Exemplo simplificado da Fase II.....	63
Tabela 4.5 - Resultados após Fase II .....	63
Tabela 4.6 - Exemplo simplificado da Fase III .....	65
Tabela 4.7 - Resultados após Fase III .....	65
Tabela 5.1 - Parâmetros do AG .....	67
Tabela 5.2 - Resultados do algoritmo para resolução do <i>BPP</i> (120 itens).....	68
Tabela 5.3 - Resultados do algoritmo para resolução do <i>BPP</i> (250 itens).....	69
Tabela 5.4 - Resultados do método proposto comparados ao método atual .....	71
Tabela 5.5 – Resultados do método proposto comparados aos do método exato .....	72

# Capítulo 1 - Introdução

O presente trabalho estuda a implementação de métodos heurísticos, normalmente utilizados na resolução de problemas de *bin-packing*, aplicados na resolução de um problema de dimensionamento e roteirização de uma frota heterogênea de veículos (*Fleet Size and Mix Vehicle Routing Problem – FSMVRP*). O problema consiste em definir quais entregas de cimento ensacado são carregadas juntas em um determinado veículo, de modo a aproveitar ao máximo sua capacidade e ao mesmo tempo reduzir o custo com o frete pago aos transportadores que farão sua distribuição.

Este capítulo apresenta primeiramente o cenário em que o problema é tratado. Em seguida, apresenta-se a importância e justificativa do tema. Por fim, é definido o delineamento do trabalho.

## 1.1 - Apresentação do cenário

O mercado de cimento atual no Brasil é muito competitivo. A presença de vários competidores em busca de uma fatia do mercado faz com que os preços baixem, e com isso o lucro operacional das empresas deste segmento também. Desta forma, surge uma necessidade incessante de aumentar a margem comercial das vendas, seja através da redução dos custos, onde a logística possui grande representatividade, seja na busca de canais de distribuição mais rentáveis, além de outros fatores. Com isso, uma tendência de algumas das empresas desse segmento passou a ser o foco em clientes menores, no varejo, que podem proporcionar uma margem de venda melhor. Por outro lado, o aumento da participação de clientes de pequeno porte dificulta a atividade logística de distribuição, devido à pulverização geográfica dos mesmos. Além disso, a diminuição do tamanho do pedido médio feito por cada cliente faz com que o trabalho de agrupamento das entregas nos veículos se torne fundamental para a redução dos custos.

O cimento ensacado não necessita de um acondicionamento especial para seu transporte, podendo ser transportado em caminhões para carga geral. Associado a isso, o fato do

cimento ser um produto de valor agregado relativamente baixo, faz com que muitas empresas no setor busquem por transportes mais simples e menos custosos, utilizando para essa atividade, transportadores autônomos ou pequenas empresas transportadoras.

Nesse contexto, o presente trabalho aborda o problema da distribuição de cimento, que compreende a otimização do agrupamento das entregas nos veículos, de forma a reduzir a frota necessária e o custo com o frete pago aos transportadores.

A distribuição de cimento normalmente não se restringe a uma cidade ou região metropolitana. Geralmente são atendidos vários municípios a partir de uma base de origem, de diferentes tamanhos, cada qual com diversos pontos de entrega. Normalmente, a distância de deslocamento desde o local de origem onde é carregado o veículo até um município onde são realizadas as entregas é de magnitude muito superior às distâncias de percurso entre entregas consecutivas. Tendo em vista o mercado abrangido, as características do produto a ser entregue, os tempos envolvidos (de entrega e de deslocamento do veículo), o número de entregas por veículo é normalmente baixo, inferior a 10 e na média em torno de 5 a 6. Conseqüentemente, torna-se em geral necessário despachar mais de um veículo para as entregas em um município ou em municípios vizinhos, próximos entre si. A forma de contratação do transporte, e de remuneração dos transportadores autônomos ou pequenas empresas transportadoras, favorece esse agrupamento por município, uma vez que um custo adicional é incorrido quando as entregas ocorrem em municípios distintos, levando à necessidade do veículo se deslocar uma distância maior para completar as entregas. Além da distância adicional a ser percorrida, entregas em municípios distintos, ou locais distantes entre si, leva a uma diminuição da produtividade dos veículos, tendo em vista o menor tempo disponível para as entregas, que devem ocorrer dentro das janelas de recebimento dos clientes, normalmente limitadas ao horário comercial de funcionamento desses estabelecimentos.

Tudo isso, associado à não existência de mapas digitais para a grande maioria dos municípios onde devem ser realizadas entregas, com exceção daqueles de maior porte, leva a uma programação da distribuição onde a localização dos destinos é considerada em nível de município. Este tipo de consideração é muito conveniente para a situação real, visto que, a cada dia, surgem novos clientes, de modo que, mapear a localização

exata de cada um é inviável. Desta forma, são formados *clusters* de clientes em macro regiões, onde para cada cliente novo cadastrado na base de dados, define-se este ponto de localização chamado de “zona de transporte”, que pode representar sua cidade ou bairro (no caso de cidades grandes). Com isso, podemos ter muitas entregas para um mesmo ponto de destino.

Nesse contexto propõe-se uma estratégia de solução buscando otimizar a programação da distribuição diária de cimento aos pontos de entrega, que compreende a definição, para cada município ou zona de transporte, do número de veículos de cada tipo que são necessários, e dos pontos de entrega que são alocados a cada veículo, buscando minimizar a despesa total com frete. Como são normalmente reduzidas as distâncias entre pontos de entrega alocados a um mesmo veículo, diminui a necessidade de definição do roteiro ou seqüência de entregas alocadas ao veículo, o que simplifica o problema e permite considerar uma abordagem de solução heurística, inspirada em uma generalização do chamado problema de *bin-packing* (*BPP*). Para os pontos de entrega remanescentes, isto é, todos aqueles que não permitiram completar a carga de algum dos veículos alocados a cada um dos municípios ou zonas de transporte, é então resolvido um problema de roteirização, que consiste em agrupar e roteirizar esses conjuntos de pontos, de forma a minimizar o custo de transporte. Para tanto, são incorporadas ao modelo algumas heurísticas tradicionais para resolução do problema de roteirização de veículos.

A estratégia de solução proposta é aplicada a um problema real de uma grande empresa do setor de cimentos, comparando-se os resultados obtidos com a solução praticada pela empresa.

## 1.2 - Importância

O planejamento logístico decide de uma maneira geral ou estratégica como um produto se move através de canais de distribuição e fornecimento. Especificar a configuração da rede de instalações, fixar políticas de estoque, ou definir um modo de transportar os bens são decisões estratégicas típicas dentro da logística. O planejador logístico freqüentemente se torna envolvido com uma variedade de decisões sobre alocação de

recursos e seqüenciamento dos processos, ambos de uma natureza operacional e estratégica. Dentre estas, destacam-se as decisões de transportes.

A roteirização de veículos se destaca, então, como uma maneira de visitar uma série de clientes ao menor custo (frete) possível, atendendo a todas as demais imposições do problema. A resolução deste tipo de problema pode gerar benefícios muito grandes em termos de redução de custo, como também redução do tempo de transporte. Além disso, em qualquer ramo de negócios em que a atividade de transporte estiver inserida, isto pode provocar um ganho significativo em produtividade operacional, uma melhoria no sistema de distribuição de produtos, e um aumento da satisfação dos clientes.

A maioria dos trabalhos na literatura para problemas de roteirização de veículos não leva em conta a dificuldade da aplicação do modelo teórico à prática, principalmente quando o intuito é torná-lo uma ferramenta operacional e de boa utilidade. Isto advém do fato de que a obtenção de alguns detalhes exigidos pelo modelo é na realidade trabalhosa e inviável. Por exemplo, a grande maioria das empresas não é capaz de definir as coordenadas geodésicas de cada cliente novo que é cadastrado em sua base de dados, bem como de definir as distâncias deste novo ponto para todos os outros clientes e depósitos.

Dentro deste contexto, o presente trabalho visa adaptar um modelo teórico de modo a atender essas dificuldades e servir como uma ferramenta útil operacionalmente. Por outro lado, esta estratégia de agrupar ou “*clusterizar*” os pontos de entregas em macro-regiões não resulta em uma simplificação ou relaxação do problema clássico que afete a qualidade dos dados ou o objetivo do problema proposto. Isto porque, como é abordado mais adiante no detalhamento do problema, a idéia principal é reduzir o custo com o frete pago aos transportadores subcontratados, cujas tarifas estão vinculadas a estas macro-regiões.

Outro destaque é a aplicação de métodos de resolução de problemas de *bin-packing* em problemas de roteirização de veículos, visto que poucos trabalhos foram desenvolvidos na literatura abordando este tema em específico.

### **1.3 - Delineamento do trabalho**

O trabalho está organizado em cinco capítulos, cada qual com o conteúdo mostrado a seguir.

O capítulo 2 corresponde à revisão bibliográfica. Neste capítulo, são apresentados os conceitos teóricos que envolvem cada etapa do modelo. Além disso, são apresentados alguns resultados de pesquisa feitos por outros autores sobre o tema relacionado e que auxiliaram na preparação do presente trabalho.

No capítulo seguinte, o problema é descrito em detalhes, ressaltando-se os aspectos a serem adotados em sua modelagem.

O capítulo 4 descreve a modelagem e a estratégia de solução. Neste capítulo, a descrição do algoritmo é aprofundada em cada uma de suas etapas. Além disso, são apresentadas passo a passo todas as metodologias utilizadas.

No capítulo 5 são apresentados os experimentos computacionais, mostrando alguns resultados obtidos com o modelo e a análise na comparação com outras metodologias.

Finalmente, o último capítulo é dedicado à conclusão do trabalho e recomendações, onde são comentados os resultados finais do modelo proposto e algumas sugestões de melhoria para estudos futuros.

## Capítulo 2 - Revisão Bibliográfica

### 2.1 - Introdução

O presente trabalho engloba dois tipos de problemas clássicos da literatura: o problema de roteirização de veículos e o problema de *bin-packing*. Neste capítulo, são discutidos detalhadamente cada um dos temas e seus principais métodos de resolução, que serviram de base para a construção do modelo final.

Nas seções seguintes, é apresentado o conceito de problemas de roteirização de veículos, passando pela classificação para estes tipos de problema. Em seguida, são revisados o problema clássico de roteirização de veículos (*VRP*) e algumas de suas variações: o problema de roteirização de veículos com frota heterogênea (*HFVRP*), o problema de roteirização com entregas fracionadas (*VRPSD*), e por fim, o problema de dimensionamento e roteirização de veículos (*FSMVRP*). Na seção 2.7, é apresentado o problema do *bin-packing* e alguns trabalhos da literatura relacionando-o com problemas de roteirização de veículos.

### 2.2 - O problema de roteirização de veículos

Para entender os problemas de roteirização de veículos faz-se necessária uma definição sobre os sistemas de roteirização. Considera-se um sistema de roteirização um conjunto organizado de meios com o objetivo de atender pontos de demanda localizados em arcos ou vértices de alguma rede de transportes (Goldbarg e Luna, 2000). Os problemas de roteirização de veículos pertencem a uma categoria ampla de problemas de pesquisa operacional conhecida como problemas de otimização de rede. Dentro desta categoria estão os problemas clássicos, como o problema do fluxo máximo, problema do caminho mínimo, problema de transporte, problema de designação, dentre outros (Golden, Ball, Bodin, 1981).

Um dos primeiros problemas de roteirização de veículos apresentado na literatura é o problema do caixeiro viajante (*traveling salesman problem – TSP*). O problema consiste em encontrar o roteiro ou seqüência de cidades a serem visitadas por um caixeiro viajante que minimize a distância total percorrida e assegure que cada cidade seja visitada exatamente uma vez. Neste problema não há restrições de capacidade de veículos, e a demanda é determinística.

Desde então, novas restrições vêm sendo incorporadas ao problema do caixeiro viajante, de modo a melhor representar os diferentes tipos de problemas que envolvem roteiros de pessoas e veículos. Dentre estas restrições destacam-se: restrições de horário de atendimento, capacidades dos veículos, frota composta de veículos de diferentes tamanhos, duração máxima dos roteiros dos veículos e restrições de tipos de veículos que podem atender determinados clientes (Cunha, 2000). Christofides (1985) define o problema de roteirização de veículos como um problema de distribuição onde os clientes espalhados geograficamente devem ter suas demandas atendidas por veículos partindo de um depósito central. Ainda segundo Goldbarg e Luna (2000), a idéia básica do problema de roteirização é, com o uso de veículos, visitar uma série de clientes ao menor custo possível, atendendo a todas as demais imposições do problema.

Uma das principais dificuldades em se modelar e resolver um problema de roteirização de veículos advém da grande quantidade de parâmetros que podem influenciar o tipo de problema. A adequada classificação dos problemas de roteirização permite uma melhor compreensão dos aspectos mais relevantes. Portanto, é fundamental identificar, modelar e classificar corretamente o problema, determinando suas características mais relevantes e sua relação com outros problemas, para que seja proposta uma estratégia de solução adequada. Um dos primeiros trabalhos mais abrangentes de classificação dos problemas de roteirização e programação de veículos é apresentado em Bodin *et al.* (1983). Nele estão expostas revisões dos principais algoritmos, estratégias de solução e aplicações conhecidas até então, abrangendo cerca de 700 referências bibliográficas. Alguns autores definem alguns parâmetros para taxonomia dos problemas de roteirização de veículos. A taxonomia a seguir (Tabela 2.1) foi proposta por Belfiore (2005) e reúne trabalhos de Christofides (1985), Bodin e Golden (1981), Bodin *et al.* (1983), Assad (1988), Ronen (1988) e Brejon (1998).

**Tabela 2.1 - Taxonomia para problemas de roteirização de veículos**

<b>Parâmetro</b>	<b>Tipos</b>
Função objetivo	<ul style="list-style-type: none"> <li>• Minimizar a distância total percorrida;</li> <li>• Minimizar a duração das rotas;</li> <li>• Minimizar o número de veículos;</li> <li>• Minimizar os custos totais de distribuição.</li> </ul>
Restrições dos veículos	<ul style="list-style-type: none"> <li>• Limite de capacidade dos veículos;</li> <li>• Limite com relação ao tipo de carga dos veículos;</li> <li>• Número e tipo de veículos disponíveis;</li> <li>• Operação de carga e descarga dos veículos.</li> </ul>
Restrições dos clientes	<ul style="list-style-type: none"> <li>• Janelas de tempo dos clientes;</li> <li>• Limite com relação ao tipo de carga ou tipo de veículo;</li> <li>• Clientes prioritários;</li> <li>• Atendimento total ou parcial das demandas;</li> <li>• Tempo máximo permitido para carga e descarga.</li> </ul>
Restrições de rotas	<ul style="list-style-type: none"> <li>• Horários de início e término das viagens;</li> <li>• Tempo máximo de viagem de um veículo;</li> <li>• Distância máxima percorrida;</li> <li>• Locais de parada.</li> </ul>
Variáveis de decisão	<ul style="list-style-type: none"> <li>• Roteiro a ser percorrido por cada veículo;</li> <li>• Qual veículo é designado para cada cliente;</li> <li>• Qual a quantidade de carga transportada para cada cliente da rota;</li> <li>• Tempo de início de atendimento do primeiro cliente da rota.</li> </ul>
Tipo de operação	<ul style="list-style-type: none"> <li>• Coleta;</li> <li>• Entrega;</li> <li>• Coleta e entrega simultaneamente.</li> </ul>
Tipo de carga	<ul style="list-style-type: none"> <li>• Única, completa ou carga de lotação;</li> <li>• Múltiplas cargas ou carga fracionada.</li> </ul>
Tipo de demanda	<ul style="list-style-type: none"> <li>• Determinística;</li> <li>• Estocástica.</li> </ul>
Localização da demanda	<ul style="list-style-type: none"> <li>• Demanda localizada somente em arcos;</li> <li>• Demanda localizada somente em nós;</li> <li>• Demanda localizada em arcos e nós.</li> </ul>
Tamanho da frota	<ul style="list-style-type: none"> <li>• Limitada;</li> <li>• Ilimitada.</li> </ul>
Tipo de frota	<ul style="list-style-type: none"> <li>• Homogênea;</li> <li>• Heterogênea.</li> </ul>
Depósitos e localização de veículos	<ul style="list-style-type: none"> <li>• Um único depósito;</li> <li>• Vários depósitos;</li> <li>• Quantidade de produtos disponíveis no depósito central para entrega aos clientes;</li> <li>• Número de bases de origem e destino dos veículos.</li> </ul>
Jornada de trabalho	<ul style="list-style-type: none"> <li>• Duração;</li> <li>• Horário de almoço e outras interrupções;</li> <li>• Permissão para viagens com mais de um dia de duração;</li> <li>• Número de tripulantes por veículo.</li> </ul>
Estrutura da rede	<ul style="list-style-type: none"> <li>• Direcionada;</li> <li>• Não direcionada;</li> <li>• Mista;</li> <li>• Euclidiana.</li> </ul>
Horizonte de planejamento	<ul style="list-style-type: none"> <li>• Curto prazo;</li> <li>• Longo prazo.</li> </ul>

Fonte: Belfiore (2005)

## 2.3 - Classificação dos problemas de roteirização de veículos

Conforme visto anteriormente, uma visão sistêmica dos principais problemas de roteirização de veículos e uma adequada classificação que identifique os aspectos mais relevantes permite que seja implementada uma estratégia de solução adequada. Nesta seção, são descritas algumas classificações para o problema de roteirização de veículos encontrada na literatura.

Bodin *et al.* (1983) apresentam uma classificação para problemas de roteirização de veículos em função de restrições em aspectos espaciais e/ou temporais, dentro de três grupos:

- ***Problema de roteirização pura:*** sem restrições temporais como janelas de tempo de atendimento aos clientes. Consideram-se apenas aspectos espaciais, objetivando construir roteiros viáveis a um menor custo possível.
- ***Problema de programação de veículos:*** há restrições relativas aos horários para atendimento aos clientes e/ou janelas de tempo. Neste tipo de problema consideram-se apenas os aspectos temporais.
- ***Problemas combinados de roteirização e programação de veículos:*** há restrições tanto temporais quanto espaciais. Segundo Bodin e Golden (1981), a maioria dos problemas reais encaixa-se nesta categoria.

Segundo Cunha (2000), a classificação de Bodin *et al.* (1983) é ainda hoje vista como uma das mais importantes, pois considera os principais tipos de problemas de roteirização de veículos.

Outro tipo de classificação interessante é apresentado por Ronen (1988), que utiliza uma classificação em função do ambiente operacional e objetivos do problema:

- ***Problemas relacionados ao transporte de passageiros:*** são os problemas relativos a transportes escolares de ônibus, táxi, entre outros.

- **Problemas de prestação de serviços:** roteirização e programação de serviços como coleta de lixo, entrega postal, varrição de ruas, entre outros.
- **Problemas de roteirização e programação ou transporte de cargas:** roteirização e programação de veículos no transportes de cargas.

Já Assad (1988) sugere que cada problema prático seja caracterizado particularmente, de acordo com o seguinte conjunto de elementos:

- **Natureza das demandas:** somente coletas ou entregas, um ou múltiplos produtos, atendimento total ou parcial da demanda, prioridade no atendimento aos clientes, terceirização ou não, entre outros;
- **Características de demanda:** determinística ou estocástica, constante ou variável, entre outros;
- **Frota de veículos:** homogênea ou heterogênea, restrições de capacidade de veículo, restrições de produto em função do tipo de veículo, frota fixa ou variável, frota em um único depósito ou múltiplos depósitos;
- **Requisitos de pessoal:** duração da jornada de trabalho, inclusão de hora-extra, número fixo ou variável de motoristas, horários e locais de início de jornada, hora de almoço e outros tipos de parada;
- **Requisitos de programação:** restrições de janela de tempo para coleta e entrega, tempos de carga e descarga, horários de abertura e fechamento;
- **Requisitos de informações:** disponibilidade geográfica, tempo de viagem, localização dos veículos.

As seções seguintes apresentam problemas clássicos de roteirização e programação de veículos com base nos trabalhos de Bodin *et al.* (1983) e Solomon e Desrosiers (1988).

## 2.4 - Problema clássico de roteirização de veículos

O problema de roteirização de veículos (*Vehicle Routing Problem - VRP*) tem como objetivo encontrar um conjunto de rotas com o menor custo possível (distância total percorrida, tempo total de viagem, etc.), iniciando e terminando no depósito, de forma que a demanda de todos os nós sejam atendidas. A demanda é determinística. O *VRP* é uma extensão do problema de múltiplos caixeiros viajantes, onde é acrescentada a restrição de capacidade dos veículos.

Uma formulação matemática para o problema de roteirização de veículos foi proposta por Fisher e Jaikumar (1981) e é detalhada a seguir.

### Parâmetros

$N$  = Número total de clientes a serem atendidos em um dia

$M$  = Número total de veículos disponíveis

$c_{ij}$  = Custo de percorrer o caminho entre o cliente  $i$  e cliente  $j$

$Q_k$  = Capacidade do veículo  $K$  (peso)

$q_i$  = Demanda do cliente  $i$

### Variáveis de decisão

$$X_{ijk} = \begin{cases} 1, & \text{se o arco de } i \text{ até } j \text{ é percorrido pelo veículo } k \\ 0, & \text{caso contrário} \end{cases}$$

$$Y_{ik} = \begin{cases} 1, & \text{se o veículo } k \text{ visita o endereço } i \\ 0, & \text{caso contrário} \end{cases}$$

### Formulação do modelo

$$\min z = \sum_{i,j} \left( c_{ij} \cdot \sum_k X_{ijk} \right) \quad (2.1)$$

Sujeito a:

$$\sum_{k=1}^M Y_{ik} = 1, \quad i = 2, \dots, N \quad (2.2)$$

$$\sum_{k=1}^M Y_{ik} = M, \quad i = 1 \quad (2.3)$$

$$\sum_{i=1}^N q_i \cdot Y_{ik} \leq Q_k, \quad k = 1, \dots, M \quad (2.4)$$

$$\sum_{j=1}^N X_{ij} = \sum_{j=1}^N X_{ji} = Y_{ik}, \quad i = 1, \dots, N \quad k = 1, \dots, M \quad (2.5)$$

$$\sum_{i,j \in S} X_{ijk} \leq |S| - 1, \quad \forall S \subseteq \{2, \dots, N\}, \quad k = 1, \dots, M \quad (2.6)$$

Na formulação matemática acima, a função objetivo (2.1) visa minimizar o custo total de viagem. A restrição (2.2) garante que cada cliente seja visitado por apenas um veículo. A restrição (2.3) garante que o ponto de partida ( $i = 1$ ) receba a visita de todos os veículos. Ou seja, todos os veículos devem necessariamente passar pela base. A restrição (2.4) assegura que a capacidade do veículo não seja excedida. A restrição (2.5) garante que os veículos não param suas rotas em um cliente. Ela relaciona as variáveis binárias  $X_{ijk}$  e  $Y_{ik}$ , de modo que se um cliente  $i$  for visitado por um veículo  $k$  ( $Y_{ik} = 1$ ), haverá apenas um arco chegando ( $\sum_{j=1}^N X_{ij} = 1$ ) e um arco saindo ( $\sum_{j=1}^N X_{ji} = 1$ ) deste cliente. Caso contrário, todos os arcos receberão o valor nulo. Por fim, a restrição (2.6) garante que não sejam formadas rotas desconectadas da base ou subrotas.

#### 2.4.1 - Heurísticas para resolução do VRP clássico

Os problemas de roteirização de veículos variam quanto a sua complexidade computacional dependendo do número de variáveis e restrições que o problema considera em sua formulação. Alguns problemas podem ser considerados quanto a sua

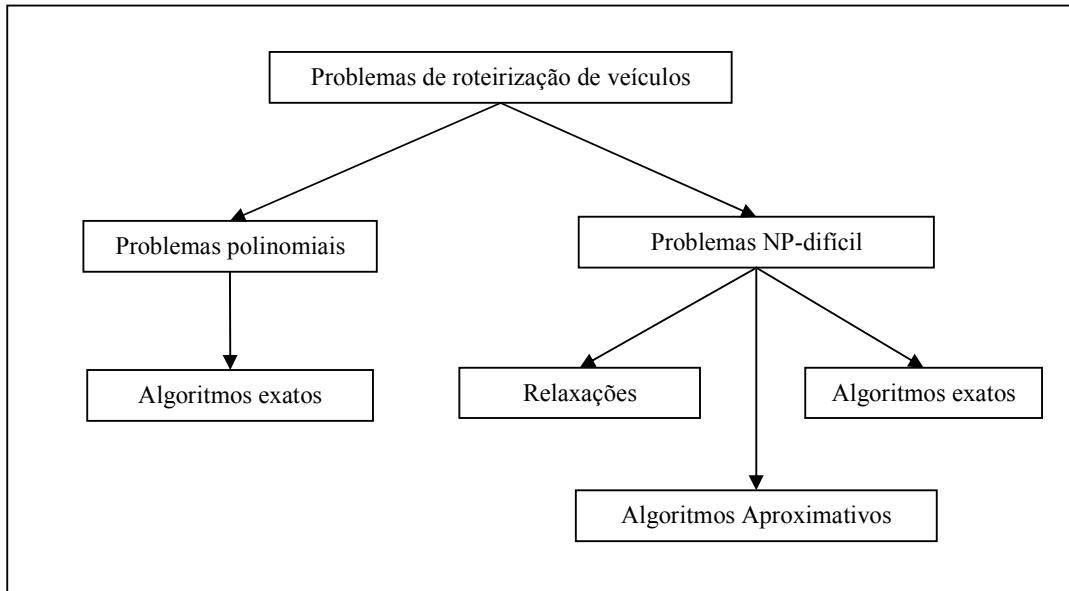
complexidade como “inratáveis” (Bodin *et al.*, 1983). Não se trata somente de aumentar a capacidade da máquina, pois a dificuldade reside na natureza combinatória desse tipo de problema que, até hoje, tem impedido a concepção de algoritmos eficientes de solução.

Esses problemas são conhecidos como *NP-difícil* (do inglês *NP-Hard*). Em outras palavras, o esforço computacional para a sua resolução cresce exponencialmente com o tamanho do problema, dado pelo número de pontos a serem atendidos (Garey e Johnson, 1999).

Para esses problemas complexos, foram desenvolvidos métodos de solução baseados em procedimentos heurísticos, isto é, que não asseguram a obtenção de soluções ótimas do ponto de vista matemático. O termo *heurística* usualmente refere-se a um procedimento de busca iterativo, baseado em alguma estrutura de controle repetitivo (Cunha, 2006).

As estratégias de solução heurísticas apóiam-se, em geral, em alguma abordagem intuitiva, na qual a estrutura particular do problema pode ser considerada e explorada de forma inteligente, a fim de se obter uma solução satisfatória, e considerando-se o compromisso qualidade versus esforço computacional para obtê-la (Silver, 2004).

A Figura 2.1, abaixo, mostra como a pesquisa operacional desenvolveu estratégias para tratamento de problemas de roteirização.



**Figura 2.1 - Estratégias para solução de PRV (Fonte: Goldbarg e Luna (2000))**

A maioria dos problemas de roteirização de veículos é do tipo *NP-difícil* (Goldbarg e Luna, 2000). Como visto na Figura 2.1, este tipo de problema pode ser tratado com algoritmos exatos, relaxações e algoritmos aproximativos. Os algoritmos exatos são geralmente utilizados em problemas menores, pois o esforço computacional demandado neste tipo de resolução é elevado. Para problemas maiores, o mesmo vem acompanhado de técnicas de relaxações, que ajudam a eliminar algumas restrições do problema. Em outros casos, são utilizados algoritmos aproximativos (heurísticas), que buscam de maneira simplificada, mas eficiente, soluções que aproximem ao máximo da solução ótima do problema em um tempo computacional reduzido.

A seguir, são apresentadas duas heurísticas clássicas para resolução do *VRP*: o Algoritmo do Vizinho Mais Próximo e o algoritmo de Clarke e Wright.

### (i) Algoritmo do Vizinho Mais Próximo

O algoritmo do Vizinho Mais Próximo (*NNA*) apresentado por Rosenkrantz *et al.* (1977) é um dos métodos heurísticos mais intuitivos para a resolução do problema de roteirização de veículos. O algoritmo segue um procedimento guloso muito simples. Considere um grafo  $G$  com  $|V| = n$  nós,  $|E| = m$  arestas, completo e simétrico. O

algoritmo do Vizinho Mais Próximo (*NNA*), de modo a determinar um circuito hamiltoniano  $T$  de  $G$ , consiste nos seguintes passos:

- (1) Escolha aleatoriamente um nó  $v \in V(G)$ ;
- (2) Inclua  $v$  no circuito  $T$ ;
- (3) Considere o nó  $u$ , mais próximo do nó  $v$  e ainda não incluído no circuito  $T$ . Inclua o nó  $u$  e a aresta  $(v, u)$  no circuito  $T$  e faça  $v = u$ ;
- (4) Se  $|V(T)| \leq |V(G)|$ , então vá para o PASSO 3;
- (5) Se não, inclua a aresta entre o primeiro e o último dos nós incluídos.

Segundo Bodin *et al.* (1983), o comprimento da solução do algoritmo do Vizinho Mais Próximo é limitada no pior caso por:

$$\text{comprimento do roteiro NNA} \leq \left( \frac{1}{2} [\lg(n)] + \frac{1}{2} \right) \times (\text{comprimento do roteiro ótimo}) \quad (2.7)$$

onde,  $\lg$  significa o logaritmo na base 2,  $[X]$  é o menor inteiro maior ou igual a  $X$ , e  $n$  é o número de nós na rede.

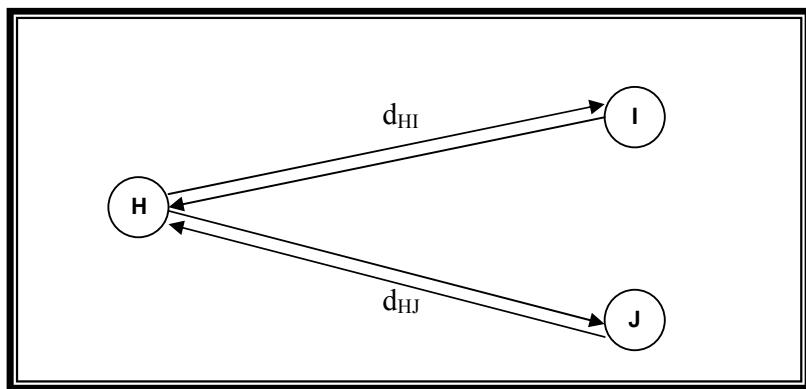
## **(ii) Heurística de Clarke & Wright**

O algoritmo de Clarke & Wright é um método heurístico do tipo *saving* (economia) que busca substituir arcos mais caros dentro da rota por arcos de menor custo, de forma a criar uma rota melhorada. O método a ser apresentado neste capítulo é uma adaptação da formulação de Clarke e Wright (1964).

Segundo Ballou (1985), a utilização deste algoritmo em problemas com um número limitado de restrições pode resultar em soluções próximas a 2% em relação à solução ótima. Por fim, o modelo *C&W* é uma das técnicas mais conhecidas e utilizadas na resolução deste tipo de problema. Além de ser capaz de gerar soluções próximas da ótima, ele é flexível e suficiente para lidar com uma larga gama de restrições, e relativamente rápido para problemas com um número moderado de paradas (Ballou, 1985).

Inicialmente, assuma a existência de  $n$  pontos a serem visitados, partindo o veículo do depósito  $H$  e retornando ao mesmo após um ciclo. De momento, vamos admitir que uma solução (a pior) seja a existência de  $n$  veículos disponíveis para realizar estas viagens. A Figura 2.2 apresenta esta relação para 3 nós (2 clientes), sendo o nó  $H$  representando o depósito e os nós  $I$  e  $J$  os clientes de coleta. A distância total percorrida pelos dois veículos é representada pela equação 2.8.

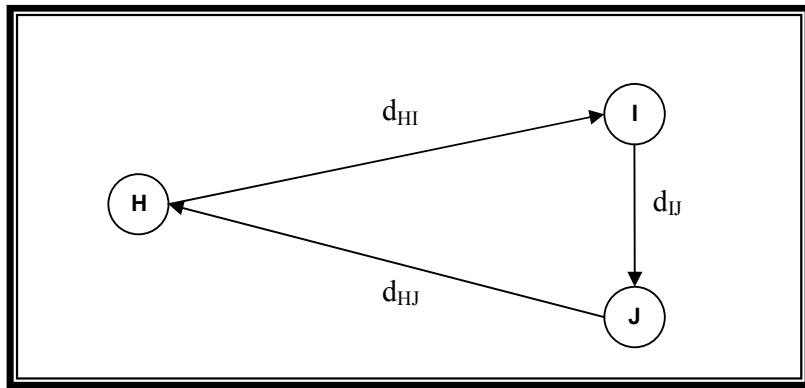
$$D = 2 \cdot (d_{HI} + d_{HJ}) \quad (2.8)$$



**Figura 2.2 - Três nós percorridos por 2 veículos**

Agora, imaginemos que possamos eliminar um veículo do roteiro acima, de forma que apenas um veículo percorra os três nós do problema. Assim, fazendo o veículo percorrer o trecho  $H - I - J$  e retornando a  $H$  no final, há uma economia de distância percorrida, pois ele deixa de viajar um trecho  $IH$  e um trecho  $HJ$ . No entanto, ele deve percorrer um trecho a mais  $IJ$  (ver Figura 2.3). Desta forma, a economia gerada por este novo percurso é representada pela equação 2.9.

$$S_{ij} = d_{HI} + d_{HJ} - d_{IJ} \quad (2.9)$$



**Figura 2.3 - Três nós percorridos por um veículo**

A solução da heurística de Clarke & Wright vem através da computação destas economias. As economias representam o quanto a distância ou o custo podem ser reduzidos, agrupando nós (*I* e *J* no exemplo) e criando a rota *H – I – J – H*, que pode ser destinada a um veículo.

Para uma rede de  $n$  nós, computam-se as economias para cada par de nós, ordenam-se as economias obtidas em ordem decrescente, e é construído um roteiro ligando estes pares até concluir toda rota.

### Descrição do método

Nesta seção, descrevemos passo a passo a formulação da heurística de Clarke & Wright. Note que existem duas versões de algoritmos que podem ser utilizadas: uma versão paralela e uma versão seqüencial. É apresentada a diferença entre as duas versões e definida a versão usada no problema.

São definidas a seguir as restrições básicas do problema, considerando que podem haver alterações dependendo da peculiaridade do problema tratado, por exemplo, a restrição de janelas de tempo. As restrições básicas do problema são:

- Cada rota inicia e termina no depósito;
- Cada cliente é atendido em uma única rota;
- A demanda total do cliente em uma rota não excede a capacidade do veículo;

- O tempo total de um roteiro não excede a duração da jornada de trabalho do motorista;

**Objetivo:** Atender todos os clientes minimizando a distância total percorrida e respeitando todas as restrições impostas pelo problema.

A descrição da heurística segue, passo a passo, a seguir:

**Passo 1:** Estabelecer como solução inicial para  $n$  nós, a formação de  $n$  rotas partindo e chegando ao depósito (esta solução, apesar de teoricamente factível, é a mais custosa).

**Passo 2:** Computar as economias,  $S_{ij}$  ligando todos os nós  $i$  e  $j$ :

$$S_{ij} = d_{Hi} + d_{Hj} - d_{ij} \quad (i, j = 2, 3, \dots, n)$$
, onde  $d$  representa a distância do trecho correspondente aos nós subscritos, sendo o depósito representado por  $H$ .

**Passo 3:** Ordenar as economias obtidas em ordem decrescente, formando uma lista de pares de nós, com suas respectivas economias.

Neste momento, o modelo de Clarke & Wright pode seguir dois caminhos:

1. *Versão Paralela*: efetua a melhor união factível;
2. *Versão Seqüencial*: une maior número de nós em um roteiro.

### **Versão Paralela**

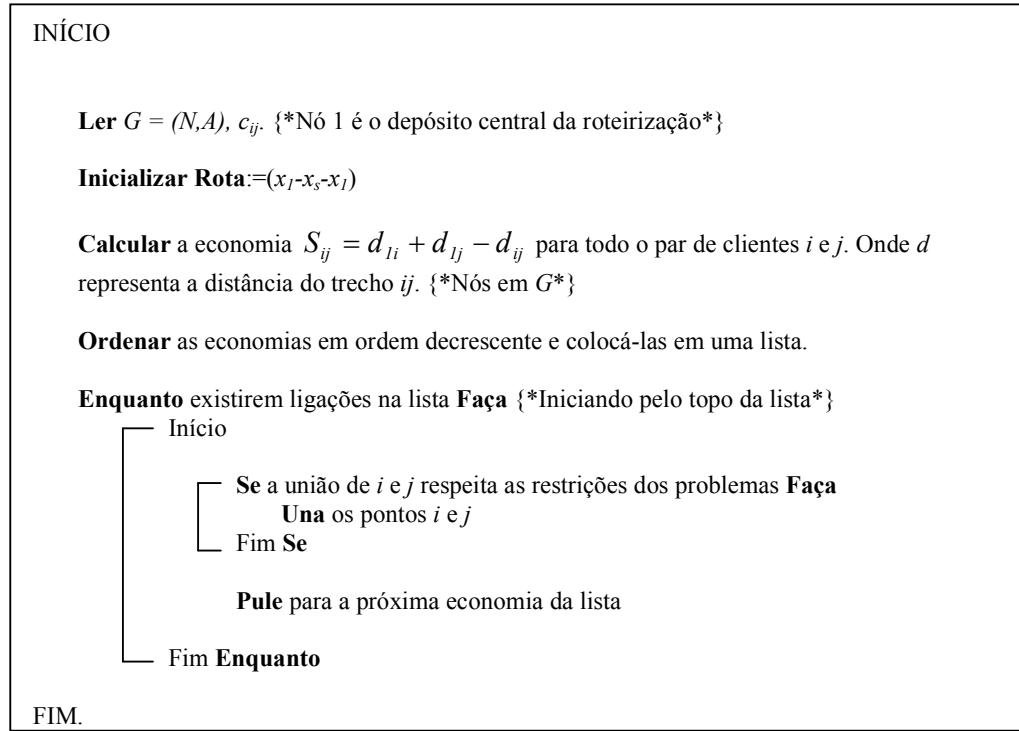
**Passo 4:** Vá para o topo da lista de economias.

**Passo 5:** Se ligando os pares de nós associados a posição da lista em questão resulta numa rota factível, adicione esta ligação à solução. Caso contrário, elimine-a.

**Passo 6:** Se ainda houver economias na lista, tome a próxima e volte ao Passo 5. Se não houver mais economias, vá para o Passo 7.

**Passo 7:** Retorne a solução obtida.

O procedimento da versão paralela da heurística de Clarke & Wright também pode ser entendido através de seu algoritmo na Figura 2.4:



**Figura 2.4 - Algoritmo para a versão paralela de Clarke & Wright**

### Versão Seqüencial

**Passo 4:** Vá para o topo da lista de economias.

**Passo 5:** Se ligando os pares de nós associados a posição da lista em questão resulta numa rota factível, adicione esta ligação à solução. Caso contrário, elimine-a.

**Passo 6:** Defina como rota atual, a rota que contém os nós  $i$  e  $j$  unidos no Passo 5.

**Passo 7:** Identifique os nós extremos (nó posterior e anterior ao depósito) da rota atual e armazene-os nas variáveis  $k$  e  $l$ . Por exemplo, após o Passo 5,  $k = i$  e  $l = j$ .

**Passo 8:** Selecione a maior economia  $s_{ik}$ ,  $s_{kj}$ ,  $s_{lj}$  ou  $s_{il}$  da lista que pode ser utilizada para estender a rota atual. Note que a rota a ser unida à rota atual deve necessariamente começar com  $(0, k)$  ou  $(0, l)$ , ou terminar com  $(k, 0)$  ou  $(l, 0)$ . Se for impossível estender a rota atual vá para o Passo 10.

**Passo 9:** Faça a união dos nós identificados e volte ao Passo 7.

**Passo 10:** Volte para o topo da lista de economias e encontre a primeira economia que gere uma união factível. Se nenhuma economia for encontrada vá ao Passo 12.

**Passo 11:** Faça a união dos nós identificados e defina como *rota\_atual* a rota que contenha esses nós. Volte ao Passo 7.

**Passo 12:** Fim.

A versão seqüencial da heurística de Clarke & Wright também pode ser entendida, de modo simplificado, através do algoritmo mostrado na Figura 2.5.

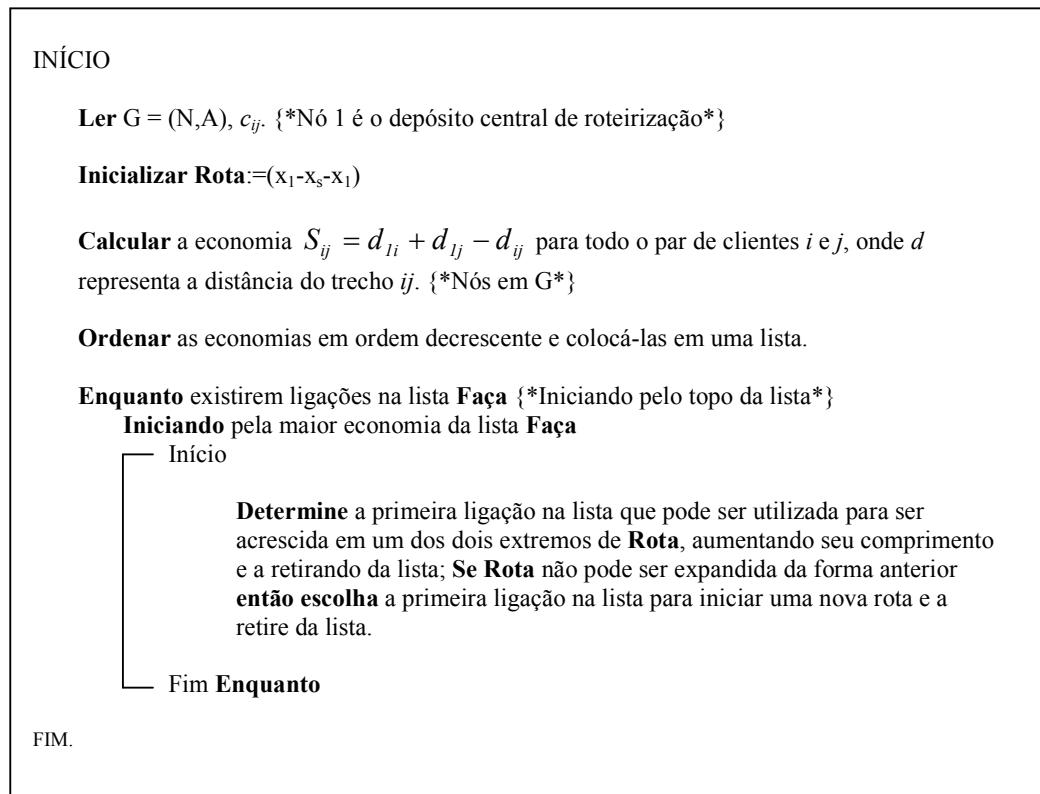


Figura 2.5 - Algoritmo para a versão seqüencial de Clarke & Wright

### (iii) Outras heurísticas para resolução do *VRP*

Além do algoritmo de Clarke & Wright e do algoritmo do Vizinho Mais Próximo, outros métodos heurísticos são propostos na literatura para o problema de roteirização de veículos. Em um dos trabalhos mais abrangentes, Laporte *et al.* (2000) fazem uma comparação entre métodos heurísticos clássicos e as heurísticas modernas para o problema de roteirização de veículos. Na primeira parte, apresenta algumas heurísticas clássicas como: (i) o método das economias de Clarke e Wright (1964), (ii) o algoritmo da varredura apresentado por Gillett e Miller (1974), e (iii) o algoritmo de 2 fases (*cluster-first, route-second*) apresentado por Fisher e Jaikumar (1981). Na segunda parte, apresenta alguns métodos heurísticos baseados em busca tabu e que provaram serem eficientes para resolução do problema de roteirização de veículo. Ao final são apresentados experimentos computacionais comparativos entre as duas abordagens.

## 2.5 - Variações do problema de roteirização de veículos

Nesta seção, são apresentadas algumas das variações do problema clássico de roteirização de veículos e alguns trabalhos encontrados na literatura para resolução dos mesmos.

Dentre todas as variações do *VRP* encontradas na literatura, são expostas nesta seção apenas aquelas que interferem no tipo de problema estudado. Primeiramente, é apresentado o problema de roteirização veículos com frota heterogênea (*HFVRP*), mostrando algumas abordagens para resolução de problemas de roteirização utilizando veículos de capacidades diferentes, como é proposto no presente trabalho. Em seguida, é conceituado o problema de roteirização com entregas fracionadas (*VRPSD*), que apesar de se assemelhar ao tipo de problema tratado, não se caracteriza como o mesmo. Por fim, é apresentado o problema de dimensionamento e roteirização com frota heterogênea (*FSMVRP*), que é o tipo de problema no qual se classifica o trabalho proposto.

### 2.5.1 - Problema de roteirização de veículos com frota heterogênea

O problema de roteirização de veículos com frota heterogênea (*Heterogeneous Fleet Vehicle Routing Problem - HFVRP*) é uma variação do problema clássico de roteirização de veículos, na qual a frota é composta por veículos de diferentes tipos. Neste caso, os diferentes tipos podem se referir a veículos com capacidades diferentes, custos diferentes, ou alguma outra característica que se faz necessária sua distinção de acordo com a necessidade do problema estudado.

Assim como a maioria dos problemas de roteirização de veículos, o *HFVRP* é do tipo *NP-difícil*. Desta forma, muitos trabalhos na literatura que abordam o tema utilizam métodos heurísticos para sua resolução.

Gendreau *et al.* (1999) propõem um método heurístico baseado em busca tabu para resolução do problema de roteirização de veículos com frota heterogênea. O método utiliza o algoritmo *GENIUS*, que é uma heurística geral de inserção desenvolvida para o problema do caixeiro viajante em Gendreau *et al.* (1992) para construção de roteiros. Além disso, utiliza o método de busca chamado *AMP (Adaptative Memory Procedure)* desenvolvido por Rochat e Taillard (1995), apresentando ao final, soluções de boa qualidade comparada a outras heurísticas eficientes.

Taillard (1999) utiliza busca tabu e programação linear para resolver o *HFVRP*. Seu método é conhecido como heurística de geração de colunas (*Heuristic Column Generation – HCG*). Para cada tipo e para um número ilimitado de veículos, um problema clássico de roteirização de veículos com frota homogênea (*VRP*) é resolvido. Uma vez que são resolvidos *VRP*'s para cada tipo de veículo, as rotas mais distantes contendo os mesmos clientes de soluções anteriores são descartadas. Ao final, a solução para o *HFVRP* é encontrada por meio de programação linear inteira nas rotas remanescentes.

### **2.5.2 - Problema de roteirização de veículos com entregas fracionadas**

O problema de roteirização de veículos com entregas fracionadas (*Vehicle Routing Problema with Split Deliveries – VRPSD*) é uma variação do problema clássico de roteirização de veículos, onde cada ponto de destino pode receber mais de um veículo. Neste caso, a demanda de cada ponto pode ser muito maior que a capacidade de um único veículo.

Segundo Dror *et al.* (1992), trata-se de uma relaxação do problema clássico de roteirização de veículos (*VRP*), cuja restrição 2.2 de sua formulação matemática (seção 2.4) é eliminada. Desta forma, a demanda do cliente pode ser fracionada em vários veículos. Dror *et al.* (1992) provaram que esta relaxação pode proporcionar economias substanciais, tanto em distância total percorrida, como no número de veículos utilizados na solução ótima.

Na literatura, poucos trabalhos abordam o *VRPSD*. Dror e Trudeau (1990) apresentam um método heurístico mostrando vantagens em entregar fracionadamente em um cliente. Dror *et al.* (1992) utilizam programação linear inteira com a relaxação de outras restrições da formulação matemática para o problema clássico de roteirização de veículos para resolução do *VRPSD*.

### **2.5.3 - Problema de dimensionamento e roteirização veículos**

No problema de dimensionamento e roteirização de veículos (*Fleet Size and Vehicle Routing Problem – FSVRP*) deve-se determinar o número de veículos necessários (frota ilimitada), assim como o roteiro de cada veículo, de forma a minimizar os custos fixos dos veículos e os custos variáveis de roteirização (em função da distância), garantindo que a demanda de todos os clientes será atendida.

No *FSVRP* original, a frota de veículos é considerada homogênea, portanto, a capacidade e os custos são idênticos. Uma variação do *FSVRP* é o problema de dimensionamento e roteirização de uma frota heterogênea de veículos (*Fleet Size and Mix Vehicle Routing Problem – FSMVRP*). Neste caso, considera-se o uso de uma frota

heterogênea ilimitada. O problema consiste em determinar, além dos roteiros e da configuração ideal dos veículos em termos de tamanho, a composição da frota, com o objetivo de minimizar a soma dos custos fixos e variáveis que podem ser dependentes ou não do tipo de veículo.

Dell'Amico *et al.* (2007) apresentam uma heurística construtiva de inserção e uma metaheurística para resolução do *FSMVRP*, considerando restrições de janela de tempo. Por meio da heurística construtiva, denominada *M\_ParallelRegret\_TW*, é formado um conjunto de soluções viáveis através da inserção de rotas para cada cliente, segundo um critério de prioridade, à cada solução formada. Novas soluções para o problema são geradas através da metaheurística *RR (Ruin and Recreate)*, apresentada por Schrimpf *et al.* (2000), destruindo algumas soluções viáveis já existentes e recriando soluções melhores.

Brandão (2008) apresenta um método heurístico baseado na busca tabu determinística para resolução do *FSMVRP*. Além da metaheurística, o método utiliza o algoritmo GENIUS de Gendreau *et al.* (1992), o algoritmo do Vizinho Mais Próximo e o algoritmo de inserção para construir as soluções iniciais. O algoritmo GENIUS também é utilizado na estratégia de intensificação na busca tabu.

### **(i) Heurísticas de economias para o *FSMVRP***

Conforme apontam Teixeira e Cunha (2002), a tradicional heurística de economias proposta por Clarke e Wright (1964), tanto na sua versão paralela (em que todos os roteiros são formados ao mesmo tempo) quanto na seqüencial (em que apenas um roteiro é montado de cada vez), não é adequada para problemas que envolvem frota heterogênea, uma vez que leva em conta só as distâncias para o cálculo das economias. Em ambas as versões, os pontos vão sendo agrupados, formando roteiros parciais, seguindo uma ordem decrescente de economias ( $S_{ij}$ ) decorrentes da sua união.

A fim de superar esse problema, Golden *et al.* (1984) propuseram heurísticas para o problema de dimensionamento e roteirização de uma frota heterogênea (*FSMVRP*), e que se baseiam em generalizações da heurística de economias.

Inicialmente, a expressão de cálculo de economias apresentada anteriormente é alterada de forma a considerar custos variáveis no lugar de distâncias:  $S_{ij}^1 = c_{0i} + c_{0j} - c_{ij}$

Partindo do princípio que os custos totais envolvem também os custos fixos dos veículos, estes devem ser incorporados à equação. Considerando que  $F(Z)$  é o custo fixo do menor veículo capaz de atender à demanda  $Z$ , tem-se o método conhecido como *Combined Savings (CS)*:

$$S_{ij}^2 = S_{ij}^1 + F(Z_i) + F(Z_j) - F(Z_i + Z_j) \quad (2.10)$$

Os custos de oportunidades devem ser incorporados em função da capacidade não utilizada de um veículo maior. Considerando  $P(Z)$  como a capacidade do menor veículo que é capaz de servir uma rota de demanda  $Z$ , tem-se a expressão abaixo, proposta por Golden *et al.* (1984) e conhecida como *Optimistic Opportunity Savings (OOS)*:

$$S_{ij}^3 = S_{ij}^2 + F(P(Z_i + Z_j) - (Z_i + Z_j)) \quad (2.11)$$

Uma variação desta equação é a *Realistic Opportunity Savings (ROS)*, que assume que os custos de oportunidade devem apenas ser incluídos quando a combinação de duas rotas requer um veículo maior do que está sendo utilizado.

$$S_{ij}^4 = \begin{cases} S_{ij}^2 + F(P(Z_i + Z_j) - (Z_i + Z_j)) & \text{se } P(Z_i + Z_j) > \max(P(Z_i) + P(Z_j)) \\ S_{ij}^2 & \text{caso contrário} \end{cases} \quad (2.12)$$

Uma outra possibilidade é variar a definição de economias parametricamente de tal forma que um número diferente de soluções é formado para cada problema, com a melhor solução escolhida no final. Um exemplo deste procedimento é a modificação do algoritmo *ROS*, conhecido como algoritmo *ROS- $\gamma$* . Este algoritmo usa o parâmetro  $\gamma$  na equação da economia, variando de 0 a 3, resultando na nova fórmula:

$$S_{ij}^5 = S_{ij}^4 + (1 - \gamma)c_{ij} \quad (2.13)$$

Desta forma, dois parâmetros  $\gamma$  diferentes produzirão formas diferentes de juntar rotas, pois há variação do peso do comprimento do arco  $(i, j)$  em relação aos outros componentes da fórmula da economia. Quando  $\gamma$  cresce, a fusão de rotas com valores de  $c_{ij}$  maiores se torna menos atrativa.

**Tabela 2.2 - Heurísticas de economias**

<b>Heurística</b>	<b>Fórmula da Economia</b>
<i>CW</i>	$S_{ij}^1 = c_{0i} + c_{0j} - c_{ij}$
<i>CS</i>	$S_{ij}^2 = S_{ij}^1 + F(Z_i) + F(Z_j) - F(Z_i + Z_j)$
<i>OOS</i>	$S_{ij}^3 = S_{ij}^2 + F(P(Z_i + Z_j) - (Z_i + Z_j))$
<i>ROS</i>	$S_{ij}^4 = \begin{cases} S_{ij}^2 + F(P(Z_i + Z_j) - (Z_i + Z_j)) & \text{se } P(Z_i + Z_j) > \max(P(Z_i) + P(Z_j)) \\ S_{ij}^2 & \text{caso contrário} \end{cases}$
<i>ROS-<math>\gamma</math></i>	$S_{ij}^5 = S_{ij}^4 + (1 - \gamma)c_{ij}$

Fonte: Golden *et al.* (1984)

## 2.6 - Trabalhos desenvolvidos no âmbito da EPUSP

Nesta seção são apresentados alguns trabalhos sobre o problema de roteirização de veículos, reunindo pesquisas nas áreas de Engenharia Naval, Engenharia de Transportes e Engenharia de Sistemas Logísticos da Escola Politécnica da Universidade de São Paulo (Fonte: Wu (2007)).

- Cunha (1997) trata o problema de roteirização de veículos com restrições operacionais, tais como janelas de tempo e duração máxima de jornada de trabalho. A estratégia de solução proposta pelo autor é baseada na relaxação lagrangeana das restrições do modelo. São desenvolvidas três heurísticas diferentes: duas para frota homogênea (alocação seqüencial e alocação paralela) e outra para dois tipos de frota (agrupamento e alocação seqüencial). Ao final, compara-se o desempenho das três heurísticas propostas e depois se utiliza a última heurística para um caso prático.
- Brejon (1998) aborda um problema de programação de transporte de suprimentos para unidades marítimas de exploração de petróleo, visando a garantia de que os suprimentos necessários estejam na unidade marítima solicitante na quantia correta

e dentro dos horários solicitados. É analisado como um problema de roteirização e programação de veículos com restrição de janelas de tempo. A estratégia de solução utilizada se baseia na heurística de inserção II, proposta por Solomon (1987), modificada para se adequar às restrições do problema estudado. A alocação dos clientes às rotas segue um critério de primeiro inserir em uma rota os clientes mais distantes e depois os clientes não alocados que têm menor valor de fim de janela de tempo, obviamente a inserção é feita juntamente com um estudo de viabilidade de inserção, que está relacionado a limites de capacidade, distância e tempo.

- Souza (1999) propõe uma estratégia de solução para o problema de transporte do tipo “Carga Única Coleta e Entrega” (*Full Truckload Pickup and Delivery*) com janelas de tempo em duas etapas: na primeira são gerados todos os roteiros tecnicamente viáveis para atender as requisições de transporte dentro do período de programação, e na segunda é resolvido um problema de programação linear, do tipo partição de conjunto (*set partitioning*) para a seleção dos melhores roteiros que minimizem o custo total do transporte.
- Teixeira (2001) trata o problema de composição (ou dimensionamento) e roteirização de uma frota de veículos heterogênea, levando em conta custos fixos e variáveis, além de restrições operacionais, com o objetivo de minimizar os custos de distribuição. A estratégia de solução está baseada em um algoritmo *out-of-kilter* para o agrupamento de clientes segundo uma medida de economia que leva em conta não só a distância como o custo fixo dos clientes. Para o agrupamento de clientes, o problema de designação foi modelado como de circulação com custo mínimo. As heurísticas produzidas (básica, híbrida e solução direta) se baseiam nas combinações de rotas a partir da solução dos problemas de designação de frota, uma vez que ao compor e roteirizar uma frota de veículos, os custos de distribuição são minimizados.
- Feriancic (2005) estuda o problema de distribuição de combustíveis com caminhões tanque para postos de abastecimento. A grande particularidade deste problema é a frota ser heterogênea devido a diferentes conjuntos possíveis de compartimentação para transporte dos produtos, permitindo assim o transporte de um ou mais produtos em um mesmo caminhão tanque para mais de um cliente. Apesar de ser permitido transportar mais de um produto por caminhão (gasolina, álcool e diesel), cada

compartimento tem que estar sempre completamente cheio ou vazio. A estratégia de solução proposta baseia-se em definir a alocação ótima dos pedidos aos veículos e a seqüência de entrega de cada veículo. A frota própria já é pré-determinada e há a possibilidade de uso de frota terceirizada. Os clientes são alocados segundo uma ordem decrescente de dificuldade nos veículos, também ordenados de acordo com a sua serventia. Isto é feito para que a heurística de inserção seqüencial inclua os clientes segundo uma ordem decrescente em veículos de maior serventia ainda disponíveis. A fim de validar a formulação matemática, bem como testar a sua consistência, foram utilizados o Solver do Excel e o CPLEX da ILOG; entretanto não foi possível obter a solução ótima mesmo para instâncias de menor porte. A heurística proposta baseia-se no GRASP (*Greedy Randomised Adaptive Search Procedure*) a fim de promover vários reinícios da heurística de alocação seqüencial, realocando apenas alguns clientes escolhidos aleatoriamente.

- Mourad (2005) trata de um problema de roteirização de carga completa com janelas de tempo que incorpora algumas singularidades encontradas nas empresas do mercado brasileiro. Uma diferença relevante para o problema clássico comumente estudado na literatura é a possibilidade de utilização de uma frota “spot” como complemento de uma frota dedicada. São consideradas ainda janelas de tempo, paradas especiais para execução de serviços (manutenção do veículo, por exemplo), e múltiplos depósitos, dentro de um horizonte de planejamento semanal, conferindo ao problema uma característica dinâmica. A frota dedicada já é pré-dimensionada, então os clientes que não puderem ser atendidos pela frota dedicada ou mesmo aqueles cujo atendimento por ela não for viável são automaticamente atendidos por uma frota terceirizada, ou “spot”. Foram propostas quatro diferentes heurísticas, todas baseadas em busca tabu. A solução inicial é obtida através de uma heurística de inserção inspirada em Solomon (1987). A diferença entre as quatro heurísticas propostas se encontra na estrutura de controle da busca, como tipos de proibição e período tabu. Primeiramente, são geradas doze instâncias com diferentes número de viagens por dia e restrições locais de coleta e entrega, denominadas como problemas reduzidos. Em seguida, quatro cenários com diferentes parâmetros para problemas triviais da literatura também foram gerados. A heurística TS4 demonstrou melhor desempenho em todos os testes, bem como no modelo exato, o

que mostra que a utilização de busca tabu para solucionar problemas de roteirização com janelas de tempo permite obter bons resultados.

- Bonasser (2005) estuda o problema de roteirização de veículos com múltiplos depósitos e frota heterogênea fixa. Sua estratégia de solução se baseia em heurísticas de economias, busca tabu, colônia de formigas e uma metaheurística híbrida (método *Routing AnTS*), desenvolvida especialmente para a solução do problema em questão. O autor também explora o desempenho dos mecanismos de diversificação e intensificação embutidos nos métodos. São utilizados com a finalidade de testar as estratégias de solução conjuntos de instâncias clássicas para problemas de roteirização de veículos em sua forma padrão, com frota heterogênea e com múltiplos depósitos. Conclui-se que a busca tabu efetua a intensificação com maior eficiência, enquanto a colônia de formigas promove melhor diversificação. A metaheurística híbrida, por sua vez, apresenta melhor desempenho que as estratégias anteriores de uma maneira geral. Alguns resultados obtidos por Bonasser (2005) superam os encontrados na literatura. Além da utilização dos problemas retirados da literatura, aplicou-se as estratégias de solução para uma situação real da Força Aérea Brasileira, no caso, a operação de assistência humanitária realizada pela mesma. O autor comprova que é possível a aplicação dos métodos neste caso.
- Wu (2007) estuda o problema de roteirização periódica de veículos no planejamento de entregas para períodos longos, fazendo com que os veículos executem diversas viagens, visitando clientes com freqüência. Dois procedimentos diferentes foram propostos para a alocação dos clientes aos dias de visitas: uma heurística de inserção seqüencial que visa equilibrar os esforços dos diferentes dias do período de planejamento, e uma heurística baseada em algoritmos genéticos. As rotas diárias são construídas através da utilização do algoritmo de economias de Clarke e Wright. Os resultados obtidos ficam próximos aos melhores da literatura, apontando para um método viável e adequado para resolução deste tipo de problema.

## 2.7 - Problema de *bin-packing*

Nesta seção é apresentado o problema de *bin-packing* (*BPP*), cujos métodos de resolução abordados a seguir são utilizados no presente trabalho, aplicados na resolução do problema de roteirização de veículos.

Problemas de empacotamento do tipo *bin-packing* (*BPP*) são problemas de otimização combinatória que buscam determinar o arranjo ótimo de unidades menores (itens) dentro de unidades maiores (objetos) satisfazendo determinadas restrições (Dyckhoff e Finke, 1992). Os problemas de empacotamento são fundamentais no planejamento da produção de vários segmentos industriais, cujo objetivo é minimizar os efeitos negativos gerados por desperdício de materiais e também relevantes no planejamento logístico como armazenagem, movimentação e transporte, visando minimizar espaços ociosos (Golden, 1976).

Para entender o problema de *bin-packing* (*BPP*) é necessária uma apresentação do problema da mochila binária (*knapsack problem*), de onde o *BPP* foi originado. A definição a seguir foi extraída de Martello e Toth (1990).

### 2.7.1 - Problema da Mochila Binária

Imagine que um alpinista precisa encher sua mochila selecionando dentre vários objetos possíveis aqueles que lhe proporcionarão a utilidade máxima. Este problema da mochila pode ser matematicamente formulado numerando os objetos de 1 até  $n$  e introduzindo um vetor formado por variáveis binárias  $x_j$  ( $j = 1, \dots, n$ ), com o seguinte significado:

$$x_j = \begin{cases} 1 & \text{se o objeto } j \text{ for selecionado} \\ 0 & \text{caso contrário} \end{cases} \quad (2.14)$$

Então, se  $p_j$  é a medida da utilidade proporcionada por um objeto  $j$ ,  $w_j$  é o seu tamanho e  $c$  a capacidade da mochila, o problema será selecionar dentre os vetores binários  $x$  que satisfazem a restrição (2.15)

$$\sum_{j=1}^n w_j x_j \leq c, \quad (2.15)$$

aquela que maximiza a função objetivo (2.16)

$$\sum_{j=1}^n p_j x_j \quad (2.16)$$

Desta forma, temos a seguinte formulação matemática para o problema da mochila binária:

$$\max z = \sum_{j=1}^n p_j x_j \quad (2.17)$$

$$\text{sujeito a: } \sum_{j=1}^n w_j x_j \leq c, \quad (2.18)$$

$$x_{ij} = 0 \text{ ou } 1, \quad j \in N = \{1, \dots, n\} \quad (2.19)$$

Desta forma, a função objetivo (2.17) visa maximizar a utilidade dos objetos colocados na mochila. A restrição 2.18 garante que a capacidade da mesma será respeitada.

No presente trabalho, pretende-se otimizar não só uma, mas várias mochilas, fazendo uma analogia com os veículos a serem carregados. Este tipo de problema é conhecido como *Bin-Packing Problem (BPP)*. Dados  $n$  itens e  $n$  mochilas (ou *bins*), de capacidade  $c$ , a formulação do problema é:

$$\min z = \sum_{i=1}^n y_i \quad (2.20)$$

$$\text{sujeito a: } \sum_{j=1}^n w_j x_{ij} \leq c y_i, \quad i \in N = \{1, \dots, n\}, \quad (2.21)$$

$$\sum_{i=1}^n x_{ij} = 1, \quad j \in N, \quad (2.22)$$

$$y_i = 0 \text{ ou } 1, \quad i \in N, \quad (2.23)$$

$$x_{ij} = 0 \text{ ou } 1, \quad i \in N, j \in N, \quad (2.24)$$

Onde,

$$y_i = \begin{cases} 1 & \text{se o bin } i \text{ é utilizado} \\ 0 & \text{caso contrário} \end{cases}$$

$$x_{ij} = \begin{cases} 1 & \text{se o item } j \text{ é designado para o bin } i \\ 0 & \text{caso contrário} \end{cases}$$

Percebe-se que no *BPP*, a função objetivo (2.20) passa a visar a minimização da quantidade de mochilas ou *bins* utilizados. A restrição 2.21 assegura que a capacidade de cada *bin* não seja excedida. A segunda restrição (2.22) garante que todos os itens serão colocados em algum bin.

O problema de *bin-packing* (*BPP*) possui grau de complexidade não polinomial, do tipo *NP - difícil*, e para sua resolução é apropriado o uso de métodos heurísticos. Alguns métodos já foram utilizados para este tipo de problema. No entanto, nem todos apresentam o grau de eficiência desejado.

### 2.7.2 - Heurísticas clássicas para resolução do *BPP*

Para a resolução dos problemas de *bin-packing*, são apresentadas algumas heurísticas clássicas da literatura, especificamente, *Next-Fit* (*NF*), *First-Fit* (*FF*) e *Best-Fit* (*BF*). A heurística *Next-Fit* (*NF*) é a mais simples. Inicia-se alocando o primeiro item ao *bin*. Os itens 2, ..., n são adicionados na seqüência dentro do *bin* corrente até que o próximo item da lista não caiba no mesmo. Neste momento, esse *bin* é fechado e começa-se a preencher um novo *bin*. A Figura 2.6 apresenta o algoritmo para heurística *Next-Fit*:

```

Início // Next Fit //
     $b(I) \leftarrow$  bin corrente
     $i(I) \leftarrow$  item corrente
    enquanto há itens não alocados em bins faca
        início
            se item corrente couber no bin corrente
                início
                    Aloque item corrente ao bin corrente
                fim // se //
            senão
                início
                     $b(x+I) \leftarrow$  bin corrente
                    Aloque item corrente ao bin corrente
                fim // senão //
             $i(y+I) \leftarrow$  item corrente
        fim // enquanto //
    fim // Next Fit //

```

**Figura 2.6 - Heurística *Next-Fit***

Segundo Martello e Toth (1990), o valor da solução obtida através da heurística *Next-Fit* é no máximo o dobro da solução ótima. Ou seja:

$$NF(I) \leq 2 \cdot z(I) \quad (2.25)$$

Onde  $z(I)$  representa o valor da solução ótima.

Já a heurística *First-Fit* (*FF*) oferece soluções melhores. Nela os *bins* não são fechados e cada item da lista é testado nos *bins* e alocado no primeiro *bin* que o comporte. A Figura 2.7 apresenta o algoritmo para heurística *First-Fit*:

```

Início // First Fit //
     $b(I) \leftarrow$  bin corrente
     $i(I) \leftarrow$  item corrente
     $Número\_de\_bins \leftarrow 0$ 
    enquanto há itens não alocados em bins faça
        início
            para  $X = 1$  até  $Número\_de\_bins$  faça
                início
                    se item corrente couber em  $b(X)$  então
                        início
                            Aloque item corrente ao  $bin$  corrente
                            Sair de para
                        fim // se //
                    se  $X = Número\_de\_bins$  então
                        início
                             $b(x+1) \leftarrow$   $bin$  corrente
                            Aloque item corrente ao  $bin$  corrente
                            Sair de para
                        fim // se //
                     $i(y+1) \leftarrow$  item corrente
                fim // para //
            fim // enquanto //
    fim // First Fit //

```

**Figura 2.7 - Heurística First-Fit**

Foi provado por Johnson, Demers, Ulman, Garey e Graham (1974) que o valor da solução obtida pela heurística *First-Fit* ( $FF(I)$ ) tem a seguinte relação com o valor ótimo ( $z(I)$ ):

$$FF(I) \leq \frac{17}{10} \cdot z(I) + 2 \quad (2.26)$$

Na heurística *Best-Fit* ( $BF$ ), os itens são alocados no *bin* que contém a menor capacidade residual. Similar a heurística *Best-Fit*, a heurística *Worst-Fit* ( $WF$ ) aloca os

itens aos *bins* com as maiores capacidades residuais. A Figura 2.8 apresenta o algoritmo para heurística *Best-Fit*:

```

Início // Best Fit //
     $b(I) \leftarrow$  bin corrente
     $i(I) \leftarrow$  item corrente
     $Número\_de\_bins \leftarrow 0$ 
    enquanto há itens não alocados em bins faz
        início
            Ordenar crescentemente bins pela quantidade residual
            para  $X = 1$  até  $Número\_de\_bins$  faz
                início
                    se item corrente couber em  $b(X)$  então
                        início
                            Aloque item corrente ao bin corrente
                            Sair de para
                        fim // se //
                    se  $X = Número\_de\_bins$  então
                        início
                             $b(x+1) \leftarrow$  bin corrente
                            Aloque item corrente ao bin corrente
                            Sair de para
                        fim // se //
                     $i(y+1) \leftarrow$  item corrente
                    fim // para //
                fim // enquanto //
fim // Best Fit //

```

**Figura 2.8 - Heurística *Best-Fit***

Foi provado também por Johnson, Demers, Ulman, Garey e Graham (1974) que o valor da solução obtida pela heurística está limitado no pior caso pela mesma relação obtida por *FF*, na equação (2.31). Outra forma de determinar a eficiência de um algoritmo para resolução do problema de *bin-packing* é através da razão entre um algoritmo aproximado  $A(I)$  com a solução ótima  $z(I)$ . Isto resulta num número mínimo real  $r^\circ(A)$ , tal que:

$$\frac{A(I)}{z(I)} \leq r^\infty(A) \quad (2.27)$$

Desta forma, pode-se prever que  $r^\infty(FF) = r^\infty(BF) = 17/10$ .

Existem ainda as versões *Next-Fit-Decreasing (NFD)*, *First-Fit-Decreasing (FFD)* e *Best-Fit-Decreasing (BFD)*, onde os itens são previamente colocados em ordem decrescente de tamanho e costumam gerar melhores resultados que as heurísticas anteriores. Johnson (1973) provou que:

$$FFD(I) \leq \frac{11}{9} \cdot z(I) + 4 \quad (2.28)$$

O que conduz a uma razão  $r^\infty(FFD) = 11/9$  para o pior caso, representando um resultado melhor que o obtido pela heurística *FF*. Os resultados das heurísticas supramencionadas foram resumidos por Coffman, Garey e Johnson (1984), na Tabela 2.3:

**Tabela 2.3 - Razões de performance do pior caso para algoritmos de *bin-packing***

Algoritmo	Tempo computacional	$r^\infty$
NF	$O(n)$	2,000
FF	$O(n \log n)$	1,700
BF	$O(n \log n)$	1,700
NFD	$O(n \log n)$	1,691...
FFD	$O(n \log n)$	1,222...
BFD	$O(n \log n)$	1,222...

### 2.7.3 - Trabalhos na literatura sobre *BPP*

Martello e Toth (1990) apresentam também as heurísticas de limitantes inferiores ou *Lower Bounds (LB)* para resolução do problema, onde as mochilas são divididas em partes menores e é montada uma regra para agrupamento dessas partes. Bourjolly e

Rebetz (2005) aprofundam sobre o mesmo tema fazendo uma comparação entre *LB2* e *LB3*, dividindo a mochila em 2 e 3 partes, respectivamente.

Um caso particular do *BPP* é o *bin-packing* com *bins* variáveis ou *variable-sized bin-packing (VSBPP)*, onde os *bins* não têm tamanhos idênticos. Zhang (1997) prova que as heurísticas clássicas não fornecem bons resultados para este tipo de problema. Kang e Park (2003) apresentam variações nas heurísticas clássicas, de modo a oferecer bons resultados.

Apesar do problema clássico de *bin-packing* ser bem conhecido e muito explorado na literatura; poucos trabalhos são encontrados relacionando sua aplicação com a de problemas de roteirização de veículos. Apenas nos últimos anos, este tema foi levantado por alguns autores. Poh *et al.*(2005) fizeram uma analogia entre o problema de *bin-packing* com o problema de programação de veículos no transporte de multi-*commodities* com frota heterogênea. Altinel e Oncan (2005) utilizaram a idéia da heurística *First-Fit-Decreasing* em uma modificação do cálculo de economias do algoritmo de Clarke e Wright (1964).

Galvão (2004) estudou o problema de otimização de um sistema de coleta de resíduos de biomassa de madeira para fins energéticos no contexto de uma situação real de abastecimento de uma central produtora de energia. O problema de dimensionamento e programação da frota para um período de planejamento semanal foi modelado como uma generalização do problema de *bin-packing*. Dada a complexidade do problema, foi proposta uma estratégia híbrida, em que é utilizado algoritmo genético para a definição dos dias de coleta e o dimensionamento de mínima frota necessária, enquanto que a programação dos veículos em cada um dos dias do período de planejamento é resolvida por meio de um algoritmo exato baseado na formulação matemática do problema de *bin-packing*.

Em resumo, poucos trabalhos foram realizados abordando a resolução de problemas de roteirização de veículos com entregas fracionadas em conjunto com a resolução de problemas de *bin-packing* com *bins* de tamanhos variáveis (*VSBPP*). Esta é uma das grandes contribuições do método proposto no presente trabalho

## 2.8 - Considerações finais

Neste capítulo, foram revistos os conceitos teóricos sobre os problemas de roteirização de veículos, onde foi apresentado um modelo de taxonomia para classificação dos problemas. Esta é utilizada no capítulo seguinte, quando o problema proposto é caracterizado. Em seguida, foram apresentadas algumas heurísticas para resolução do problema de roteirização de veículos, como o algoritmo do Vizinho Mais Próximo e o algoritmo de Clarke & Wright

Na segunda parte do capítulo, foi apresentado o problema do *bin-packing* (*BPP*) e uma extensão deste problema no caso de *bins* de tamanhos variados (*Variable-Sized Bin Packing* – *VSBPP*). Como visto, assim como o problema de roteirização de veículos, estes são problemas combinatórios de difícil resolução por algoritmos exatos.

Nos capítulos seguintes, são apresentados detalhes sobre a descrição do problema do presente trabalho e do método de resolução proposto. Todos os pontos levantados no presente capítulo serviram de base teórica para a construção do método proposto.

## Capítulo 3 - Caracterização do Problema

### 3.1 - Introdução

Neste capítulo, é apresentado o problema em detalhes. A primeira parte é destinada à descrição do problema de agrupamento de cargas fracionadas de cimento ensacado. Em seguida, uma formulação matemática para o problema é proposta. Por fim, o problema é classificado conforme a taxonomia relatada no capítulo anterior.

### 3.2 - Descrição do problema

É dado um conjunto de pontos de entrega de cimento a ser atendido em um dado dia. Para cada ponto, são conhecidos: o cliente recebedor da mercadoria, a quantidade a ser entregue (em peso), o tamanho máximo do veículo que pode atendê-lo e a sua localização em termos do município onde se situa. Esta situação é exemplificada na Tabela 3.1.

**Tabela 3.1 - Exemplo de dados de entrada para o problema**

Entrega	Cidade	Região	Cliente	Qtd Sacos	Peso Total	Veículo máximo	Material
1	BELO HORIZONTE	MG	CAPARAO/R FERNANDES TOURINHO, 611	160	8.000	15.000	CP II-E-32 U-FORTE (PL) - 50KG - CIM
2	BELO HORIZONTE	MG	CCM/R.MAESTRO DELE DE ANDRADE,25	80	4.000	15.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
3	BELO HORIZONTE	MG	ARHEA /R ANTONIO ALBURQUERQUE, 230	80	4.000	15.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
4	BELO HORIZONTE	MG	DEPOSITO IRMAOS BECKER LTDA	64	1.600	15.000	CP III-40 RS U-FORTE (PL) - 25KG - CIM
5	BELO HORIZONTE	MG	AREIA BRILHANTE COM INDUSTRIA LTDA	20	500	15.000	CP III-40 RS U-FORTE (PL) - 25KG - CIM
6	BELO HORIZONTE	MG	DEPOSITO PRUDENTE DE MORAIS LTDA	20	500	15.000	CP III-40 RS U-FORTE (PL) - 25KG - CIM
7	BELO HORIZONTE	MG	ANDRADE VIEIRA/R ITAMBE DO MATO	80	4.000	15.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
8	BELO HORIZONTE	MG	LADRIMINAS INDUSTRIA E COM LTDA	80	3.200	15.000	CP V-ARI-FACIL U-RAPIDO (PL) - 40KG
9	BELO HORIZONTE	MG	COND DO ED VERSALHES	40	2.000	15.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
10	BELO HORIZONTE	MG	DEP DE MAT PARA CONST DOURADAO	80	4.000	15.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
11	BELO HORIZONTE	MG	CERAMICA SAO CRISTOVAO LTDA	32	800	15.000	CP III-40 RS U-FORTE (PL) - 25KG - CIM
12	BELO HORIZONTE	MG	COPERMIL /R JOSE LAPORTE NETO, 95	40	2.000	15.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
13	BELO HORIZONTE	MG	ROCAS LAJES LTDA	80	4.000	30.000	CP V-ARI-PLUS U-RAPIDO (PL) - 50KG
14	BELO HORIZONTE	MG	ODINI INDUSTRIAL E COMERCIAL LTDA	40	2.000	30.000	CP V-ARI-PLUS U-RAPIDO (PL) - 50KG
15	BELO HORIZONTE	MG	SOLIDO/R.CONDE RIBEIRO DO VALE,105	80	4.000	30.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
16	BELO HORIZONTE	MG	ALBINO COSTA/R.BERNARDO GUIMARAES	80	4.000	30.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
17	BELO HORIZONTE	MG	DINAMOIR, CASTELO DE ELVAS, 297	80	4.000	30.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
18	BELO HORIZONTE	MG	ARTEFACIL INDUSTRIA E COMERCIO LTDA	120	4.800	30.000	CP V-ARI-RS FACIL RAPIDO (PL) - 40KG
19	BETIM	MG	PASSOS CAMPOS/AV.AMAZONAS,548	200	10.000	15.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
20	BETIM	MG	EDENGE/RUA DOIS,23	40	2.000	15.000	CP II-E-32 U-FORTE (PL) - 50KG - CIM
21	BETIM	MG	ITENDA/R. UM, 205	80	4.000	15.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
22	BETIM	MG	COTENCO/BR 381, KM 427	128	3.200	15.000	CP III-40 RS U-FORTE (PL) - 25KG - CIM
23	BETIM	MG	TENDA/AV EDMEIA MATOS LAZZAROTTI,	200	10.000	15.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
24	BRUMADINHO	MG	HORIZONTES/R.B, 20	120	6.000	15.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
25	CAETE	MG	CASA CAETANO LTDA	40	2.000	15.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
26	CONTAGEM	MG	MARAO/VILA MUNICIPAL 5, 43	40	2.000	30.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
27	CONTAGEM	MG	LARANJEIRAS CONSTRUCOES LTDA	80	4.000	30.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
28	CONTAGEM	MG	M&M FAB COM ARTEFATOS DE CIMENTO	80	4.000	30.000	CP III-40 RS U-FORTE (PL) - 50KG - CIM
29	CONTAGEM	MG	IMPREC IND DE PREMOL CONCRETO LTDA	80	4.000	15.000	CP V-ARI-PLUS U-RAPIDO (PL) - 50KG
30	CONTAGEM	MG	CONCREMAX LTDA ME	80	4.000	15.000	CP V-ARI-PLUS U-RAPIDO (PL) - 50KG

Embora os clientes demandem diferentes tipos de produtos, a carga pode ser considerada homogênea em termos do seu acondicionamento e transporte no veículo, já que os produtos, na sua maioria, são acondicionados em sacos de 40kg e 50kg, sem qualquer restrição para carregamento no veículo, exceto a capacidade máxima do veículo, normalmente em peso para esse tipo de produto. Pode-se assumir que todos os clientes recebem a mercadoria em horário comercial, sem restrição de janela de tempo. O número de entregas normalmente alocadas a cada veículo, e o tempo por entrega, permitem o cumprimento das mesmas durante o horário comercial, sem que isso constitua uma restrição da programação.

Um mesmo cliente pode demandar mais de uma carga (tipos diferentes de cimento). No entanto, a soma das mesmas não ultrapassa a capacidade de um veículo. Além disso, as entregas para um mesmo cliente devem ser agrupadas prioritariamente na mesma viagem.

As entregas devem ser realizadas utilizando uma frota de veículos de terceiros, composta de veículos de diferentes tipos/tamanhos (ver Tabela 3.2) e considerada ilimitada em termos do número de veículos disponíveis de cada tipo. Para cada tipo de veículo é conhecida a sua capacidade de carga (em peso). Todos os veículos partem de um único ponto de origem, onde é feito o carregamento.

**Tabela 3.2 - Capacidade máxima dos veículos considerados**

	<b>Veículo</b>	<b>Capacidade</b>
Toco		8 ton
Truck		15 ton
Carreta		28 ton
Bitrem		40 ton

Os clientes possuem restrições quanto ao tamanho máximo de veículos que podem receber, visto que muitos deles não possuem infra-estrutura suficiente para receber

veículos grandes como carretas e bitrens, por exemplo. Além disso, alguns clientes podem estar localizados em regiões metropolitanas, onde a circulação de veículos pesados é proibida.

Para cada município de destino, é conhecido o valor do frete de entrega, que é proporcional à distância desde a origem, e depende do tipo/tamanho do veículo. Adicionalmente, o frete é pago por viagem, independente se o veículo foi totalmente ocupado ou não. Quando mais de uma entrega de diferentes municípios são agrupadas em um mesmo veículo, é considerado o valor de frete correspondente ao município mais distante da origem.

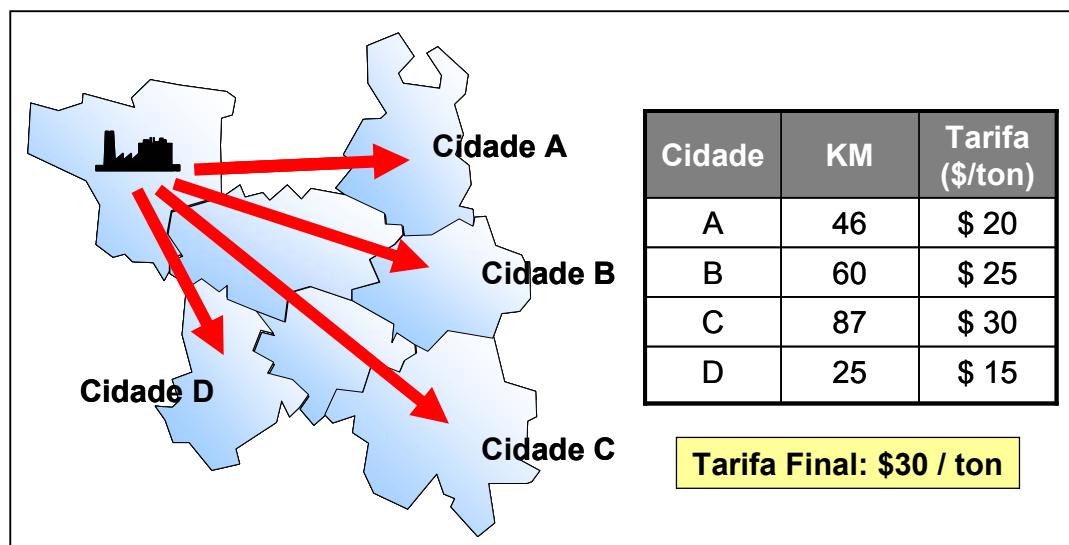
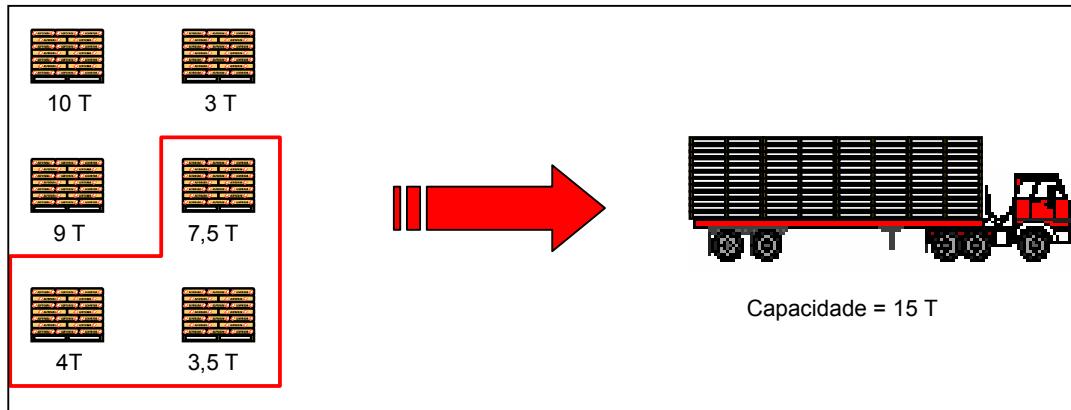


Figura 3.1 - Exemplo de cálculo de tarifa para agrupamento de entregas de 4 cidades

O objetivo do problema é designar/locar todas as entregas nos veículos, de modo a minimizar o custo total com o pagamento de fretes para esses transportadores.

Muitas entregas são formadas por cargas pertencentes a um mesmo cliente ou a um mesmo município. Neste caso, como as distâncias entre os pontos são consideradas irrelevantes, o problema de roteirização dos veículos se transforma num problema de simplesmente agrupar as cargas em um menor número de veículos possíveis. Neste ponto, surge a analogia com o problema de *bin-packing* (Figura 3.2).



**Figura 3.2 - O problema de agrupamento de cargas de cimento**

### 3.3 - Formulação matemática

A partir do detalhamento do problema acima, uma formulação matemática é proposta para o mesmo.

#### Parâmetros

$N$  = Número total de clientes a serem atendidos em um determinado período

$K$  = Número total de veículos disponíveis

$c_k$  = custo da viagem feita pelo veículo  $k$

$f_{ik}$  = frete do veículo  $k$  para atender cliente  $i$

$Q_k$  = Capacidade máxima do veículo  $k$

$w_i$  = demanda do cliente  $i$

$M_i$  = Tamanho máximo do veículo que poderá atender o cliente  $i$  (capacidade)

#### Hipótese

O número de veículos de cada tipo é considerado ilimitado. Em outras palavras, não é necessário alocar um veículo de um dado tipo devido à falta de disponibilidade de veículos de outro tipo.

### Variáveis de decisão

$$x_{ik} = \begin{cases} 1, & \text{se o cliente } i \text{ é atendido pelo veículo } k \\ 0, & \text{caso contrário} \end{cases}$$

$$y_k = \begin{cases} 1, & \text{se o veículo } k \text{ é usado} \\ 0, & \text{caso contrário} \end{cases}$$

### Formulação do modelo

$$\text{Min} \sum_{k=1}^K c_k \cdot y_k \quad (3.1)$$

$$c_k \geq f_{ik} \cdot x_{ik} \quad , \text{ para todo } k \in K, i \in N \quad (3.2)$$

$$\sum_{k=1}^K x_{ik} = 1 \quad , \text{ para todo } i \in N \quad (3.3)$$

$$\sum_{i=1}^N w_i \cdot x_{ik} \leq Q_k \cdot y_k \quad , \text{ para } k \in K \quad (3.4)$$

$$M_i \cdot x_{ik} \geq Q_k \cdot y_k \quad , \text{ para todo } k \in K, i \in N \quad (3.5)$$

- A função objetivo (3.1) visa minimizar o custo total das viagens.
- As restrições (3.2) garantem que o custo de cada viagem seja o valor referente ao maior frete entre as cargas dos clientes alocadas no veículo.
- As restrições (3.3) garantem que cada cliente será atendido por apenas um veículo.
- As restrições (3.4) garantem que a capacidade de cada um dos veículos não seja violada.
- As restrições (3.5) garantem que o tamanho do veículo alocado a cada cliente seja menor ou igual ao tamanho de veículo máximo permitido para aquele cliente.

Percebe-se através da formulação acima que o problema de roteirização ou seqüenciamento das entregas é deixado de lado, visto que o objetivo principal é a minimização do custo total com os frete pagos aos transportadores, o que independe da seqüência que os mesmos irão fazer. Desta forma, a formulação matemática proposta se assemelha muito a formulação do problema do *bin-packing* apresentada na seção 2.3.

### 3.4 - Classificação do problema

Conforme observado no Capítulo 2, há um número grande de variações possíveis para o problema de roteirização de veículos. Segundo a taxonomia apresentada na Tabela 2.1, pode-se classificar o problema proposto da seguinte forma (ver Tabela 3.3):

**Tabela 3.3 - Taxonomia para o problema proposto**

Parâmetro	Tipos
Função objetivo	<ul style="list-style-type: none"> <li>• Minimizar os custos totais de distribuição</li> </ul>
Restrições dos veículos	<ul style="list-style-type: none"> <li>• Limite de capacidade dos veículos;</li> </ul>
Restrições dos clientes	<ul style="list-style-type: none"> <li>• Limite com relação ao tipo de carga ou tipo de veículo;</li> </ul>
Restrições de rotas	<ul style="list-style-type: none"> <li>• Nenhuma</li> </ul>
Variáveis de decisão	<ul style="list-style-type: none"> <li>• Qual veículo é designado para cada cliente;</li> </ul>
Tipo de operação	<ul style="list-style-type: none"> <li>• Entrega;</li> </ul>
Tipo de carga	<ul style="list-style-type: none"> <li>• Múltiplas cargas ou carga fracionada.</li> </ul>
Tipo de demanda	<ul style="list-style-type: none"> <li>• Determinística.</li> </ul>
Localização da demanda	<ul style="list-style-type: none"> <li>• Demanda localizada somente em nós;</li> </ul>
Tamanho da frota	<ul style="list-style-type: none"> <li>• Ilimitada</li> </ul>
Tipo de frota	<ul style="list-style-type: none"> <li>• Heterogênea</li> </ul>
Depósitos e localização de veículos	<ul style="list-style-type: none"> <li>• Um único depósito</li> </ul>
Estrutura da rede	<ul style="list-style-type: none"> <li>• Direcionada</li> </ul>
Horizonte de planejamento	<ul style="list-style-type: none"> <li>• Curto prazo</li> </ul>

Por fim, baseando-se nos tipos de problemas de roteirização de veículos da seção 2.5, o problema proposto pode ser classificado como o problema de dimensionamento e

roteirização de uma frota heterogênea de veículos (*Fleet Size and Mix Vehicle Routing Problem – FSMVRP*).

É importante ressaltar que o conceito de cargas fracionadas tratado no presente trabalho diz respeito ao fracionamento das mesmas dentro de um mesmo veículo considerando entregas de diferentes clientes em um único caminhão. Este conceito é diferente do apresentado na seção 2.5.2, onde o fracionamento é caracterizado pela possibilidade de dividir as cargas a serem entregues para um único cliente, de modo a gerar maiores economias. Desta forma, a classificação como “*split deliveries*” não cabe ao problema aqui tratado.

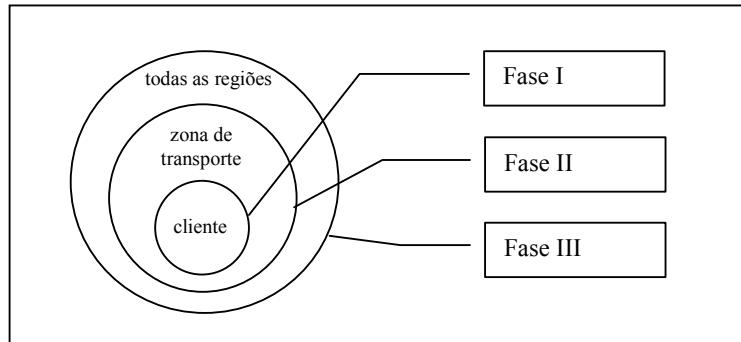
## Capítulo 4 - Estratégia de Solução

### 4.1 - Introdução

O problema de programação de entregas de cimento pode ser visto como uma combinação entre um problema que apresenta características de *bin-packing* e um problema de roteirização de veículos. A determinação de como vão ser feitas as entregas para o subconjunto de clientes correspondente a cada município, em termos de quantos e quais veículos utilizar e quais entregas alocar a cada veículo pode ser vista como uma generalização do problema de *bin-packing*. Os veículos representam as mochilas (ou *bins*), neste caso de diferentes tipos e tamanhos, nos quais as entregas devem ser alocadas de modo a minimizar a despesa total de frete. Para aquelas entregas que não permitiram completar um veículo deve ser então resolvido um problema de roteirização, para definir quais grupos de entrega devem ser alocadas a cada veículo, de modo que o frete seja minimizado.

Como visto anteriormente, tanto o problema de *bin-packing* quanto o de roteirização de veículos são, do ponto de vista matemático, problemas de natureza combinatória, em que o esforço computacional cresce exponencialmente com o tamanho do problema, o que limita a sua resolução através de algoritmos exatos (Garey e Johnson, 1999). Assim, propõe-se uma estratégia de solução heurística.

O método de solução é dividido em três fases, conforme mostrado na Figura 4.1: a primeira fase agrupa cargas de um mesmo cliente, a segunda agrupa cargas de clientes de uma mesma zona de transporte (cidade ou bairro), e por último, a terceira consolida cargas de clientes de regiões diferentes, as quais implicam em fretes distintos. Considera-se que a cada fase, as cargas já agrupadas na etapa anterior não poderão ser reagrupadas separadamente. O método descrito acima é similar à estratégia de solução “*cluster first-route second*” utilizada para a resolução de problemas de roteirização de veículos nos trabalhos de Gillett e Miller (1974) e Fisher e Jaikumar (1981).



**Figura 4.1 - As três fases do modelo**

Esta estratégia de divisão por fases foi utilizada de modo a reduzir o número de cargas a ser considerado em cada fase, e proporcionando um tratamento diferenciado a cada etapa do algoritmo.

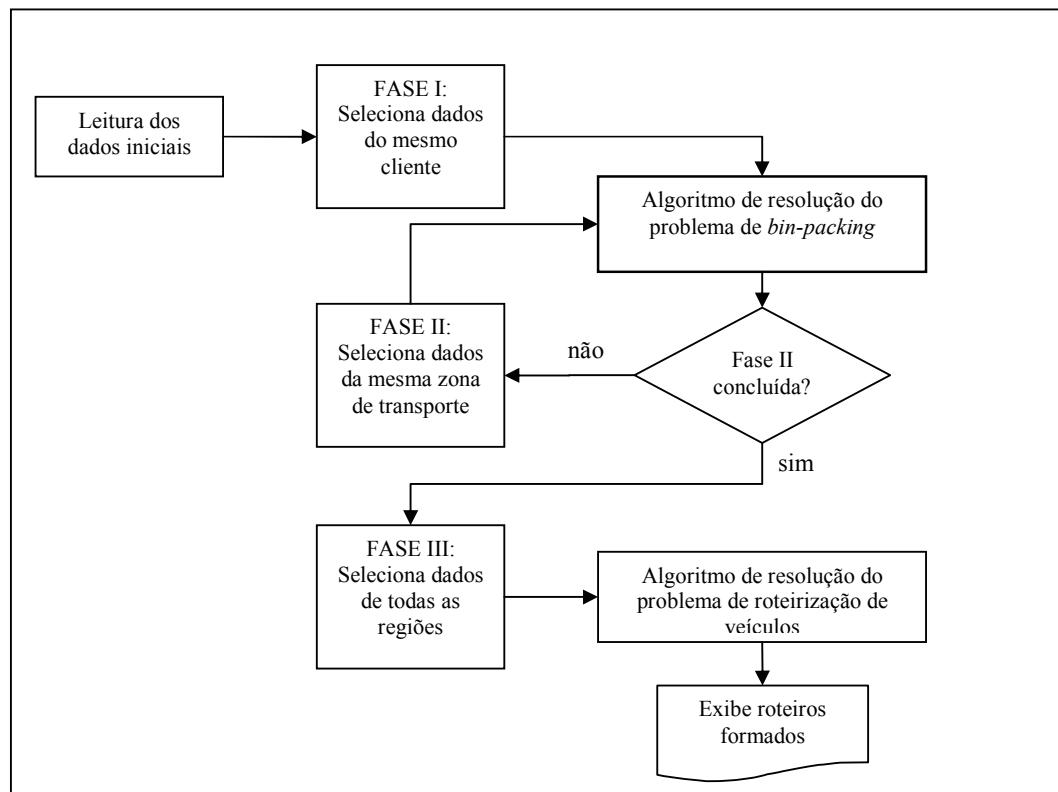
Para as cargas do mesmo cliente e da mesma cidade (primeira e segunda fase), em que o frete unitário não muda, será utilizado um método heurístico para resolução de problema com características de *bin-packing*, como explicado anteriormente.

Como já foi mencionado no item anterior, o menor ponto de localização dos clientes são as zonas de transporte, representadas dentro de uma cidade. Desta forma, a roteirização dos pontos dentro de uma zona de transporte não se faz necessária, dada a ausência de informações mais detalhadas sobre a posição geográfica dos clientes. O problema passa a ser o agrupamento das cargas de uma mesma cidade de modo a reduzir o número de veículos necessários para realizar as viagens e, com isso, minimizar a despesa total com o pagamento dos fretes. Lembrando também que os mesmos são pagos por viagem, independente se a capacidade do veículo foi totalmente utilizada ou não.

Para a Fase III, onde as distâncias devem ser levadas em consideração, serão incorporadas aos algoritmos genéticos outras heurísticas para obtenção da população inicial, tais como a versão paralela do método de Clarke & Wright ou algoritmo das economias e a heurística do Vizinho Mais Próximo. Nesta etapa, as cargas que não

foram agrupadas ou que foram agrupadas mas não completaram a capacidade total dos veículos nas etapas anteriores serão consideradas.

Dessa forma, podemos definir o fluxograma geral do método heurístico proposto como mostrado na Figura 4.2.



**Figura 4.2 - Fluxograma geral do método heurístico proposto**

#### **4.2 - Estratégia baseada em algoritmos genéticos para o problema de agrupamento de cargas**

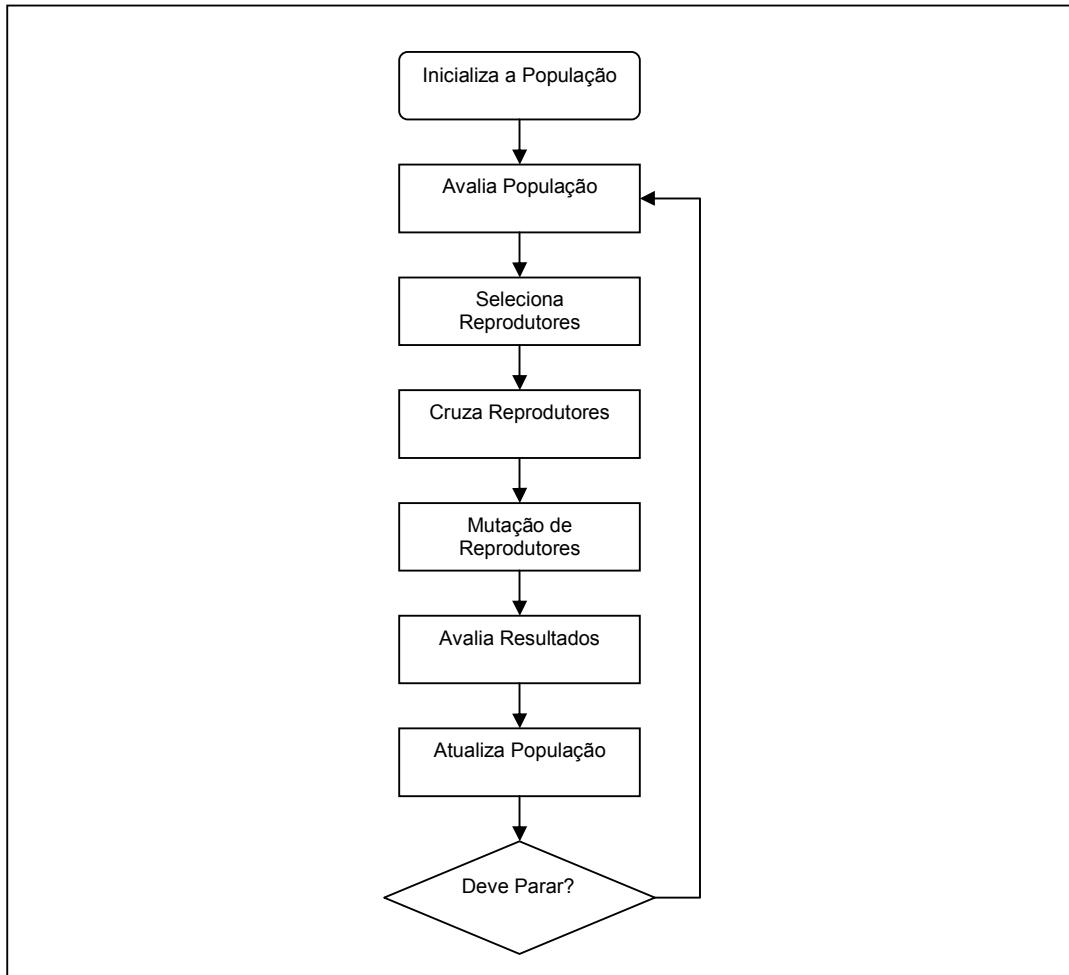
Nesta seção é apresentado o método baseado em algoritmos genéticos para resolução do problema de agrupamento de cargas modelado como problema de *bin-packing*.

Os algoritmos genéticos são métodos de busca baseados nos mecanismos de seleção natural e na genética. Idealizados e estabelecidos por Holland (1975) em sua obra *Adaptation in Natural and Artificial Systems*, esses algoritmos implementam estratégias

de buscas paralelas e aleatórias para solucionar problemas de otimização. No entanto, o fato de apresentarem buscas aleatórias não significa que não possuam direção. Eles exploram eficientemente informação histórica para encontrar novos pontos de resultado com expectativa de melhora do desempenho. Segundo Beasley (1999), pertencem a uma classe geral de algoritmos de busca probabilística “inteligente”, que se baseiam no processo evolucionário de algoritmos biológicos na natureza.

Em linhas gerais, os algoritmos genéticos trabalham com um grupo (ou população) de soluções. Cada indivíduo é representado por um cromossomo, que armazena as informações genéticas. Os cromossomos são compostos por genes, os quais são responsáveis pelas características dos seres e são trocados ou transmitidos durante o processo de reprodução. A cada iteração, indivíduos da população são selecionados para reprodução, sendo os mais aptos com maiores chances de serem transmitidos para a geração seguinte e os demais menos aptos simplesmente eliminados de acordo com o princípio darwiniano de seleção natural e sobrevivência do mais forte.

O funcionamento dos algoritmos genéticos pode ser representado pela Figura 4.3, apresentada em Cunha (2006). É dada uma população de soluções  $P(t)$  de uma iteração  $t$ , cada solução é avaliada a fim de se obter alguma medida da sua qualidade (“*fitness*”). Uma nova população  $P(t+1)$  correspondente à iteração  $t+1$  é então gerada, em duas etapas. Inicialmente indivíduos de  $P(t)$  são selecionados para agir como pais que vão produzir a geração seguinte  $P(t+1)$ . Em seguida, os pais passam por um cruzamento (acasalamento) para produzir os filhos que formarão a nova população  $P(t+1)$ . Uma mutação em algum indivíduo também pode ser realizada com certa probabilidade, geralmente muito baixa, com a finalidade de gerar diversidade ou variabilidade de características genéticas.



**Figura 4.3 - Esquema geral de funcionamento do Algoritmo Genético (Fonte: Cunha, 2006)**

O desenvolvimento de algoritmos genéticos envolve algumas etapas:

- **Representação.** Codificação de indivíduos de forma a representar as soluções do problema;
- **Geração da população inicial.** Definição do método para geração dos indivíduos que iniciam o processo de busca heurística;
- **Função aptidão.** Mede o grau de qualidade ou aptidão (“fitness”) de cada indivíduo da população;
- **Seleção.** Definição do método de seleção dos indivíduos que se reproduzirão;
- **Operador genético de reprodução (“crossover”).** Definição do método de troca de características entre os indivíduos;
- **Operador genético de mutação.** Definição do método de mutação dos indivíduos após a reprodução;

Referências sobre Algoritmos Genéticos podem ser encontradas em Goldberg (1989), Davis (1991), Michalewicz (1994), Gen e Cheng (1997), Beasley (1999), entre outros.

#### 4.2.1 - Representação

Num espaço de busca ou espaço de possíveis soluções para o problema, os algoritmos genéticos mantêm uma população de indivíduos, também chamados de “cromossomos”. Estes representam possíveis soluções para o problema a ser resolvido.

A representação cromossômica adotada para a resolução de cada problema de *bin-packing* é dada pela indicação do veículo (*bin*) em que cada entrega é alocada. A Figura 4.4 exemplifica esse conceito, indicando a alocação em termos do veículo, para cada uma das oito entregas a serem programadas, que demandaram quatro veículos.

1	2	3	4	5	6	7	8
4	1	1	3	2	2	3	4

Figura 4.4 - Representação do cromossomo no problema estudado

#### 4.2.2 - População inicial

A população inicial para o método proposto é formada quase em sua totalidade por indivíduos gerados aleatoriamente, com base em ordenações randômicas de entregas que são alocadas aos veículos até preencherem sua capacidade. Além desses indivíduos, são criados mais três cromossomos utilizando as heurísticas: *Next-Fit-Decreasing*, *First-Fit-Decreasing* e *Best-Fit-Decreasing* (Martello e Toth, 1990), conforme Figura 4.5.

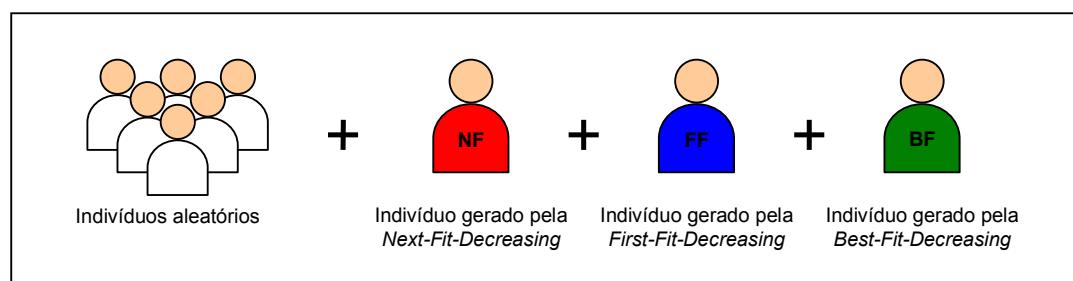


Figura 4.5 - Indivíduos da população inicial

#### 4.2.3 - Aptidão

Cada indivíduo de uma população tem associado um valor de aptidão, identificando seu grau de adaptabilidade ao ambiente. A função aptidão mostra a quanto uma solução é boa se comparada com outra solução da população. Esta função apresenta o grau de adaptação do indivíduo ao meio ambiente, para saber se possui ou não condição de sobreviver e criar descendentes.

A função aptidão para o método proposto é definida através da seguinte expressão proposta por Falkenauer e Delchambre (1992) para problemas de *bin-packing*:

$$f(S) = \frac{\sum_{i=1}^N (C_i/C_{\max})^k}{N} \quad (4.1)$$

Onde  $N$  é o número de veículos usados na solução  $S$ ;  $C_i$  é o peso total alocado ao veículo  $i$ ;  $C_{\max}$  é a capacidade máxima do veículo, e  $k$  é uma constante, onde  $k > 1$ . Esta função aptidão visa minimizar a ociosidade total dos veículos. A mesma não faz distinção em relação aos diferentes tipos de veículos e seus custos relacionados.

A constante  $k$  expressa a concentração na solução de *bins* igualmente preenchidos ou por desigualdades no preenchimento dos mesmos (alguns *bins* completos e outros vazios). Quanto maior o valor de  $k$ , maior a preferência pela formação de um grupo de *bins* completamente cheios, em oposição à formação de uma solução com *bins* preenchidos igualmente. Falkenauer (1996) prova que com  $k=2$  obtém-se as melhores soluções para o problema de *bin-packing*, e este valor será utilizado nos experimentos do presente trabalho.

Como é visto mais adiante, esta função é aplicada apenas às fases I e II do modelo, onde o problema proposto se comporta como um problema clássico *bin-packing*. Na seção 4.4 é explicado o procedimento para tratamento dos diferentes tipos de veículos dentro do algoritmo genético.

#### 4.2.4 - Seleção

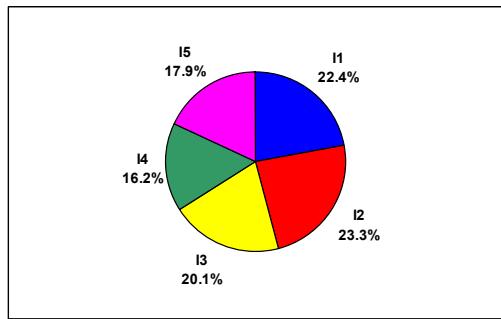
A etapa de seleção envolve a escolha dos indivíduos que irão para etapa de reprodução. A idéia é que aqueles que estiverem mais adaptados terão maiores chances de serem selecionados e repassarem suas características para a próxima geração.

O método utilizado para a seleção dos pais é o método da roleta. Neste método, cada indivíduo da população é representado na roleta conforme seu valor de aptidão. Desta forma, os indivíduos com elevada aptidão recebem uma área ou intervalo maior na roleta, enquanto aqueles que têm mais baixa aptidão recebem um menor intervalo na roleta. Após a distribuição na roleta, são gerados aleatoriamente números no intervalo entre 0 e o total do somatório da aptidão de todos os indivíduos da população. O indivíduo que possuir em seu intervalo o número gerado é selecionado para o cruzamento.

A Tabela 4.1 mostra o método da roleta. Na primeira coluna, estão os indivíduos da população. A segunda coluna apresenta os valores de aptidão para cada indivíduo. Na terceira coluna, estão os intervalos na roleta para cada indivíduo, calculados com o valor acumulado da soma da aptidão de todos os indivíduos. As porcentagens de cada indivíduo relativas à área da roleta são mostradas na quarta coluna. Desta forma, são gerados um conjunto de números aleatórios entre 0 e 4.07; os indivíduos selecionados serão os representantes do intervalo da roleta correspondentes a esses números, conforme Figura 4.6.

**Tabela 4.1 - Seleção através da roleta**

Indivíduo	Aptidão	Roleta	Porcentagem
I1	0.91	0.91	22%
I2	0.95	1.86	23%
I3	0.82	2.68	20%
I4	0.66	3.34	16%
I5	0.73	4.07	18%

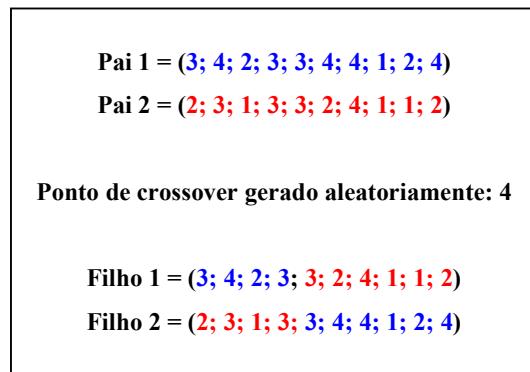


**Figura 4.6 - Percentuais de cada indivíduo na seleção**

#### 4.2.5 - Crossover

O *crossover* é o operador responsável pela combinação de características dos pais durante a reprodução, permitindo que as próximas gerações herdem essas características. Ele é considerado o operador genético predominante, por isso é aplicado com probabilidade dada pela taxa de *crossover*, que deve ser maior que a taxa de mutação.

Para o problema estudado foi considerado o cruzamento (*crossover*) de um único ponto. Para isso, uma posição dentro do gene é selecionada aleatoriamente. A partir daí, o restante dos genes são trocados entre os pais, conforme Figura 4.7.



**Figura 4.7 - Crossover**

Para o caso de cromossomos muito longos talvez haja a necessidade de considerar mais de um ponto de *crossover*.

#### 4.2.6 - Mutação

O operador mutação é considerado um operador secundário e consiste em inverter cada gene com uma probabilidade de mutação muito baixa. É necessário para a manutenção da diversidade genética da população, assegurando que as soluções não se prendam a mínimos locais. Com este mecanismo, altera-se levemente a direção da busca dando um toque de aleatoriedade.

Para o método proposto, a mutação ocorre com o sorteio de dois genes dentro de um cromossomo que trocam suas posições, conforme Figura 4.8.

<b>Cromossomo normal</b>	4	1	1	3	2	2	3	4
<b>Cromossomo mutado</b>	4	3	1	3	2	2	1	4

**Figura 4.8 – Mutação**

#### 4.2.7 - Reparo

Após as etapas de cruzamento e mutação, o cromossomo resultante pode ter violado a restrição de capacidade dos veículos (*bins*). Para reparar esse problema, é adotado um procedimento de reparo do cromossomo que consiste em percorrer todos os veículos e guardar num vetor todas as cargas que superam a capacidade dos veículos para o qual estão alocados. As cargas são selecionadas de acordo com o maior valor indexado ao vetor veículo. Em seguida, os itens são realocados nos veículos que estão mais próximos de sua capacidade máxima, num movimento similar à heurística de *Best-Fit*. Para os itens que não cabem em nenhum dos veículos ativos, torna-se necessária a alocação de veículos adicionais. A Figura 4.9 apresenta o algoritmo utilizado para o procedimento de reparo.

```

Início // Reparo //

para  $X = 1$  até  $Número\_de\_bins$  faz // Identificação dos bins violados //
  início
     $b(X) \leftarrow$  bin corrente
    se bin corrente está com a capacidade violada então
      guarda o último item de  $b(X)$  no vetor reparo  $r(Y)$ .
    fim // se //
  fim // para //

para  $Y = 1$  até  $Número\_de\_itens\_em\_r(Y)$  faz // Realocação dos itens para reparo //
  início
     $r(Y) \leftarrow$  item corrente
    Ordenar crescentemente bins pela quantidade residual
    para  $X = 1$  até  $Número\_de\_bins$  faz
      início
         $b(X) \leftarrow$  bin corrente
        se item corrente couber em bin corrente então
          início
            Aloque item corrente ao bin corrente
            Sair de para
          fim // se //
        se  $X = Número\_de\_bins$  então
          início
             $b(X+1) \leftarrow$  bin corrente
            Aloque item corrente ao bin corrente
            Sair de para
          fim // se //
         $r(Y+1) \leftarrow$  item corrente
      fim // para //
    fim // para //

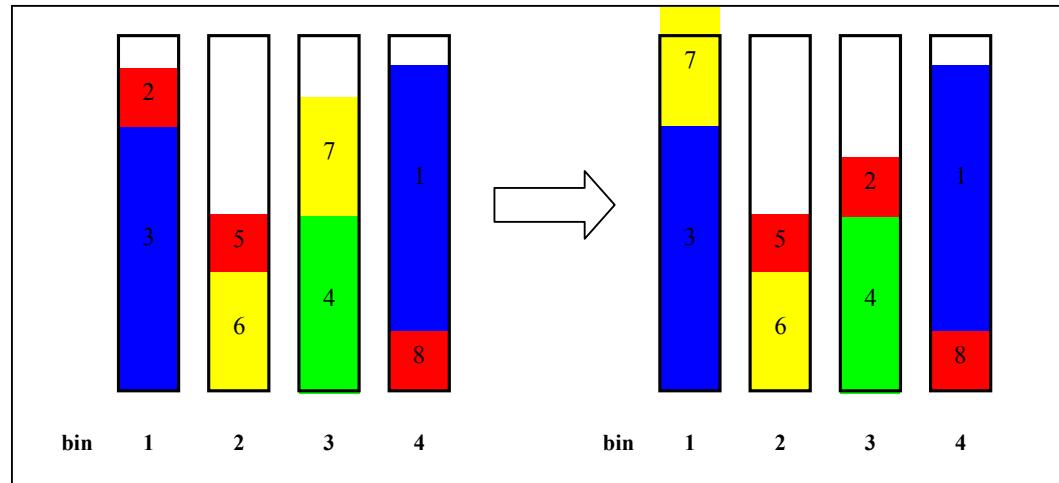
fim // Reparo //

```

**Figura 4.9 - Algoritmo para procedimento Reparo**

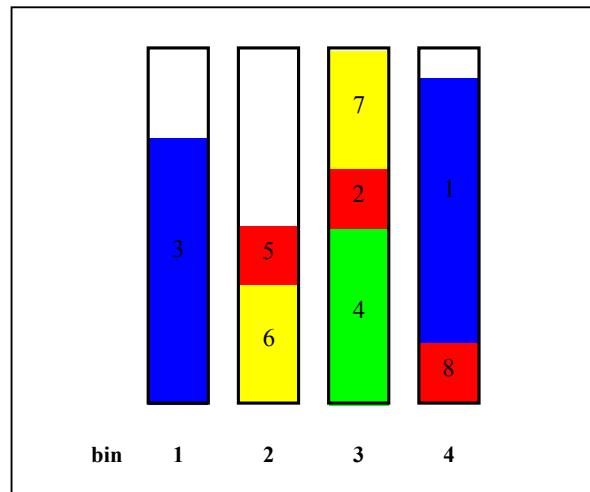
Para melhor entender esta etapa do algoritmo, é apresentado o exemplo a seguir. A Figura 4.10 abaixo representa os cromossomos durante o processo de mutação da Figura

4.8. Os itens selecionados para a mutação foram os itens 2 e 7, que estavam inicialmente nos *bins* 1 e 3, respectivamente. Percebe-se que após a troca, a adição do item 7 superou a capacidade do *bin* 1.



**Figura 4.10 - Exemplo da solução após mutação**

No processo de reparo, o item 7 é retirado do *bin* 1. Em seguida, ele é recolocado no *bin* com a menor capacidade e capaz de comportá-lo. Neste caso, seria o *bin* 3 conforme Figura 4.11. Caso, não houvesse outro *bin* com capacidade para realocá-lo, um novo *bin* seria criado.



**Figura 4.11 - Exemplo processo de reparo do cromossomo**

Dessa forma, são formados os indivíduos de cada geração da população.

Resumindo, as etapas do algoritmo genético utilizado para a resolução do problema de *bin-packing* podem ser resumidas como mostrado na Figura 4.12.

```

Procedimento Bin-Packing AG
Início
Geração da população inicial;
Para (t=1 até t = Max_iter), faça:
    Seleção;
    Reprodução;
    Mutação;
    Reparo;
Fim Para
Fim
```

**Figura 4.12 - Esquema geral do algoritmo genético para resolução do problema de *bin-packing***

Como resultado, têm-se todas as entregas alocadas aos veículos, respeitando restrições de capacidade.

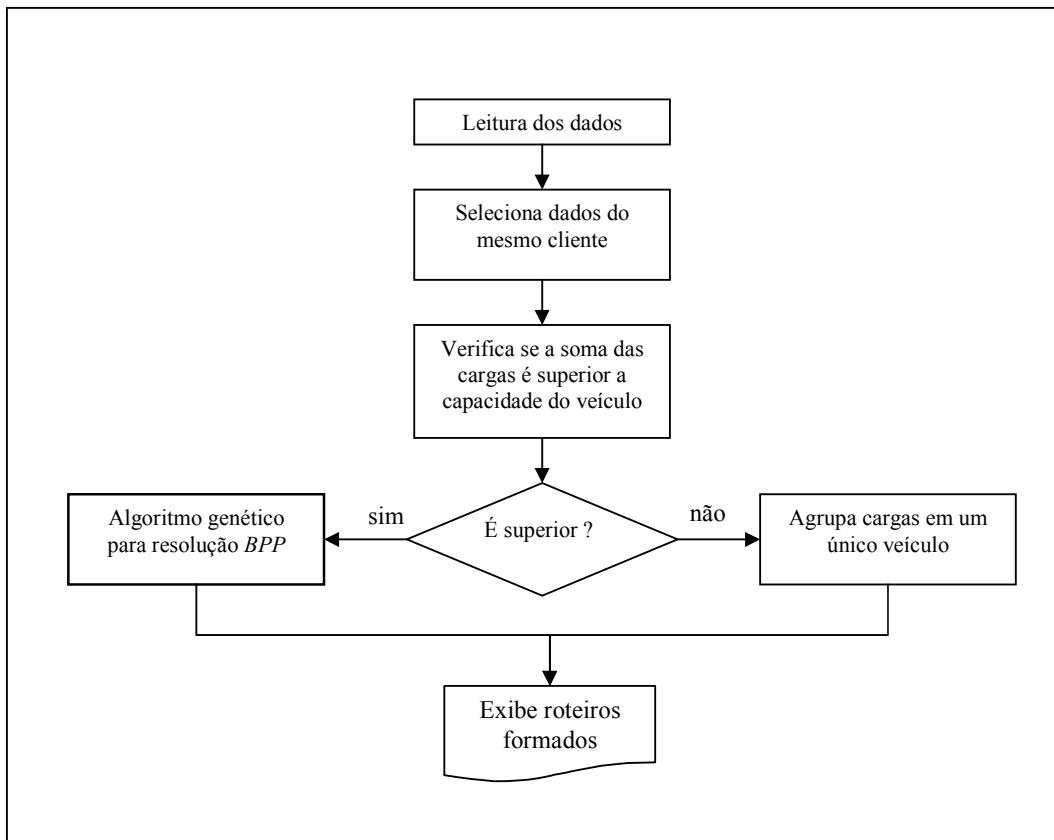
### **4.3 - Fase I: Agrupamento de cargas de um mesmo cliente**

Existem vários tipos de cimento no mercado, de forma que muitos clientes solicitam pedidos diferentes, uma para cada tipo de cimento. Com isso, no final do dia forma-se um conjunto de pedidos de cimento para um único cliente. Assim, é estabelecido o primeiro critério para agrupamento de pedidos ou entregas: o agrupamento de cargas de um mesmo cliente.

Nesta primeira fase, são listados todos os clientes que possuam mais de uma carga a ser entregue. Por cliente, estas cargas são agrupadas através de um algoritmo baseado em métodos heurísticos para resolução de problemas de *bin-packing* (*BPP*).

É importante frisar que nesta primeira fase, não é tratado ainda o problema de *bin-packing* com *bins* variáveis (*VSBPP*). Isto porque até este momento, as cargas estão sendo selecionadas por cliente, cada qual possuindo a restrição quanto a um único

tamanho de veículo máximo permitido. A Figura 4.13 abaixo mostra o fluxograma desta primeira fase do algoritmo:



**Figura 4.13 - Fluxograma da Primeira Fase**

Para demonstrar de uma forma mais clara o algoritmo, é proposto um exemplo simplificado. A Tabela 4.2, representa um conjunto de 10 entregas, solicitadas por 8 clientes, pertencentes a 4 cidades distintas. Além disso, são informados também o peso de cada entrega e o tamanho máximo do veículo que pode atender cada cliente.

**Tabela 4.2 - Exemplo simplificado da Fase I**

<b>Entrega</b>	<b>Cliente</b>	<b>Cidade</b>	<b>Peso</b>	<b>Cap. Veículo máximo</b>
1	1	A	5	15
2	1	A	5	15
3	2	A	2	40
4	3	A	3	40
5	4	B	10	28
6	4	B	10	28
7	5	B	10	40
8	6	C	15	40
9	7	C	15	40
10	8	D	8	40

Como pode ser observado na Tabela 4.2, apenas os clientes 1 e 4 possuem entregas que podem ser agrupadas em um único veículo. A Tabela 4.3 abaixo mostra o resultado deste agrupamento após a Fase I do algoritmo.

**Tabela 4.3 - Resultados após Fase I para exemplo simplificado**

<b>Veículo</b>	<b>Entregas</b>	<b>Cliente</b>	<b>Cidade</b>	<b>Peso</b>	<b>Cap. Veículo máximo</b>
1	1,2	1	A	10	15
2	3	2	A	2	40
3	4	3	A	3	40
4	5,6	4	B	20	28
5	7	5	B	10	40
6	8	6	C	15	40
7	9	7	C	15	40
8	10	8	D	8	40

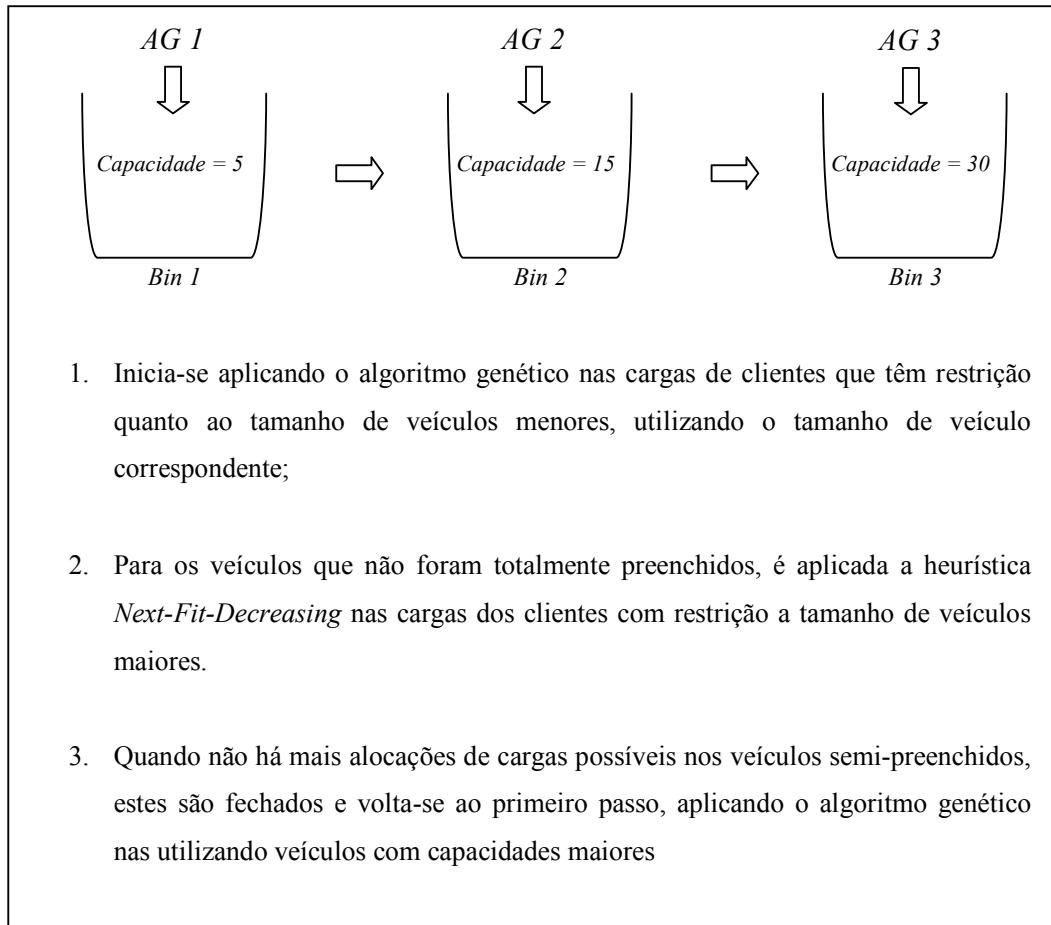
Para o restante das cargas foi designado um único veículo para atender cada entrega. Os resultados obtidos no final desta fase formarão os dados iniciais para a Fase II, onde serão agrupadas as cargas pertencentes à mesma cidade.

#### 4.4 - Fase II: Agrupamento de cargas de uma mesma cidade

Conforme visto anteriormente, o problema de alocação das entregas de cada município aos veículos pode ser visto como problema de *bin-packing* com *bins* de tamanhos variados (*Variable Sized Bin Packing Problem* – *VSBPP*). Dada a dificuldade de resolução desse problema, foi proposta uma estratégia de solução heurística baseada na metaheurística algoritmo genético.

Uma forma de contornar a complexidade adicional proporcionada pela possibilidade de usar veículos de tamanhos diferentes é aplicar a heurística proposta, baseada em algoritmos genéticos, para o problema do *bin-packing* tradicional (*BPP*), considerando os diferentes tipos/tamanhos de veículos separadamente, a partir do menor para o maior tamanho. Assim, resolve-se inicialmente, para cada município de entrega, o sub-problema correspondente aos clientes que requerem entregas com veículos menores. Caso tenha-se algum veículo não totalmente utilizado, busca-se alocar mais clientes de modo a completar a sua capacidade, considerando clientes que podem ser atendidos por veículos maiores.

Para tanto, é utilizada a heurística *Next-Fit-Decreasing*, descrita em Martello e Toth (1990), onde são consideradas as entregas remanescentes em ordem decrescente de peso. Uma vez atendidos todos os clientes que requerem atendimento pelo menor tamanho de veículo, e completados os veículos que realizam esse atendimento, passa-se então à resolução do problema de *bin-packing* correspondente aos clientes que só podem ser atendidos pelo tipo/modelo de veículo imediatamente superior. Esse procedimento é realizado sucessivamente, e finalizando com os clientes que não apresentem restrição quanto ao tipo/tamanho do veículo que possa atendê-lo, como ilustrado na Figura 4.14.



**Figura 4.14 - Abordagem para caminhões de diferentes capacidades**

Continuando o exemplo simplificado apresentado na seção anterior, é aplicada a heurística proposta para a Fase II para agrupar cargas de uma mesma cidade. A Tabela 4.4 abaixo mostra dois agrupamentos possíveis para as cidades A e C. Apesar de haver duas cargas para a cidade B, estas não podem ser agrupadas já que a soma dos pesos das cargas supera a capacidade do veículo máximo permitido para o cliente 4.

**Tabela 4.4 - Exemplo simplificado da Fase II**

<b>Veículo</b>	<b>Entrega</b>	<b>Cliente</b>	<b>Cidade</b>	<b>Peso</b>	<b>Cap. Veículo máximo</b>
1	1,2	1	A	10	15
2	3	2	A	2	40
3	4	3	A	3	40
4	5,6	4	B	20	28
5	7	5	B	10	40
6	8	6	C	15	40
7	9	7	C	15	40
8	10	8	D	8	40

A Tabela 4.5 apresenta o resultado do agrupamento após a Fase II da heurística proposta.

**Tabela 4.5 - Resultados após Fase II**

<b>Veículo</b>	<b>Entrega</b>	<b>Cliente</b>	<b>Cidade</b>	<b>Peso</b>	<b>Cap. Veículo máximo</b>
1	1,2,3,4	1,2,3	A	15	15
2	5,6	4	B	20	28
3	7	5	B	10	40
4	8,9	6,7	C	30	40
5	10	8	D	8	40

No Final da Fase II, todas as entregas pertencentes a clientes de uma mesma cidade foram agrupadas. As entregas restantes serão agrupadas na Fase III. É o caso dos veículos 2, 3, 4 e 5, que possuem capacidades residuais de forma a permitir o agrupamento de cargas.

#### **4.5 - Fases III: Agrupamento de cargas de cidades diferentes**

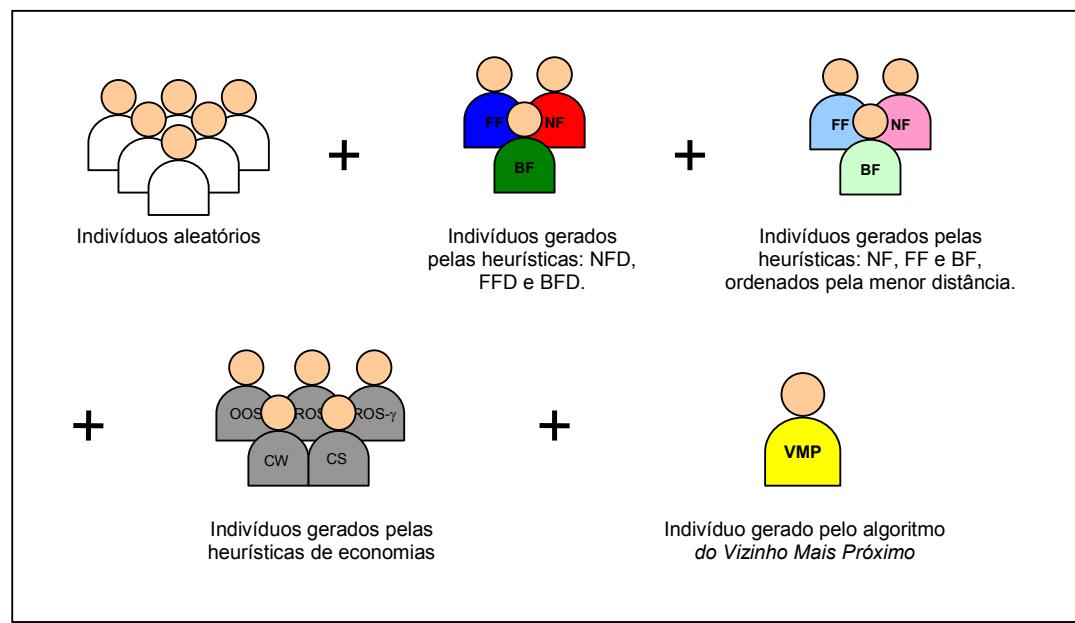
Na Fase III, as distâncias são levadas em consideração para o agrupamento das entregas remanescentes em veículos que não tiveram a sua carga completada. Para tanto, algumas modificações são aplicadas ao algoritmo genético utilizado nas fases I e II. Essas mudanças ocorrem nas fases de geração da população inicial e na função aptidão utilizada. O restante das fases permanece igual às das fases I e II.

#### 4.5.1 - População inicial para a Fase III

Tendo em vista que na Fase III as distâncias entre as cidades interferem na escolha do melhor agrupamento de rotas, algumas heurísticas construtivas foram adicionadas para formação da população inicial:

- Heurísticas *Next-Fit*, *First-Fit* e *Best-Fit* ordenadas pela menor distância da cidade ao local de origem;
- Algoritmo de Clarke & Wright clássico e outras extensões do algoritmo de economias (apresentado na seção 2.5.3);
- Algoritmo do Vizinho Mais Próximo (apresentado na seção 2.4.1).

Com isso, a população inicial para a Fase III apresenta os seguintes indivíduos apresentados na Figura 4.15.



**Figura 4.15 - Indivíduos da população inicial da Fase III**

#### 4.5.2 - Aptidão para a Fase III

Para considerar a distância na escolha da melhor solução, é proposta a seguinte função aptidão que substitui a apresentada na seção 4.2.3:

$$f(S) = \sum_{i=1}^N fmax_i \quad (4.2)$$

Onde  $N$  é o número de roteiros formados na solução  $S$ ; e  $fmax_i$  representa o valor do maior frete encontrado dentre os pontos do roteiro  $i$ . Esta equação calcula o custo total das viagens e está condizente com a formulação matemática para o problema proposto (seção 3.3). Desta forma, quanto menor a função aptidão melhor é a solução encontrada. Assim, na Fase III os indivíduos com menores valores de aptidão são mantidos.

#### 4.5.3 - Exemplo simplificado

Finalizando o exemplo simplificado da seção anterior, a heurística da Fase III é aplicada para agrupar o restante das cargas. A Tabela 4.6 abaixo mostra os agrupamentos possíveis para o problema.

**Tabela 4.6 - Exemplo simplificado da Fase III**

Veículo	Entrega	Cliente	Cidade	Peso	Cap. Veículo máximo
1	1,2,3,4	1,2,3	A	15	15
2	5,6	4	B	20	28
3	7	5	B	10	40
4	8,9	6,7	C	30	40
5	10	8	D	8	40

A Tabela 4.7 apresenta o resultado final do agrupamento, após a Fase III da heurística proposta.

**Tabela 4.7 - Resultados após Fase III**

Veículo	Entrega	Cliente	Cidade	Peso	Cap. Veículo máximo
1	1,2,3,4	1,2,3	A	15	15
2	5,6,10	4,8	B,D	28	28
3	7,8,9	5,6,7	B,C	40	40

Ao final da Fase III, todas as cargas são alocadas aos veículos correspondentes, formando a solução final para o problema. No capítulo seguinte, são apresentados os experimentos computacionais com resultados do modelo proposto.

## Capítulo 5 - Experimentos Computacionais

Os experimentos computacionais foram divididos em três etapas. Na primeira, é testado o algoritmo para resolução do problema de *bin-packing*, utilizado nas fases I e II, com dados de *benchmark* extraídos de uma base de dados (OR-Library). Na segunda etapa, o método completo é testado comparando a um programa atual utilizado na empresa estudada. Por fim, os resultados do método proposto são comparados a um método exato na terceira etapa.

Os parâmetros genéticos para o método proposto utilizados nas três etapas de são apresentados na tabela 5.1, abaixo:

**Tabela 5.1 - Parâmetros do AG**

Tamanho da população	100
Taxa de seleção (nº de pais)	4
Probabilidade de cruzamento	50%
Taxa de mutação	1%
Número máximo de iterações	200

Os algoritmos que compõem a estratégia de solução proposta foram implantados em linguagem *C++*, utilizando o compilador *DEV-C++*. Os testes foram realizados em um computador com processador *Pentium Centrino 1.8 GHz, 512 MB RAM*.

### 5.1 - Avaliação das Fases I e II

Primeiramente, tendo em vista a sua aplicação nas Fases I e II, foi avaliada a eficiência do algoritmo genético proposto para a resolução de nove instâncias de problemas extraídos da base OR-Library. Esta é uma coleção de dados de testes para uma série de problemas de Pesquisa Operacional. A OR-Library foi originalmente descrita por Beasley (1990) em seu artigo “OR-Library: distributing test problems by eletronic mail”.

Os dados de entrada de cada problema foram formados por itens de tamanhos variados, uniformemente distribuídos entre 20 e 100, para serem alocados em *bins* de capacidade igual a 150. Os resultados do algoritmo proposto foram comparados com os resultados do método *GGA* (*Grouping Genetic Algorithm*) apresentado por Falkenauer (1996) e com os resultados do procedimento *MTP*, proposto por Martello e Toth (1990), que utiliza um método enumerativo (*branch-and-bound*) com um critério de dominância embutido.

Os resultados da comparação foram divididos em duas tabelas: a primeira para testes apresentando 120 itens (Tabela 5.2) e a segunda apresentando 250 itens (Tabela 5.3). Cada tabela mostra, para cada uma das 10 instâncias: o número da instância (problema); o número mínimo teórico de bins (dado pelo quociente entre a soma dos tamanhos dos itens e a capacidade total dos *bins*); e os respectivos resultados do algoritmo proposto, do *GGA* e do *MTP*, com o número de *bins* utilizados e a perda em relação ao número teórico.

**Tabela 5.2 - Resultados do algoritmo para resolução do *BPP* (120 itens)**

<b>Problema</b>	<b>Bins teórico</b>	<b>Método proposto</b>		<b>HGGA</b>		<b>MTP</b>	
		<b>Bins</b>	<b>Diferença</b>	<b>Bins</b>	<b>Diferença</b>	<b>Bins</b>	<b>Diferença</b>
1	48	48	0	48	0	48	0
2	49	49	0	49	0	49	0
3	46	46	0	46	0	46	0
4	49	49	0	49	0	49	0
5	50	50	0	50	0	50	0
6	48	48	0	48	0	48	0
7	48	48	0	48	0	48	0
8	49	49	0	49	0	49	0
9	50	50	0	51	1	51	1
10	46	46	0	46	0	46	0
<b>Totais</b>		<b>0</b>		<b>1</b>		<b>1</b>	

**Tabela 5.3 - Resultados do algoritmo para resolução do *BPP* (250 itens)**

<b>Problema</b>	<b>Bins teórico</b>	<b>Método proposto</b>		<b>HGGA</b>		<b>MTP</b>	
		<b>Bins</b>	<b>Diferença</b>	<b>Bins</b>	<b>Diferença</b>	<b>Bins</b>	<b>Diferença</b>
1	99	100	1	99	0	100	1
2	100	101	1	100	0	100	0
3	102	103	1	102	0	102	0
4	100	101	1	100	0	100	0
5	101	102	1	101	0	101	0
6	101	103	2	101	0	103	2
7	102	103	1	102	0	102	0
8	103	105	2	104	1	104	1
9	105	106	1	105	0	106	1
10	101	102	1	101	0	102	1
<b>Totais</b>		<b>12</b>		<b>1</b>		<b>6</b>	

Observando os resultados da Tabela 5.2 com dados de entrada para 120 itens, percebe-se que o método proposto foi mais eficiente que o *GGA* de Falkenauer (1996) e o *MTP* de Martello e Toth (1990), alcançando a solução ótima para todas as 10 instâncias testadas. Entretanto, o método proposto perdeu eficiência quando o número de itens foi aumentado para 250, o que pode ser percebido observando os resultados da Tabela 5.3. Nesta segunda etapa de testes, o algoritmo genético híbrido de Falkenauer (1996) obteve os melhores resultados; mostrando ser mais eficiente até mesmo se comparado ao método *MTP* na medida em que se aumenta a quantidade de itens.

Com isso, conclui-se que o algoritmo proposto fornece excelentes resultados até uma certa quantidade de itens como dado de entrada. Levando-se em conta que para o problema proposto no presente trabalho o número de itens agrupados em cada fase do algoritmo é bem inferior a quantidade de itens testada acima, pode-se concluir que este método proposto é ideal para o problema real, e que estas perdas apresentadas na Tabela 5.3 não serão verificadas com dados efetivos da empresa. Vale lembrar também que no modelo completo, este procedimento é repetido inúmeras vezes, uma para cada cidade ou agrupamento e também para o subconjunto de clientes com restrição de tamanho de veículo; o que reduz ainda mais a quantidade de itens a alocar em cada uma destas etapas.

Até aqui, foi comprovada a eficiência do algoritmo utilizado nas Fases I e II, que basicamente é um algoritmo para resolução do problema de *bin-packing*. Na seção seguinte o algoritmo completo (com as três fases) é testado e comparado a um programa utilizado atualmente pela empresa estudada no presente trabalho.

## 5.2 - Comparação com o procedimento atual

Numa segunda etapa, a abordagem completa proposta no presente trabalho foi testada considerando dados reais de uma empresa de cimento. A abordagem foi comparada a um método atualmente utilizado pela mesma, baseado em heurística para agrupamento das cargas, chamado *Pegasus*.

O *Pegasus* é um método desenvolvido internamente, baseado em uma heurística simples de agrupamento de cargas, cuja prioridade é encontrar a solução que utiliza o menor número de veículos (ver Anexo A). Em segundo lugar, considera a menor distância entre os pontos. Desta forma, o método oferece resultados muito bons, mas apesar de poder trabalhar com uma base de dados de entrada relativamente grande, possui a restrição de considerar somente frota homogênea. Esta deficiência do *Pegasus* faz com que os operadores executem rotinas seguidas para cada tipo de veículo, aumentando o tempo de operação e podendo reduzir significativamente a qualidade da solução gerada.

Tendo em vista a limitação do *Pegasus* supramencionada, os dados de entrada para comparação continham apenas clientes com o mesmo tipo de veículo máximo permitido. Foram consideradas 15 instâncias reais para comparação, cada qual representada por entregas pertencentes a um dia normal partindo de um único depósito. Essas cargas estão espalhadas em diversos municípios num raio de até 500 km do centro. Os resultados foram comparados em relação ao número mínimo de veículos designados para realizar as entregas e o custo total provindo do pagamento do frete aos transportadores, conforme o cálculo descrito no item 3 deste trabalho. Os resultados obtidos estão demonstrados na Tabela 5.4, abaixo:

**Tabela 5.4 - Resultados do método proposto comparados ao método atual**

<b>Teste</b>	<b>Quantidade de entregas</b>	<b>Método Pegasus</b>		<b>Método Proposto</b>		<b>Variação</b>	
		<b>Veículos</b>	<b>Custo</b>	<b>Veículos</b>	<b>Custo</b>	<b>Veículos</b>	<b>Custo</b>
1	47	30	3,012	29	2,976	-1	-1.2%
2	53	30	3,285	30	3,234	0	-1.6%
3	69	24	22,278	23	21,561	-1	-3.2%
4	72	23	20,951	21	19,465	-2	-7.1%
5	55	20	17,741	19	17,245	-1	-2.8%
6	59	19	17,757	18	17,346	-1	-2.3%
7	56	21	19,835	20	19,447	-1	-2.0%
8	58	18	17,444	17	16,702	-1	-4.3%
9	75	25	22,996	24	22,453	-1	-2.4%
10	64	21	19,268	20	18,563	-1	-3.7%
11	60	18	17,287	17	16,868	-1	-2.4%
12	70	23	21,745	22	21,594	-1	-0.7%
13	46	15	14,417	13	12,996	-2	-9.9%
14	64	23	19,237	22	18,839	-1	-2.1%
15	51	18	17,014	16	15,434	-2	-9.3%
<b>Média</b>		<b>60</b>	<b>22</b>	<b>16,951</b>	<b>21</b>	<b>16,315</b>	<b>-1</b>
							<b>-3.7%</b>

A partir dos resultados desta simulação, percebe-se que o método proposto apresenta economias significativas em relação ao método *Pégusus*. Em média, o método proposto apresentou uma economia média de 3.7% sobre o método atual. Além disso, para a maioria dos testes obteve menos veículos alocados, o que pode representar um ganho extra em aumento de capacidade de transporte. Isso comprova a eficiência do algoritmo proposto e a oportunidade de gerar economias se aplicado na prática.

### 5.3 - Comparação com método exato

Na última etapa de testes, o desempenho do método proposto foi analisado frente aos resultados obtidos com o modelo de programação matemática do problema da seção 3.3. Foram consideradas todas as restrições impostas pelo problema, inclusive a restrição quanto ao tamanho máximo de veículo utilizado. Entretanto, este método só pode ser testado para problemas com poucas entregas, pois para problemas maiores o mesmo incorria em tempos computacionais muito elevados.

A eficiência do método proposto foi analisada tomando-se oito instâncias reais. Assim como na seção 5.2, os dados de entrada foram retirados da situação real, sendo

comparados custo e tempo computacional requeridos por ambas as abordagens. O método exato foi resolvido através do programa *What's Best 8.0*, que utiliza a plataforma do *Microsoft Excel XP*. A CPU utilizada é a mesma descrita para o método proposto. Os resultados são apresentados na Tabela 5.5, abaixo:

**Tabela 5.5 – Resultados do método proposto comparados aos do método exato**

<b>Teste</b>	<b>Quantidade de entregas</b>	<b>Método exato</b>		<b>Método proposto</b>		<b>Desvio Custo</b>
		<b>Custo</b>	<b>Tempo comp (segundos)</b>	<b>Custo</b>	<b>Tempo comp (segundos)</b>	
1	12	3257	90	3257	0,98	0,0%
2	12	2443	12	2459	0,31	0,7%
3	14	3217	4272	3217	0,81	0,0%
4	13	746	132	746	1,00	0,0%
5	13	709	42	709	0,33	0,0%
6	13	764	12	764	0,27	0,0%
7	13	740	96	740	0,61	0,0%
8	13	523	84	523	0,33	0,0%

Através dos resultados obtidos, observa-se que o método proposto atingiu a solução ótima (menor custo) em sete dos oito problemas testados. Além disso, nota-se que o método proposto obteve os resultados em tempos computacionais bem inferiores ao método exato. Deve-se ressaltar que para um modelo para o qual se pretende a utilização como uma ferramenta de nível operacional, tempos acima de 1 minuto já são considerados muito elevados. Percebe-se também que para alguns casos o tempo computacional cresce exponencialmente para os testes com o método exato, apenas acrescentando uma entrega a mais aos dados de entrada, como é o caso do teste 3. Este resultado mostra que o método proposto além de apresentar resultados satisfatórios, é razoavelmente rápido; o que é fundamental para que a utilização uma ferramenta de nível operacional seja viável quando aplicada à prática.

## Capítulo 6 - Conclusões e Recomendações

O presente trabalho propôs a solução do problema de agrupamento de entregas de cimento através da construção de um modelo heurístico, baseado em métodos para resolução de problemas de *bin-packing* e roteirização de veículos. Trata-se de um tema relevante, pois se levou em conta um modelo de tarifação muito utilizado no Brasil para pagamento de transportadores terceiros. Além disso, ofereceu uma abordagem interessante para o problema da falta de informações quanto ao detalhamento na localização exata dos pontos de entrega.

O algoritmo foi dividido em três fases, representando três prioridades no agrupamento das cargas. Na primeira fase, as entregas do mesmo cliente são agrupadas; na segunda fase, passa-se a agrupar as entregas pertencentes à mesma cidade e, por fim, na última fase o restante das entregas é agrupado levando-se em conta a distância entre os pontos. Esta divisão é muito útil para diminuir a base de dados em cada execução, incorrendo em tempos computacionais menores e ideais para aplicação da abordagem.

Na etapa dos experimentos computacionais, os mesmos também foram divididos em três etapas. A primeira etapa dos testes foi concentrada na validação da eficiência do algoritmo de resolução do problema de *bin-packing*, presentes nas fases I e II do algoritmo. Os resultados foram comparados aos de outros métodos eficientes presentes na literatura para resolução do mesmo tipo de problema. Através deles, comprovou-se a eficiência do método para a resolução do *bin-packing* através de resultados bem satisfatórios, para a quantidade de itens apresentados no problema tratado.

Em seguida, a abordagem completa foi comparada ao método atual utilizado na empresa. Desta vez, levando-se em conta as distâncias, mas considerando a frota homogênea, devido à limitação do método atual. Os resultados apresentados foram novamente bem satisfatórios com uma economia média de 3,7%, que representa milhões de reais anuais economizados, se considerada a aplicação do método na empresa estudada.

Por fim, os resultados do método proposto foram comparados aos de um método exato, levando-se em conta todas as restrições impostas pelo problema. Contudo, os tamanhos dos problemas considerados foram pequenos devido aos tempos computacionais muito elevados obtidos pelo método exato. Mais uma vez, os resultados obtidos foram excelentes, chegando aos valores ótimos na maioria dos testes e incorrendo de tempos computacionais bem inferiores.

Todos os testes comprovaram a eficiência do algoritmo proposto, bem como sua boa aplicabilidade à prática. Além de considerar a restrição que muitas empresas apresentam quanto à ausência de mapas digitais para localização exata de seus clientes, o método apresentou excelentes resultados em todas as etapas dos experimentos, gerando ganhos em custos e capacidade da frota. Finalmente, os resultados foram obtidos em tempos bem satisfatórios, fundamental para que se torne uma ferramenta de apoio a decisões no futuro.

Como recomendações para trabalhos futuros, sugere-se a alteração da representação cromossômica utilizada no presente trabalho pela apresentada por Falkenauer (1996) para resolução de problemas de *bin-packing*, através de um algoritmo genético híbrido. Esta obteve resultados melhores para problemas maiores, o que pode ser o caso de algumas empresas. Outra recomendação é testar mudanças nos operadores dos algoritmos genéticos como outros métodos de seleção, *crossover* e mutação. A qualidade da população inicial também pode ser melhorada utilizando outras heurísticas construtivas eficientes. Além disso, outras metaheurísticas também podem ser testadas para resolução do problema de *bin-packing* como Busca Tabu, *Simulated Annealing*, *GRASP*, dentre outras.

Por fim, como estrutura do problema, sugere-se a consideração de outras restrições como janelas de tempo. Além disso, sugere-se a aplicação dos experimentos em problemas maiores, com todas as restrições, visto que não foi possível testá-los devido às limitações do método exato.

## Referências Bibliográficas

- Altinel, I.K. e Oncan, T. (2005) A New Enhancement of the Clarke and Wright Savings Heuristic for the Capacitated Vehicle Routing Problem. *Journal of Operation Research Society*, v. 56, p. 954-961.
- Assad, A.A. (1988) Modeling and Implementation Issues in Vehicle Routing. In: *Vehicle Routing: Methods and Studies*, Elsevier Sciences Publishers, Amsterdam, p. 7-45.
- Ballou, R.H. (1985) *Business Logistics Management – Planning And Control*. 2.ed. New Jersey: Prentice Hall, p. 445-450.
- Beasley, J.E. (1990) OR-Library: Distribution Test Problems by Electronic Mail. *Journal of the Operational Research Society*, v. 41, p. 1069-1072.
- Beasley, J.E. (1999) *Population Heuristics*. The Management School, Imperial Colleg, London.
- Belfiore, P.P (2005) *Scatter Search Para Problemas de Roteirização de Veículos com Frota Heterogênea, Janelas de Tempo e Entregas Fracionadas*. São Paulo. Dissertação (Doutorado) - Escola Politécnica da Universidade de São Paulo, São Paulo.
- Bodin, L.; Golden, B. (1981) Classification in Vehicle Routing and Scheduling. *Networks*, v. 11, n. 2, p. 63-211.
- Bodin, L.; Golden, B.; Assad, A. e Ball, M. (1983) Routing and Scheduling of Vehicles and Crews – The State of the Art. *Computers and Operations Research*, v. 10, p. 63-211.
- Bonasser, U. O. (2005) *Meta-heurísticas Híbridas para Solução do Problema de Roteirização de Veículos com Múltiplos Depósitos e Frota Heterogênea Fixa*:

*Aplicação na Força Aérea Brasileira.* 288 p. Tese (Doutorado) - Escola Politécnica, Universidade de São Paulo, São Paulo.

Boujorlly, J.M. e Rebetez, V. (2005) An Analysis of Lower Bound Procedures for the Bin Packing Problem. *Computers & Operations Research*, v. 32, p. 395-405.

Brandão, J. (2008) A Deterministic Tabu Search Algorithm for the Fleet Size and Mix Vehicle Routing Problem. *European Journal of Operational Research*, doi:10.1016/j.ejor.2007.05.059.

Brejon, S.R.C. (1998) *Algoritmo para Resolução do Problema de Programação do Transporte de Suprimentos para Unidades Marítimas de Exploração do Petróleo.* 132 p. Dissertação (Mestrado em Engenharia Naval) – Escola Politécnica da Universidade de São Paulo, São Paulo.

Christofides, N. (1985) *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*, John Wiley & Sons.

Clarke, G e Wright, J.W. (1964) Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. *Operations Research*, v. 12, p. 568-581.

Coffman, E.G.; Garey, M.R.; Johnson, D.S. (1984). Approximation Algorithms for Bin-Packing – an Updated Survey. *Algorithm Design for Computer System Design*, Springer, Vienna, p. 49-106.

Cunha, C.B. (1997) *Uma Contribuição Para o Problema de Roteirização de Veículos Com Restrições Operacionais.* 222 p. Tese (Doutorado) – Escola Politécnica, Universidade de São Paulo, São Paulo.

Cunha, C.B. (2000) Aspectos Práticos da Aplicação de Modelos de Roteirização de Veículos a Problemas Reais. *Transportes*, Rio de Janeiro, v.8, n.2, p. 51-74.

Cunha, C.B. (2006) *Contribuição à Modelagem de Problemas em Logística e Transportes*. 315 p. Tese de Livre Docência em Logística e Sistemas de Transporte - Departamento de Engenharia de Transportes da Universidade de São Paulo, São Paulo.

Davis, L. (1991) *Handbook of Genetic Algorithms*. Van Nostrand. New York.

Dell'Amico, M.; Monaci, M.; Pagani, C.; Vigo, D. (2007) Heuristic Approaches for the Fleet Size and Mix Vehicle Routing Problem with Time Windows. *Transportation Science*, v. 41, n. 4, p. 516-526.

Dror, M. e Trudeau, P. (1990). Split Delivery Routing. *Naval Research Logistics*, v. 37, n.3, p. 383-402.

Dror, M.; Laporte, G. e Trudeau (1994) Vehicle Routing with Split Deliveries. *Discrete Applied Mathematics*, v. 50, n. 3, p. 229-254.

Dyckhoff, H. e Finke, U. (1992) *Cutting and Packing in Production and Distribution: Typology and Bibliography*, Springer-Verlag Co, Heidelberg.

Falkenauer, F. (1996) A Hybrid Grouping Genetic Algorithm for Bin Packing. *Journal of Heuristics*, v. 2, p. 5-30.

Falkenauer, F. e Delchambre, A. (1992) A Genetic Algorithm for Bin Packing and Line Balancing. *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*, p. 1186-1192.

Feriancic, G. (2005) *Modelagem Matemática do Problema de Programação de Entregas de Derivados de Petróleo*. 104 p. Dissertação (Mestrado) – Escola Politécnica, Universidade de São Paulo, São Paulo.

Fisher, M. e Jaikumar, R. (1981) A Generalized Assignment Heuristics for Vehicle Routing. *Networks*, v. 11, n. 2, p. 109-124.

Galvão, F. A. (2004). *Otimização do Sistema de Coleta de Resíduos de Biomassa de Madeira para Fins Energéticos*. 81p. Dissertação (Mestrado) – Escola Politécnica, Universidade de São Paulo, São Paulo.

Garey, M.R.e Johnson, D.S. (1999). *Computers and Intractability: A guide to the Theory of NP-Completeness*. W.H. Freeman, San Francisco.

Gen, M. e Cheng, R. (1997) *Genetic Algorithms and Engineering Design*, Wiley, New York, 411p.

Gendreau, M.; Hertz, A.; Laporte, G. (1992) New Insertion and Postoptimization Procedures for the Traveling Salesman Problem. *Operations Research*, v. 40, p. 1086-1094.

Gendreau, M.; Laporte, G.; Musaraganyi, C.; Taillard, E.D. (1999) A Tabu Search for the Heterogeneous Fleet Vehicle Routing Problem. *Computers & Operations Research*, v.26, p. 1153-1173.

Gillet, B.L. e Miller, L. (1974) A Heuristic Algorithm for the Vehicle Dispatch Problem. *Operations Research*, v. 22, n. 4, p. 340-349.

Goldbarg, M.C.; Luna, H.P. (2000) *Otimização Combinatória e Programação Linear – Modelos e Algoritmos*. Editora Campus.

Goldberg, D.E (1989) *Genetic Algorithm in Search, Optimization and Machine Learning*. Reading, M.A: Addison-Wesley

Golden, B.L. (1976) Approaches to the Cutting-Stock Problem. *AIEE Transactions*, v. 8, p. 265-274.

Golden, B.L.; Ball, M.; Bodin, L (1981) Current and Future Research Directions in Network Optimization. *Computers & Operations Research*, v. 8, n. 2, p. 71-81.

Golden, B.L.; Assad, A.; Levy, L e Gheysens, F. (1984) The Fleet Size and Mix Vehicle Routing Problem. *Computers & Operations Research*, v. 11, n. 1, p. 49-65.

Holland, J. (1975) *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Michigan, USA.

Johnson, D.S. (1973) Near-Optimal Bin Packing Algorithms. *Technical Report MAC TR-109*, Project MAC, Massachusetts Institute of Technology, Cambridge, MA.

Johnson, D.S.; Demers, A.; Ulman, J.D.; Garey, M.R.; Graham, R.L. (1974) Worst-Case Performance Bounds for Simple One-Dimensional Packing Algorithms. *SIAM Journal on Computing*, v. 3, p. 299-325.

Kang, J. e Park, S. (2003) Algorithms for the Variable Sized Bin Packing Problem. *European Journal of Operation Research*, v. 147, p. 365-372.

Laporte, G.; Gendreau, M.; Potvin, J.Y.; Semet, F. (2000) Classical and Modern Heuristics for the Vehicle Routing Problem. *International Transactions in Operational Research*, Oxford, v.7, n. 4/5, p. 285-300.

Martello, S. e Toth, P. (1990) *Knapsack Problems: Algorithms and Computer Implementations*. Wiley.

Michalewicz, Z. (1994) *Genetic Algorithms + Data Structures = Evolution Programs*. 2.ed. Springer-Verlag, USA.

Mourad, F. A. (2005) *O problema de roteirização de veículos com carga completa e janelas de tempo*. 248 p. Dissertação (Mestrado) – Escola Politécnica, Universidade de São Paulo, São Paulo.

OR-LIBRARY. Produced by J. E. Beasley. Disponível em <<http://people.brunel.ac.uk/~mastjjb/jeb/info.html>>. Acesso em: 01 fev. 2008.

Poh, K.L.; Choo K.W. e Wong, C.G. (2005) A Heuristic Approach to the Multi-period Multi-commodity Transportation Problem. *Journal of Operation Research Society*, v. 56, p. 708-718.

Rochat, Y.; Taillard, E.D. (1995) Probabilistic Diversification and Intensification in Local Search for Vehicle Routing. *Journal of Heuristics*, v.1, p. 147-167.

Ronen, D. (1988) Perspectives on Practical Aspects of Truck Routing and Scheduling. *European Journal of Operational Research*, v.35, n.2, p.137-145.

Rosenkrantz, D.; Stearns, R. e Lewis, P. (1977) An Analysis of Several Heuristics for the Traveling Salesman Problem. *SIAM J. Comp.*, v. 6, p. 563-581.

Schrimpf, G.; Schneider, J.; Stamm-Wilbrandt, H.; Dueck, G. (2000) Record Breaking Optimization Results Using the Ruin and Recreate Principle. *J. Computational Phys.*, v. 159, p. 139-171.

Silver, E.A. (2004) An Overview of Heuristic Solution Methods. *Journal of the Operational Research Society*, 55(9), p. 936-956.

Solomon, M.M. (1987) Algorithms for the Vehicle Routing and Scheduling Problem with Time Window Constraints. *Operations Research*, v.32, p. 254-265.

Solomon, M.M.; Desrosiers, J. (1988) Time Window Constrained Routing and Scheduling Problem. *Transportation Science*, v.22, n.1, p. 1-13.

Souza, E. C. (1999) *Modelagem e Resolução de um Problema de Transporte do Tipo: Carga Única-coleta e Entrega com Janelas de Tempo*. 89 p. Dissertação (Mestrado) – Escola Politécnica, Universidade de São Paulo, São Paulo.

Taillard, E. (1999) A Heuristic Column Generation Method for Heterogeneous fleet VRP. *RAIRO*, v. 33(1), p. 1-14.

Teixeira, R. G. (2001) *Heurística para o Problema de Dimensionamento e Roteirização de uma Frota Heterogênea Utilizando o Algoritmo Out-of-kilter.* 118 p. Dissertação (Mestrado) – Escola Politécnica, Universidade de São Paulo, São Paulo.

Teixeira, R. G. e Cunha, C. B. (2002) Heurísticas para o Problema de Dimensionamento e Roteirização de uma Frota Heterogênea Utilizando o Algoritmo Out-of-kilter. *Transportes*, v.10, n.2.

Wu, L. (2007) *O Problema de Roteirização Periódica de Veículos.* 109p. Dissertação (Mestrado) – Escola Politécnica, Universidade de São Paulo, São Paulo.

Zhang, G. (1997) A New Version of On-line Variable-sized Bin Packing. *Discrete Applied Mathematics*, v. 72, p.193-197.

## ANEXO A – O algoritmo do modelo Pégasus

Exemplo simplificado: problema de agrupamento de seis cargas em veículos com capacidade de 30 toneladas.

1. Geração da matriz binária. O algoritmo ordena as cargas pelo peso em ordem crescente. Em seguida, inicia-se a construção de uma matriz binária partindo de duas cargas, sempre do menor para o maior peso. Para cada linha da matriz binária, calcula-se o peso total, multiplicando pelos pesos das respectivas cargas. Esse processo continua até que a soma das cargas superar a capacidade do veículo.

**Cargas →** 1 2 3 4 5 6

**Pesos →**

	7	8	10	10	10	15	$\Sigma$
1	1	0	0	0	0	0	15
1	0	1	0	0	0	0	17
1	0	0	1	0	0	0	17
1	0	0	0	0	1	0	17
1	0	0	0	0	0	1	22
0	1	1	0	0	0	0	18
0	1	0	1	0	0	0	18
0	1	0	0	1	0	0	18
...							
1	1	1	1	1	0	0	35

**Matriz Binária**

Ordenar pesos crescentemente

Critério de parada:  
Soma dos mínimos > capacidade

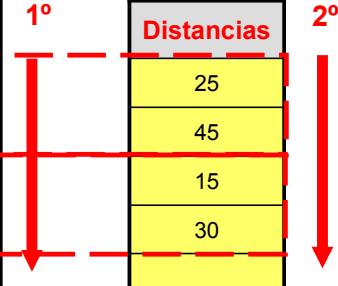
2. No passo seguinte, as linhas da matriz são ordenadas decrescentemente em relação ao peso total. O modelo então partindo da primeira linha agrupará as cargas, desde que essas não sejam coincidentes nas linhas anteriores. O processo termina depois que todas as cargas foram agrupadas.

1	2	3	4	5	6	$\Sigma$
7	8	10	10	10	15	Σ
1	1	0	0	0	1	30
0	0	1	1	1	0	30
0	1	1	1	0	0	28
0	1	1	0	1	0	28
...						

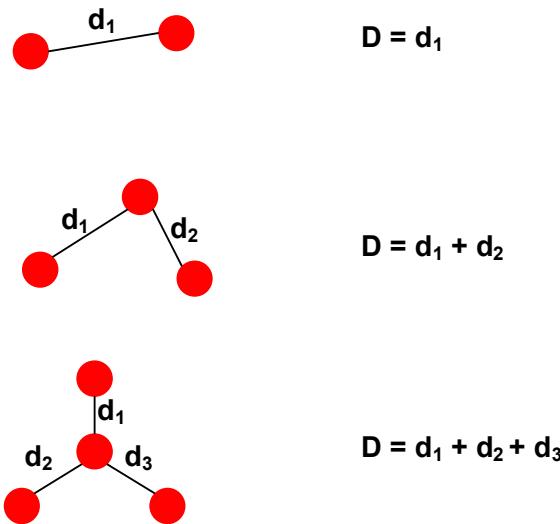


3. No caso de agrupamento de cargas de cidades diferentes, as distâncias entre as cidades são utilizadas como segundo critério de ordenação. Com isso, o modelo tende a agrupar as cargas mais próximas.

1	2	3	4	5	6	$\Sigma$	1º	Distâncias	2º
7	8	10	10	10	15	Σ			
1	1	0	0	0	1	30			
0	0	1	1	1	0	30			
0	1	1	1	0	0	28			
0	1	1	0	1	0	28			
...									



Nesta etapa, o cálculo da distância entre as cidades é dada pelo método Minimum Spanning Tree.



4. O modelo é dividido em três fases. Agrupando prioritariamente as cargas por cliente, cidade e grupo de cidades, nesta seqüência. O algoritmo acima é aplicado para cada fase isoladamente, obtendo-se então a solução no final das três fases.